

Open Geospatial Consortium, Inc.

Date: 2011-22-03

Reference number of this document: OGC 10-194r3

Version: 1.0.0

Category: OGC™ Interoperability Experiment Report

Editors: Boyan Brodaric, Nate Booth

OGC® Groundwater Interoperability Experiment FINAL REPORT

Copyright © 2010 Open Geospatial Consortium, Inc. All Rights Reserved.
To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is not an OGC Standard. This document presents a discussion of technology issues considered in an initiative of the OGC Interoperability Program. This document does not represent an official position of the OGC. It is subject to change without notice and may not be referred to as an OGC Standard. However, the discussions in this document could very well lead to the definition of an OGC Standard. Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type:	OpenGIS® Engineering Report
Document subtype:	NA
Document stage:	Not approved for public release
Document language:	English

Preface

This report describes the methods, results, issues and recommendations generated by the Groundwater Interoperability Experiment (GWIE). As an activity of the OGC Hydrology Domain Working Group (HDWG), the GWIE is designed to: (1) test the use of WaterML2 with the SOS interface, and Groundwater ML (GWML) with the WFS interface, (2) test compatibility with software clients, and (3) facilitate sharing of massive volumes of sensor-based water level observations and related water well features across the Canada and United States border.

Contents	Page
1 Introduction.....	1
1.1 Scope	1
1.2 Document contributor contact points	2
1.3 Revision history.....	2
1.4 Future work	2
1.5 Forward	3
2 References.....	4
3 Terms and definitions	6
4 Conventions	7
4.1 Abbreviated terms	7
4.2 UML notation	8
5 GWIE Overview	9
5.1 Background	9
5.2 Motivation and Goals	9
5.3 User Scenarios.....	9
5.4 Use Cases	10
6 GWIE Architecture	11
6.1 Motivation and Goals	11
6.2 Design and Implementation.....	11
6.3 Architectural Issues and Recommendations.....	12
7 GWIE Best Practices and Outstanding Issues	14
7.1 GWIE Services Profile	14
7.2 Issues and Recommendations.....	19
7.3 GWIE WaterML2 Profile.....	31
7.4 GWIE GWML Profile	35
8 GWIE Client Implementation	36
8.1 Motivation and Goals	36
8.2 GSC Client	37
8.3 University of Calgary Client	38
8.4 34 North Client	40
8.5 NCSA GWIE Client and Service	42
9 Acknowledgements.....	44

Figures	Page
Figure 1: GWIE architecture.	11
Figure 2: WaterML2 schema used by the GWIE.	31
Figure 3: GWML water well model fragment.....	35
Figure 4: display of time series observation and water well feature rock log.	37
Figure 5: using Sensor web browser to browse sensors	38
Figure 6: displaying water well info.....	39
Figure 7: sensor meta data via SOS DescribeSensor interface.....	39
Figure 8: data export to CSV format	40
Figure 9: NRC WMS Layer demonstrating the density of the Canadian GW sites.	40
Figure 10: USGS service for water well features throughout the State of Michigan.	41
Figure 11: single observation query for Ontario region.	41
Figure 12: NCSA web client for browsing groundwater levels.	43

1 INTRODUCTION

1.1 Scope

This report describes the methods, results, issues and recommendations generated by the Groundwater Interoperability Experiment (GWIE). As an activity of the OGC Hydrology Domain Working Group (HDWG), the GWIE is designed to: (1) test the use of WaterML2 with the SOS interface, and Groundwater ML (GWML) with the WFS interface, (2) test compatibility with software clients, and (3) facilitate sharing of massive volumes of sensor-based water level observations and related water well features across the Canada and United States border.

The GWIE concludes that delivery of groundwater feature data via WFS and GWML, and delivery of groundwater observation data via SOS and WaterML2, is adequate for schematic interoperability across the United States and Canadian border, once best practices and project-specific profiles are deployed. To this end, the GWIE makes 27 profile recommendations, identifies 10 standards-related issues, and makes 7 recommendations for OGC standards modification. It also concludes that deployment of the services by web clients enables reasonable viewing and downloading of feature and observation data. However, querying groundwater observations, as time series, via OGC services is still a challenging task requiring further development of standards.

