Sensor Web Enablement

Installation Guide
for
Sensor Observation Service
with Generalizer, Diagram and KML extension
Version 2.0
# Document Change Control

<table>
<thead>
<tr>
<th>Revision Number</th>
<th>Date Of Issue</th>
<th>Author(s)</th>
<th>Brief Description Of Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2010-11-22</td>
<td>Carsten Hollmann</td>
<td>initial version</td>
</tr>
<tr>
<td>0.2</td>
<td>2012-03-21</td>
<td>Carsten Hollmann</td>
<td>Update of used tags</td>
</tr>
<tr>
<td>1.0</td>
<td></td>
<td></td>
<td>final review</td>
</tr>
</tbody>
</table>
Editor

Carsten Hollmann
52°North
Initiative for Geospatial Open Source Software GmbH
Martin-Luther-King-Weg 24
D-48155 Münster

Email: c.hollmann@52north.org
Licence

This document is part of 52°North
Copyright (C) 2008 52°North

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

For more information, contact:
52°North
Martin-Luther-King-Weg 24
48155 Münster
Germany
http://www.52north.org

[This page has been intentionally left blank.]
# Table of Contents

1 Introduction .......................................................................................................................... 7
   1.1 Scope .............................................................................................................................. 7
   1.2 What you are doing following the installation procedure? ............................................. 7
   1.3 Characteristics of the Sensor Observation Service with Generalizer, Diagram and KML extension ................................................................................................................ 7
   1.4 Supported operations of the Sensor Observation Service with Generalizer, Diagram and KML extension ................................................................................................................ 7

2 Requirements ...................................................................................................................... 8

3 Installation Procedure ........................................................................................................ 8
   3.1 Get the programs ............................................................................................................. 8
      3.1.1 Configure Maven ...................................................................................................... 8
   3.2 Get the sources ................................................................................................................ 8
   3.3 Folder structures ............................................................................................................. 9
   3.4 Configure the properties ................................................................................................ 10
      3.4.1 Configure build.properties in 52n-sos-dao-postgis-generalizer-diagram-kml ......................................................................................................................... 10
   3.5 Configure the pom.xml files .......................................................................................... 11
      3.5.1 Configure the main pom.xml in the standard SOS project ........................................ 11
      3.5.2 Configure the 52n-sos-service pom.xml in the standard SOS project ....................... 11
   3.6 Build the web application ............................................................................................... 12
      3.6.1 Configuration of the generalization ......................................................................... 12
      3.6.2 Deploy the web application ....................................................................................... 15
   3.7 Tests .................................................................................................................................. 16
      3.7.1 Example for GetCapabilities request ....................................................................... 16
         3.7.1.1 Capabilities-Request via HTTP GET .................................................................... 16
         3.7.1.2 Capabilities Request via HTTP POST: ................................................................. 16
         3.7.2 Other tests .............................................................................................................. 17

4 Extensions .......................................................................................................................... 18
   4.1 Generalization ................................................................................................................ 18
   4.2 Diagram and KML response in GetObservation request ............................................... 19
4.2.1 Diagram in GetObservation request ................................................................. 19
4.2.2 KML in GetObservation request ................................................................. 21
4.3 Diagram pre-rendering ....................................................................................... 24
4.3.1 Diagram pre-rendering frequency ............................................................. 24
4.3.2 Replacement of invalid characters ............................................................ 24

5 Appendix ............................................................................................................... 26
5.1 Samples of generalized time series ................................................................. 26
5.2 Additional properties in build.properties for the example requests .............. 27
5.3 References ......................................................................................................... 28
1 Introduction

1.1 Scope
This document describes the installation process of the Sensor Observation Service. Since the installation process is quite similar to the installation of the standard Sensor Observation Service, this is only an addition to the installation manual of the standard Sensor Observation Service. Please refer also to this manual if any questions occur during the installation process. A copy of this document can be found in the standard Sensor Observation Service repository.

1.2 What you are doing following the installation procedure?
The base Sensor Observation Service version is the 52°North SOS 3.2.0 release. You will download the source release from the 52°North download page and add some additional packages containing files for the SOS as a Java servlet.

The 52n-sos-dao-postgis-generalizer-diagram-kml extension is a replacement of the SOS 52n-sos-dao-postgis module. It brings own data source adapters for the SOS web service with PostgreSQL and PostGIS. The 52n-sos-generalizer module is an extension for generalizing time series. The 52n-sos-coding-diagram module is an extension diagram rendering and the 52n-sos-coding-kml module is an extension for generating KML files form the data. It is necessary to check out the four repositories. Executing the installation steps [build] will deploy this service in your Apache Jakarta Tomcat web container as an web application (webapp).