1.2 Document contributor contact points

All questions regarding this document should be directed to the editors or contributors:

Name	Organization
Boyan Brodaric (Ed.)	Geological Survey of Canada
Nate Booth (Ed.)	US Geological Survey
Eric Boisvert	Geological Survey of Canada
Thomas Kunicki	US Geological Survey
I-Lin Kuo	US Geological Survey
Ilya Zaslavsky	San Diego Supercomputer Center, UC-San Diego
David Valentine	San Diego Supercomputer Center, UC-San Diego
Peter Taylor	AU Commonwealth Scientific and Research Organization
Phil Stefanoff	Kisters
Stefan Fuest	Kisters
Amye Rita Osti	34North
Nathan Hemenway	34North
Yong Liu	National Center for Supercomputing Applications, University of Illinois
Steve Liang	University of Calgary

1.3 Revision history

Date	Release	Editor	Primary clauses modified	Description
16-Nov-10	0.1.0	BB, NB, EB	All	First draft
30-Nov-10	0.2.0	BB, NB, EB	All	Second draft at Dec10 TC meeting
14-Feb-11	0.3.0	BB, NB, EB	All	Interim final draft
18-Mar-11	1.0.0	BB, NB	All	Final draft

1.4 Future work

Future improvements to this document are desirable to clarify technical details arising from subsequent implementation of OGC standards and related technologies.

Aspects of the GWIE will continue informally amongst participants interested in maintaining and expanding groundwater data sharing between Canada and United States. Technical aspects requiring further investigation will be forwarded for consideration to the Surface Water Interoperability Experiment of the OGC Hydrology Domain Working Group.

1.5 Forward

This is an information document that describes the methods, issues, solutions, and recommendations identified by the GWIE.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

2 REFERENCES

The following documents are referenced in this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

OGC 01-068r3, Web Map Service Implementation Specification, 1.1.1, 2002-01-16, <http://www.opengeospatial.org/standards/wms>

OGC 04-094, Web Feature Service Implementation Specification, 1.1.0, 2005-05-03, <http://www.opengeospatial.org/standards/wfs>.

OGC 04-095, OpenGIS Filter Encoding Implementation Specification, 1.1.0, 2005-05-03, http://portal.opengeospatial.org/files/?artifact_id=8340. In addition to this document, this report includes several XML Schema Document files as specified in Annex A.

OGC 06-009r6, OpenGIS Sensor Observation Service, 1.0, 2007-10-26, <http://www.opengeospatial.org/standards/sos>.

OGC 06-042, OpenGIS Web Map Service (WMS) Implementation Specification, 1.3.0, 2006-03-15, <http://www.opengeospatial.org/standards/wms>.

OGC 06-121r3 OGC Web Services Common Specification, http://portal.opengeospatial.org/files/?artifact_id=20040.

OGC® 07-000, OpenGIS® Sensor Model Language (SensorML) Implementation Specification, 1.0.0, 2007-07-17, http://portal.opengeospatial.org/files/index.php?artifact_id=21273&passcode=fxphjb8qrc a4gwy7g626.

OGC 07-022r1, Observations and Measurements – Part 1 - Observation schema 1.0, 2007-12-08 http://portal.opengeospatial.org/files/?artifact_id=22466.

OGC 07-036 OpenGIS Geography Markup Language (GML) Encoding Standard, 3.2.1, 2007-08-27, http://portal.opengeospatial.org/files/?artifact_id=20509.

OGC 09-001, OpenGIS SWE Service Model Implementation Standard, 2009-09-30.

OGC 09-025r1 and ISO/DIS 19142, OpenGIS Web Feature Service 2.0 Interface Standard, 2010-11-02, <http://www.opengeospatial.org/standards/wfs>

OGC 09-026r1 and ISO/DIS 19143, OpenGIS Filter Encoding 2.0 Encoding Standard, 2010-11-22, <http://www.opengeospatial.org/standards/filter>

OGC 10-037, OGC SOS 2 Interface Standard, 2010-09-02, <http://www.opengeospatial.org/standards/sos>

OGC 10-037, OGC SOS 2 Interface Standard, 2010-09-02, <http://www.opengeospatial.org/standards/sos>.

OGC 10-126r1, OGC® WaterML2.0: An O&M profile for water observations data, 2010-05-27.

OGC 10-004r2 and ISO/DIS 19156, Geographic information - Observations and measurements, 2010-05-03.

OGC 10-025r1, Observations and Measurements - XML Implementation, 2010-11-05.

3 TERMS AND DEFINITIONS

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Specification [OGC 06-121r3], clause 4 of Sensor Observation Service [OGC 06-009r6], and Clause 4 of Observations and Measurements – Part 1 [OGC 07-022r1].

Mediator – a component to perform schema and/or semantic mediation, often assuming service integration responsibilities as well.

GIN Mediator – GSC’s Groundwater Information Network mediator, repurposed for use in the GWIE.

Schema mediation – a process of transforming the structure of schemas, often between public and private models.