1.3 Characteristics of the Sensor Observation Service with Generalizer, Diagram and KML extension
This Version of the SOS is based on the 52°North implementation of the SOS. Major functionality of the standard SOS is supported as well. In addition some more features are available, e.g.:

- Generalize the data
- Query KML in GetObservation request
- Query Diagrams in GetObservation request
- …

1.4 Supported operations of the Sensor Observation Service with Generalizer, Diagram and KML extension
The SOS supports the SOS core profile with the following operations:

- GetCapabilities
- DescribeSensor
- GetObservation

Additionally, all other operation are supported.
2 Requirements

- Windows 2000 or higher [tested with Windows XP SP2]
- Sun JRE/JDK 1.6 [1.6.0_XX]
- Apache Jakarta Tomcat 5.5 or higher [6.0.20]
- SVN-Client
- Maven 2.X [2.2.1, 3.0.3]
- 52N SOS 3.2.0 release
- 52N SOS Postgis/Generalizer/Diagram/KML module
- 52N SOS Generalizer module
- 52N SOS Diagram coding module
- 52N SOS KML coding module

3 Installation Procedure

3.1 Get the programs

- Download Apache Jakarta Tomcat from:
  
  http://jakarta.apache.org/tomcat

  Follow the installation instructions given on the Apache Jakarta Tomcat.

- Download Apache Maven from:
  
  http://maven.apache.org/

  Follow the installation instructions given on the Apache website to install the Apache Maven.

3.1.1 Configure Maven

- See chapter 3.1.1 in the how2install_SOS document.

3.2 Get the sources

- Check out the sos 3.2.0 release sources from 52°North download page:
  

- Check out the 52n-sos-dao-postgis-generalizer-diagram-kml sources from 52°North SVN-Repository:
  
  Host: https://52north.org/svn/
  Repository: swe/main/SOS/Extensions/PostGISGeneralizerDiagramKML/tags/
  Tag: 52n-sos-dao-postgis-generalizer-diagram-kml_20120321
Sensor Web Enablement

Generalizer-Diagram-KML SOS Installation Guide

- Check out the **52n-sos-generalizer** sources from 52°North SVN-Repository:
  
  **Host:** https://svn.52north.org/svn
  
  **Repository:** /swe/main/SOS/Extensions/Generalization/tags/
  
  **Tag:** 52n-sos-generalizer_20120213

- Check out the **52n-sos-coding-diagram** sources from 52°North SVN-Repository:
  
  **Host:** https://svn.52north.org/svn
  
  **Repository:** /swe/main/SOS/Extensions/Diagram/tags/
  
  **Tag:** 52n-sos-coding-diagram_20120321

- Check out the **52n-sos-coding-kml** sources from 52°North SVN-Repository:
  
  **Host:** https://svn.52north.org/svn
  
  **Repository:** /swe/main/SOS/Extensions/KML/tags/
  
  **Tag:** 52n-sos-coding-kml_20120321

### 3.3 Folder structures

The folder structure of the **52n-sos-dao-postgis-generalizer-diagram-kml** module is as follows:

- **conf:** configuration files; contains also the capabilities skeleton
- **src:** source files
- **webapp:** webapp folder with a SOS TestClient and examples

The folder structure of the **52n-sos-generalizer** module is as follows:

- **conf:** configuration files; contains also the capabilities skeleton
- **src:** source files
- **schemas:** XML-schemas for configuration files

The folder structure of the **52n-sos-coding-diagram** module is as follows:

- **conf:** configuration files
- **src:** source files (maven structure)

The folder structure of the **52n-sos-coding-kml** module is as follows:

- **conf:** configuration files
- **src:** source files (maven structure)

For the folder structure of the **standard SOS** see the how2install_SOS document chapter 3.3 in the standard SOS /doc folder.
3.4 Configure the properties

Although you have four repositories in your workspace, you will only need to configure the 52n-sos-dao-postgis-generalizer-diagram-kml project. They will automatically pass the configuration to the standard SOS and compile it while building the SOS with Generalizer, Diagram and KML. You will find the configuration files in the /conf folders of your 52n-sos-dao-postgis-generalizer-diagram-kml project.

3.4.1 Configure build.properties in 52n-sos-dao-postgis-generalizer-diagram-kml

- Open the build.properties file in the /conf folder or the 52n-sos-dao-postgis-generalizer-diagram-kml project and edit the following properties. The explanation contains a hint, if the property must be changed (MANDATORY) in order to allow the correct installation of the SOS or may be changed (OPTIONAL) to alter some default values. Properties marked as ADVANCED should not be changed.

A description of the additional properties is listed below:

<table>
<thead>
<tr>
<th>Property</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The following module specific properties</strong></td>
<td></td>
</tr>
<tr>
<td>conf.sos.path</td>
<td>Path to the point where the configuration files are located. (ADVANCED)</td>
</tr>
<tr>
<td>conf.sos.files</td>
<td>Map with identifier and configuration file name. The id and file name are separated by ‘ ’ and the definitions by ‘;’. By default the following mapping is set (ADVANCED): • GeneralizerConfig GeneralizerConfig.xml • Diagram diagram.config • KML kml.config</td>
</tr>
<tr>
<td>conf.sos.useGeneralizer</td>
<td>Enables the generalization of requested SOS data (default := true). (ADVANCED)</td>
</tr>
<tr>
<td>conf.sos.prebuild.diagram</td>
<td>Indicate whether the SOS should pre render diagrams which can be accessed directly by a link. (default=false) For more information see chapter X.Y. (ADVANCED)</td>
</tr>
<tr>
<td>conf.sos.prebuild.diagram.path</td>
<td>Path where the pre rendered diagrams should be stored (default= [WEBAPP_BASE_PATH]/diagram)(ADVANCED)</td>
</tr>
<tr>
<td>conf.sos.prebuild.diagram.period</td>
<td>Indicates for how many days the diagrams should be pre rendered from actual time to past. (default=7). (ADVANCED)</td>
</tr>
<tr>
<td>A <strong>default SOS property with additional functionality for diagram pre-rendering.</strong></td>
<td></td>
</tr>
<tr>
<td>conf.sos.capabilitiesCacheUpdateInterval</td>
<td>Capabilities Cache Update Interval in minutes (0 = no automatic update). Here also used for starting the diagram pre-rendering. (ADVANCED)</td>
</tr>
<tr>
<td><strong>The following properties in the file are the default SOS properties.</strong> For detailed information see the how2install_SOS document in the doc folder of the default SOS-directory.</td>
<td></td>
</tr>
</tbody>
</table>

- Save changes

There are some additional properties in the build.properties file where you can define the parameters (e.g. offering, procedure, etc) for the examples requests. For more details see section 5.2.
3.5 Configure the pom.xml files

3.5.1 Configure the main pom.xml in the standard SOS project

- Change in modules
  
  `<module>52n-sos-dao-postgis</module>

  to

  `<module>..:/52n-sos-dao-postgis-generalizer-diagram-kml</module>

  and add

  `<module>..:/52n-sos-generalizer</module>
  `<module>..:/52n-sos-coding-diagram</module>
  `<module>..:/52n-sos-coding-kml</module>

  The modules section should look like this:

  `<modules>
    `<module>52n-sos-ogc</module>
    `<module>52n-sos-core</module>
    `<module>52n-sos-coding</module>
    `<module>52n-sos-service</module>
    `<module>..:/52n-sos-dao-postgis-generalizer-diagram-kml</module>
    `<module>..:/52n-sos-coding-diagram</module>
    `<module>..:/52n-sos-coding-kml</module>
  </modules>

- Save pom.xml

3.5.2 Configure the 52n-sos-service pom.xml in the standard SOS project

- Change in dependencies
  
  `<artifactId>52n-sos-dao-postgis</artifactId>

  to

  `<artifactId>52n-sos-dao-postgis-generalizer-diagram-kml</artifactId>

  and change the `<version>` to 1.0.0.

- and add

  `<dependency>
    `<groupId>${pom.groupId}</groupId>
    `<artifactId>52n-sos-generalizer</artifactId>
    `<version>1.0</version>
  </dependency>

  `<dependency>
    `<groupId>org.n52.sensorweb.sos</groupId>
    `<artifactId>52n-sos-coding-diagram</artifactId>
    `<version>1.0</version>
  </dependency>

  `<dependency>
    `<groupId>org.n52.sensorweb.sos</groupId>
    `<artifactId>52n-sos-coding-kml</artifactId>
    `<version>1.0</version>
  </dependency>`
The dependencies section should look like this:

```xml
<dependencies>
  <dependency>
    <groupId>${project.groupId}</groupId>
    <artifactId>52n-sos-dao-postgis-generalizer-dia-
kml</artifactId>
    <version>1.0.0</version>
  </dependency>
  <dependency>
    <groupId>${pom.groupId}</groupId>
    <artifactId>52n-sos-generalizer</artifactId>
    <version>1.0</version>
  </dependency>
  <dependency>
    <groupId>org.n52.sensorweb.sos</groupId>
    <artifactId>52n-sos-coding-diagram</artifactId>
    <version>1.0</version>
  </dependency>
  <dependency>
    <groupId>org.n52.sensorweb.sos</groupId>
    <artifactId>52n-sos-coding-kml</artifactId>
    <version>1.0</version>
  </dependency>
</dependencies>
```

• Save pom.xml

### 3.6 Build the web application

#### 3.6.1 Configuration of the generalization

The SOS is capable of generalizing time series to reduce the amount of data encoded in the O&M-response document. Therefore you need the generalizer extension. This extension contains several implementations of different generalization methods. The abstract class `AbstractGeneralizer` inside the `org.n52.sos.generalizer` package can be extended to implement own generalization methods. Generalizer implementations will be configured by using the `GeneralizerConfig.xml` file found in the `conf` folder of the 52n-sos-generalizer project. At the time of delivery four generalization methods are supported:

- **Skipping every second value**
  
  *This generalizer skips automatically every second value. The last value will be included anyway. No further configurations need to be made.*

- **Selecting every xx value**
  
  *This generalizer takes only every xx value. The last value will be included anyway. The xx value has to be defined by using the `SELECT_VALUE_NUMBER` property.*

- **Skipping values within a certain tolerance**
  
  *This generalizer calculates a line between different points and estimates the distance of the points in between to this line. If they are below a certain value,
these points will not be taken. This value is defined by using a percentage from the span of the requested interval's values. It has to be set by using the TOLERANCE_PERCENT_VALUE property. Additionally a maximum number of points to process can be defined by using the MAX_ENTRIES property. If this amount of point is reached an exception will be thrown. If this value is not defined it will be set to unlimited automatically.

- Skipping values within a certain tolerance (request time period dependant)

This generalizer acts like the one before. Only the percentage value can be set time dependant. Specific values can be set for different request time periods. To set these values use the TOLERANCE_PERCENT_VALUE. The values are defined as follows: Percent/Hours_from-Hours_till, e.g. 0/0-24. That means no tolerance will be applied for the period from 0 to 24 hours. Overlapping of percent/time values is not allowed. All periods from 0 hours to UNL need to be covered. The MAX_ENTRIES can be set as well as described above.

To select and configure the volitional means of generalization edit the GeneralizerConfig.xml file in the conf folder of the 52n-sos-generalizer project. First edit the Generalizer-tag and the DecimalPlaces-tag. Then define the GeneralizerToUse-Attribute. A description of the XML-schema can be found in figure 3.

Take a look at 4.1.7 for some sample graphics that show the effect of generalization.
<?xml version="1.0" encoding="UTF-8"?>
GeneralizerToUse="5" DecimalPlaces="1">
  <Generalizer>
    <ID>1</ID>
    <Name>SkipEverySecond-Generalizer</Name>
    <Description>This generalizer skips every second value.</Description>
    <Procedure>urn:ogc:generalizationMethod:IFGI:SkipEverySecond</Procedure>
    <Class>org.n52.sos.generalizer.implementation.SkipSecondValueGeneralizer</Class>
    <Properties/>
  </Generalizer>
  <Generalizer>
    <ID>2</ID>
    <Name>SelectValue-Generalizer</Name>
    <Description>This generalizer selects only each defined value.</Description>
    <Procedure>urn:ogc:generalizationMethod:IFGI:SelectValue</Procedure>
    <Class>org.n52.sos.generalizer.implementation.SelectValueGeneralizer</Class>
    <Properties>
      <Value key="SELECT_VALUE_NUMBER">4</Value>
    </Properties>
  </Generalizer>
  <Generalizer>
    <ID>3</ID>
    <Name>Tolerance-Generalizer</Name>
    <Description>This generalizer skips values that are within a predefined tolerance.</Description>
    <Procedure>urn:ogc:generalizationMethod:IFGI:Tolerance</Procedure>
    <Class>org.n52.sos.generalizer.implementation.ToleranceGeneralizer</Class>
    <Properties>
      <Value key="TOLERANCE_PERCENT_VALUE">2.0</Value>
    </Properties>
  </Generalizer>
  <Generalizer>
    <ID>4</ID>
    <Name>Tolerance-Generalizer time dependant</Name>
    <Description>This generalizer skips values that are within a predefined tolerance depending on the defined tolerance for a certain time period.</Description>
    <Procedure>urn:ogc:generalizationMethod:IFGI:ToleranceTime</Procedure>
    <Class>org.n52.sos.generalizer.implementation.ToleranceTimeGeneralizer</Class>
    <Properties>
      <!--MAX_ENTRIES is the value for the maximum points the generalizer will handle, otherwise an exception will be thrown; -1 is unlimited-->
      <Value key="MAX_ENTRIES">-1</Value>
      <Value key="PERCENT/HOURS FROM-HOURS TILL --
        TOLERANCE_PERCENT_TIME_VALUE1">0/0-24</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE2">0.5/24-48</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE3">1.0/48-72</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE4">1.5/72-168</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE5">2.0/168-768</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE6">3.0/768-5440</Value>
      <Value key="TOLERANCE_PERCENT_TIME_VALUE7">5.0/5440-UNL</Value>
    </Properties>
  </Generalizer>
</GeneralizerConfiguration>
3.6.2 Deploy the web application

- Make sure that your Tomcat is started.
- Open a command line and assume to the project's folder, e.g. C:\SOS
- Type the following command line expression:
  - mvn -Pwith-deploy install
- The SOS with extensions is now available.
3.7 Tests

Manual tests can be run with own requests. Sample requests can be found in the xml folder. Sending of those requests is described in the following paragraph:

3.7.1 Example for GetCapabilities request

The GetCapabilities-request is the only request which is possible to be sent via HTTP GET- and HTTP POST. The SOS webclient (see below) is attached to enable POST requests to the SOS.

3.7.1.1 Capabilities-Request via HTTP GET

- Open your browser. Type in
  
  `http://your.sos.location:port/webapp_name/sos?REQUEST=GetCapabilities&SERVICE=SOS&ACCEPTVERSIONS=1.0.0`

- The capabilities response document is now shown in your browser.
- You can extend the request above through appending the optional parameters SECTIONS and ACCEPTFORMATS. Look at the SOS specification for more details.

3.7.1.2 Capabilities Request via HTTP POST:

- Open your Internet Browser (e.g. Mozilla Firefox). You can find the SOS Test Client under `http://your.sos.location:port/webapp_name/testClient-v2.html` (Make sure that your tomcat is running)

![SOS TestClient](image)

This TestClient was successfully tested in Firefox 3.5.2, Safari 4.0.3, Opera 9.64 and InternetExplorer 8.0.0001.18702 and should work properly in Firefox 1.0 or higher, Safari 1.2 or higher, Opera 8 or higher and InternetExplorer 5 or higher. The editor is based on CodeMirror (http://marijn.haverbeke.nl/codemirror/).

*Figure 3: Screenshot of the SOS test client 2.0 for capabilities request*
• Chose a request from the drop down list named 'Request Examples' or create your own request and paste your request in the test box.

• Click the „Send“-Button. The capabilities response xml-document is now shown in your browser.

3.7.2 Other tests

• You can try the other XML request files in the xml folder analogous to the HTTP Post based GetCapabilities request (e.g. copy and paste the content of the GetObs_BBOX.xml file).
## 4 Extensions

### 4.1 Generalization

With the generalization extension the user can query a generalized time series directly from the SOS. The generalizers are listed in the Capabilities document as a procedure and you can information about the generalize,r if you do a DescribeSensor request with the generalizer id as parameter. In a GetObservation request aou can chose some valid combination to query a generalized time series.

In the following list the additional combinations are listed.

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No parameter</td>
<td>No procedure in request.</td>
</tr>
<tr>
<td>&lt;procedure&gt;procedureID&lt;/procedure&gt; (…)</td>
<td>One or many procedure(s) as procedure parameter in request.</td>
</tr>
<tr>
<td>&lt;procedure&gt;generalizerID&lt;/procedure&gt; (…)</td>
<td>One or many generalizer(s) as procedure parameter in request.</td>
</tr>
<tr>
<td>&lt;procedure&gt;procedureID,generalizerID&lt;/procedure&gt;</td>
<td>One procedure with one generalizer as procedure parameter in request.</td>
</tr>
<tr>
<td>&lt;procedure&gt;procedureID_1,generalizerID_1&lt;/procedure&gt; or &lt;procedure&gt;procedureID_1,generalizerID_2&lt;/procedure&gt; or &lt;procedure&gt;procedureID_1,generalizerID_1&lt;/procedure&gt; and &lt;procedure&gt;procedureID_2,generalizerID_1&lt;/procedure&gt; or &lt;procedure&gt;procedureID_1,generalizerID_1&lt;/procedure&gt; and &lt;procedure&gt;procedureID_2,generalizerID_2&lt;/procedure&gt;</td>
<td>One procedure with different generalizers or Many procedures with the same generalizer or Many procedures with different generalizers as procedure parameter in request.</td>
</tr>
<tr>
<td>&lt;procedure&gt;procedureID_1,generalizerID_1&lt;/procedure&gt; or &lt;procedure&gt;procedureID_1&lt;/procedure&gt; or &lt;procedure&gt;procedureID_1,generalizerID_1&lt;/procedure&gt; or &lt;procedure&gt;procedureID_2&lt;/procedure&gt;</td>
<td>Combinations of procedure and generalizer and procedures as procedure parameter in request.</td>
</tr>
</tbody>
</table>

All other combinations are not supported and the response is an exception.
The default implemented generalizers are:

<table>
<thead>
<tr>
<th>Name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkipEverySecond-Generalizer</td>
<td>This generalizer skips every second value.</td>
</tr>
<tr>
<td>SelectValue-Generalizer</td>
<td>This generalizer selects only each defined value.</td>
</tr>
<tr>
<td>Tolerance-Generalizer</td>
<td>This generalizer skips values that are within a predefined tolerance.</td>
</tr>
<tr>
<td>Tolerance-Generalizer time dependant</td>
<td>This generalizer skips values that are within a predefined tolerance depending on the defined tolerance for a certain time period.</td>
</tr>
<tr>
<td>Douglas-Peucker-Generalizer</td>
<td>This generalizer skips non-characteristic values depending on a given tolerance.</td>
</tr>
</tbody>
</table>

**4.2 Diagram and KML response in GetObservation request**

The SOS supports KML files or diagrams as response of a GetObservation request. Difference to the SOS 1.0 Specification, the featureOfInterest parameter is mandatory in a KML or DIAGRAM request and the observableProperty parameter is restricted to only one value. Here is a short description which parameters are used:

- Build a GetObservation with the following parameters
  - offering (MANDATORY)
  - observedProperty (only one) (MANDATORY)
  - featureOfInterest id (only one) (MANDATORY)
  - procedure id (only one) (MANDATORY)
  - responseFormat (MANDATORY)
  - eventTime with TimePeriod (OPTIONAL)
- send the request to the SOS (via TestClient-v2 section 3.7).

**Example requests:**

**4.2.1 Diagram in GetObservation request**

The diagram rendering is excluded into a separate module and renders a diagram from the requested values. The 52n-sos-coding-diagram module has a separate configuration file (52n-sos-coding-diagram/conf/diagram.config) where some properties can be set to modify the styling of the returned diagram. Here is a list of the properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOS_NO_DATA_VALUE</td>
<td>The SOS no data value. Used to eliminate no data values from the diagram view.</td>
</tr>
<tr>
<td>DIAGRAM_WIDTH</td>
<td>The width of the diagram</td>
</tr>
<tr>
<td>DIAGRAM_HEIGHT</td>
<td>The height of the diagram</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SHOW_TITLE</td>
<td>Indicates whether a title should be set. By default the title is the featureOfInterest id.</td>
</tr>
<tr>
<td>SHOW_HORIZONTAL_GRID_LINES</td>
<td>Indicate whether horizontal grid lines should be plotted</td>
</tr>
<tr>
<td>SHOW_VERTICAL_GRID_LINES</td>
<td>Indicate whether vertical grid lines should be plotted</td>
</tr>
<tr>
<td>LINE_COLOR</td>
<td>Define the color of the line by color name, e.g. RED, BLUE, GREEN, or by RGB value definition, e.g. (123,1,89).</td>
</tr>
<tr>
<td>COPYRIGHT_TEXT</td>
<td>Definition of the copyright text which should be plotted.</td>
</tr>
<tr>
<td>X_AXIS_LABEL</td>
<td>The label name of the x axis.</td>
</tr>
<tr>
<td>NO_DATA_MESSAGE</td>
<td>The text which should be plotted if no data are available.</td>
</tr>
</tbody>
</table>

During the Maven build process this file is copied and is then located in the [TOMCAT_HOME]/webapps/[SOS_NAME]/WEB-INF/conf folder. It is mandatory to define this file in the SOS build.properties file or in the dssos.config file of the deployed webapp.

Add the following definition 'Diagram diagram.config' to the following properties (by default set in the SOS build.properties file):

- conf.sos.files in build.properties (before the SOS is deployed)
- CONFIGURATION_FILES in dssos.config (in the deployed SOS)

To get a diagram as response from the SOS you have to set the following value in the parameter `<responseFormat>`:

- image/jpeg

Here is an example request of a diagram request and the returned diagram.
4.2.2 KML in GetObservation request

The KML coding is excluded into a separate module and generates a KML file from the request. The 52n-sos-coding-kml module has a separate configuration file (52n-sos-coding-kml/conf/kml.config). It contains the definition of the used XSLT transformation file and some indicators how the diagram can be accessed (link in the KML file):

```xml
<?xml version="1.0" encoding="UTF-8"?>
<GetObservation xmlns="http://www.opengis.net/sos/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:om="http://www.opengis.net/om/1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/sos/1.0 http://schemas.opengis.net/sos/1.0.0/sosGetObservation.xsd"
service="SOS" version="1.0.0">
<offering>WASSERTEMPERATUR_ROHDATEN</offering>
<procedure>Temperatur-Stoer-Sperrwerk_Bp_5970040</procedure>
<observedProperty>Temperatur</observedProperty>
<featureOfInterest>
<ObjectID>Stoer-Sperrwerk_Bp_5970040</ObjectID>
</featureOfInterest>
</GetObservation>
Listing 1: GetObservation request example for Diagram response.

**Figure 4: Diagram response**
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XSL_10</td>
<td>Path to the XSLT file.</td>
</tr>
<tr>
<td>QUERY_DIAGRAM</td>
<td>Indicates whether a HTTP-GET request is set (true) or a direct link (false) to get the diagram (default=true)</td>
</tr>
<tr>
<td>QUERY_PATH</td>
<td>The URL to the SOS for the HTTP-GET request. Mandatory if QUERY_DIAGRAM=true.</td>
</tr>
<tr>
<td>DIAGRAM_DOWNLOAD_BASE_PATH</td>
<td>Base path to the pre rendered diagrams. Mandatory if QUERY_DIAGRAM=false.</td>
</tr>
<tr>
<td>DIAGRAM_DAYS</td>
<td>The number of days which should be queried from actual time to the past for the diagram. Only set if request did not contain a eventTime parameter. Used if QUERY_DIAGRAM=true.</td>
</tr>
<tr>
<td>DIAGRAM_WIDTH</td>
<td>The width of the diagram</td>
</tr>
<tr>
<td>DIAGRAM_HEIGHT</td>
<td>The height of the diagram</td>
</tr>
<tr>
<td>KML_RESPONSE_FORMAT</td>
<td>KML response format definition. Valid are:</td>
</tr>
<tr>
<td></td>
<td>• application/vnd.google-earth.kml+xml (DEFAULT)</td>
</tr>
<tr>
<td></td>
<td>• text/plain</td>
</tr>
<tr>
<td>DIAGRAM_RESPONSE_FORMAT</td>
<td>Diagram response format definition, set in the HTTP-GET URL of KML response.</td>
</tr>
</tbody>
</table>

During the Maven build process this file is copied and is then located in the [TOMCAT_HOME]/webapps/[SOS_NAME]/WEB-INF/conf folder. It is mandatory to define this file in the SOS build.properties file or in the dssos.config file of the deployed webapp.

Add the following definition 'KML kml.config' to the following properties (by default set in the SOS build.properties file):

- conf.sos.files in build.properties (before the SOS is deployed)
- CONFIGURATION_FILES in dssos.config (in the deployed SOS)

To get a diagram as response from the SOS you have to set the following value in the parameter <responseFormat>:

- application/vnd.google-earth.kml+xml

Here is an example request of a kml request and the returned kml file.
<?xml version="1.0" encoding="UTF-8"?>
<GetObservation xmlns="http://www.opengis.net/sos/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:om="http://www.opengis.net/om/1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/sos/1.0.0/sosGetObservation.xsd"
service="SOS" version="1.0.0">
<offering>WASSERTEMPERATUR_ROHDATEN</offering>
<precedure>Temperatur-Stoer-Sperrwerk_Bp_5970040</procedure>
<observedProperty>Temperatur</observedProperty>
<featureOfInterest>
<ObjectID>Stoer-Sperrwerk_Bp_5970040</ObjectID>
</featureOfInterest>
<responseFormat>application/vnd.google-earth.kml+xml</responseFormat>
</GetObservation>

Listing 2: GetObservation request example for KML response.

<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns:sa="http://www.opengis.net/sampling/1.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:om="http://www.opengis.net/om/1.0"
xmlns:gml="http://www.opengis.net/gml"
xmlns:swe="http://www.opengis.net/swe/1.0.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://earth.google.com/kml/2.1">
<Document>
<open>0</open>
<Style id="FOI_STYLE_Stoer-Sperrwerk_Bp_5970040">
<BalloonStyle>
<bgColor>ffffff</bgColor>
<text>$[name]</font>$[description]'/id := 'Stoer-Sperrwerk_Bp_5970040'</description>
</Style>
<Placemark>
<name>Stör-Sperrwerk Bp - Kilometer: 50</name>
<description>Feature Of Interest: 'Stör-Sperrwerk Bp - Kilometer: 50'</description>
<styleUrl>#FOI_STYLE_Stoer-Sperrwerk_Bp_5970040</styleUrl>
<Point>
<coordinates>9.40234781963614, 3.82793502621518</coordinates>
</Point>
</Placemark>
</Document>
</kml>

Listing 3: KML response
Here is an HTTP-GET example to query a GetObservation request with KML response which can be used directly in Google Maps:

```
http://maps.google.de/maps?f=q&hl=de&q=[SOS_URL]/sos?service=SOS
&REQUEST=GetObservation&version=1.0.0&offering=WASSERSTAND_ROHDATEN
&procedure=Wasserstand-Koeln_2730010&observedProperty=Wasserstand
&featureOfInterest=Koeln_2730010&responseFormat=text/xml;
subtype=%22kml/2.1%22
```

### 4.3 Diagram pre-rendering

This SOS has an additional functionality to pre-render diagrams for a defined period from the actual time.

To enable pre-rendering you have to set the properties in the build.properties file (before deployment) or in the dssos.config file of the deployed service. After changes in the dssos.config file it is required to reload the SOS.

Here are the properties from the dssos.config file, the corresponding properties from the build.properties file are described in chapter 3.4.1.

- **PREBUILD_DIAGRAM**
  - enables pre-rendering of diagrams

- **PREBUILD_DIAGRAM_PATH**
  - path where the diagrams are stored
  - if the definition contains the string 'basePath', this string will be replaced with the webapps base path
  - The sub structure from this path is `OFFERING_NAME/OBSERVABLE_PROPERTY_ID/FEATURE_OF_INTEREST_ID__PROCEDURE_ID.jpg`

- **PREBUILD_DIAGRAM_PERIOD**
  - The period for which the diagrams should be pre-rendered in days, actual time minus defined days

#### 4.3.1 Diagram pre-rendering frequency

The frequency of the diagram pre-rendering depends on the SOS capabilities cache update interval. The capabilities cache updater is a thread that periodically starts requesting the database and update the SOS internal cache. For diagram pre-rendering the capabilities cache update thread starts a new thread that pre-renders the diagrams. If the diagram pre-rendering thread is not finished until the capabilities cache update thread is started again, no new diagram pre-rendering thread will be started. When the next time the capabilities cache update thread runs and the diagram pre-rendering thread is finished, a new diagram pre-rendering thread will be started.

The capabilities cache update interval can be set in the build.properties file (see chapter 3.4.1).

#### 4.3.2 Replacement of invalid characters

The pre-rendered diagrams are stored in the described structure. In some cases the offering,
observableProperty, procedure or featureOfInterest can contain characters which are not allowed in a path, filename or URL. Then the pre-rendered diagram can not be stored in the file system or the URL is invalid. To prevent this, the not allowed characters will be replaced in the path/filename/URL with '_' depending on the operating system.

Here are the characters which are replaced with '_':

\ / : " * ? < > ; # % = @

Special german characters are replaced as defined in the following:

- ä -> ae
- ö -> oe
- ü -> ue
- Å -> AE
- Ö -> OE
- Ü -> UE
- ß -> ss

If you generate a direct link to the pre-rendered diagrams automatically in a program make sure that the characters mentioned before in offering, observableProperty, procedure and featureOfInterest are replaced with the '_'.

Here is an example for a procedure id:

<table>
<thead>
<tr>
<th>Original</th>
<th>replaced</th>
</tr>
</thead>
</table>
5 Appendix

5.1 Samples of generalized time series

The sample images (figures 5-9) are taken from the ThinSweClient application. All generalized data is derived from the original data shown in figure 5.

![Figure 5: screenshot of the original time series](image)

![Figure 6: generalizer skipped every second value](image)
5.2 Additional properties in build.properties for the example requests

This chapter describes some additional properties in the build.properties file where you can define the parameters (e.g. offering, procedure, etc) for the examples requests. These properties will be automatically set to the examples during the build process.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sos.examples.def.offering</td>
<td>The offering id for the requests.</td>
</tr>
<tr>
<td>sos.examples.def.observableProperty</td>
<td>The observableProperty id for the requests.</td>
</tr>
<tr>
<td>sos.examples.def.procedure</td>
<td>The procedure id for the requests.</td>
</tr>
<tr>
<td>sos.examples.def.featureOfInterest</td>
<td>The featureOfInterest id for the requests.</td>
</tr>
<tr>
<td>sos.examples.def.generalizer</td>
<td>The generalizer id for the requests.</td>
</tr>
<tr>
<td>sos.examples.def.time.min</td>
<td>The begin time for temporal filter.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>sos.examples.def.time.max</td>
<td>The end time for temporal filter</td>
</tr>
<tr>
<td>sos.examples.def.googleRequest.time.min</td>
<td>The begin time for the temporal filter in the 'Station in Google-Maps via KML-Request' link in the testClient.</td>
</tr>
<tr>
<td></td>
<td>!!! It is required to define a time stamp without time zone/offset !!!</td>
</tr>
<tr>
<td>sos.examples.def.googleRequest.time.max</td>
<td>The end time for the temporal filter in the 'Station in Google-Maps via KML-Request' link in the testClient.</td>
</tr>
<tr>
<td></td>
<td>!!! It is required to define a time stamp without time zone/offset !!!</td>
</tr>
<tr>
<td>sos.examples.def.srs</td>
<td>The EPSG number for spatial filter request.</td>
</tr>
<tr>
<td>sos.examples.def.bbox.lower</td>
<td>Lower corner of BBOX in spatial filter requests (GetObservation, GetFeatureOfInterest)</td>
</tr>
<tr>
<td>sos.examples.def.bbox.upper</td>
<td>Upper corner of BBOX in spatial filter requests (GetObservation, GetFeatureOfInterest)</td>
</tr>
<tr>
<td><strong>If the default properties contains special sign like</strong> \ / : &quot; * ? &lt; &gt; ; # % = @ replace these characters with a '_' in the following properties.</td>
<td></td>
</tr>
<tr>
<td>sos.examples.def.direktLink.offering</td>
<td>The offering id for the direct diagram link.</td>
</tr>
<tr>
<td>sos.examples.def.direktLink.procedure</td>
<td>The observableProperty id for the direct diagram link.</td>
</tr>
<tr>
<td>sos.examples.def.direktLink.observableProperty</td>
<td>The procedure id for the direct diagram link.</td>
</tr>
<tr>
<td>sos.examples.def.direktLink.featureOfInterest</td>
<td>The featureOfInterest id for the direct diagram link.</td>
</tr>
</tbody>
</table>

5.3 References