Semantic mediation – a process of transforming schema content, often with intermediate transformations also required (such as language or units).

4 CONVENTIONS

4.1 Abbreviated terms

API	Application Programming Interface
CSIRO	Australian Commonwealth Scientific and Research Organization
CUAHSI	Consortium of Universities for the Advancement of Hydrologic Science
GSC	Geological Survey of Canada
GML	Geography Markup Language
GWIE	OGC Groundwater Interoperability Experiment
GWML	Groundwater Markup Language
HDWG	OGC Hydrology Domain Working Group
ISO	International Organization for Standardization
KML	Keyhole Markup Language
NCSA	National Center for Supercomputing Applications, U Illinois Urbana
NetCDF	network Common Data Form
OGC	Open Geospatial Consortium
OWS	OGC Web Services
O&M	Observations and Measurements
SensorML	Sensor Model Language
SOS	Sensor Observation Service
SWE	Sensor Web Enablement
SWG	Standards Working Group
UML	Unified Modeling Language
USGS	US Geological Survey
WaterML2	Water Markup Language
WMC	Web Mapping Context
WMS	Web Mapping Service
WFS	Web Feature Service
XML	eXtensible Markup Language
XSD	XML Schema Definition

4.2 UML notation

Class diagrams that appear in this report are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 06-121r3].

5 GWIE OVERVIEW

5.1 Background

Members of the OGC/WMO Hydrology Domain Working Group (HDWG) are developing WaterML2. WaterML2 is a data transfer format used in conjunction with various OGC web service standards for the transmission of water resources time series data. Development of WaterML2 involves the harmonization of several existing representations such as WaterML (US), Observations & Measurements (O&M—OGC), and GroundwaterML (GWML—Can).

5.2 Motivation and Goals

This interoperability experiment advances the development of WaterML2, tests its use in conjunction with various OGC service standards (SOS, WFS, WMS and CSW) as well as emerging web clients, and investigates its interaction with GroundwaterML (GWML). This occurs within the context of sharing groundwater level data between the USA and Canada.

5.3 User Scenarios

The GWIE considers four user scenarios that are functionally carried out using a web client: (1) finding observations or features, (2) querying one or more observations or features, (3) displaying the query results, and (4) downloading the query results.

- **Discovery scenario:** includes finding one or more features, i.e. water wells, or observations, i.e. time series. This might be accomplished in a web client by browsing a map that displays the locations of features or observations, or browsing a catalog showing feature or catalog ids or metadata. For the purposes of the GWIE, discovery includes all operations required to obtain ids or locations, of desired features or observations.
- **Query scenario:** includes simple requests to retrieve the characteristics of one observation or feature, via an id or location. Also includes complex requests to retrieve multiple observations and features using thematic, temporal, and spatial conditions.
- **Display scenario:** includes the on-screen display of query results in graphs, tables, or raw GWML or WaterML2.
- **Download scenario:** includes the storage of query results to a local file in a variety of formats, such as GWML, KML, Excel® and WaterML2.

5.4 Use Cases

The user scenarios described above imply deployment of specific OGC standards and technologies:

- **Discovery use-case:** Service providers make groundwater well locations available as a map via WMS.
- **Query use-case:** Users query for groundwater well characteristics or water levels by invoking WFS or SOS, respectively. Service providers make well characteristics available in GWML via WFS. Clients call WFS based on a featureId or bounding box (BBOX). Service providers make water levels available in WaterML2 via SOS. Clients call SOS using a featureId or envelope (BBOX). The WMS GetFeatureInfo is also converted into a WFS GetFeature, or SOS GetObservation.
- **Display use-case:** Once well characteristics and levels are queried, a server renders the WFS or SOS responses into various user-friendly formats.

For well characteristics returned from WFS, aquifer, lithology, and screened intervals are shown in tabular and graphic formats. The well characteristics graph shows the vertical aspect of the well using various colors for the different lithologies and screened intervals.

For water levels returned from SOS, tables and time-series graphs help users visualize changes in water levels over time.

- **Download use-case:** Web clients allow users to download data from multiple wells, from multiple service providers, into a unified file. Based on a BBOX, a set of wells is selected, and downloads of well characteristics or water levels are available. The download option should normalize differing vocabularies for well characteristics, units for water levels, and maintain provenance for each service provider.

5.4.1 Contributors

Role	Contributor
Service provider	GSC, USGS, NCSA
Client developer	GSC, USGS, 34 North, University of Calgary, Kisters
WaterML2 developer	CSIRO, CUAHSI, GSC, USGS

6 GWIE ARCHITECTURE

6.1 Motivation and Goals

The architecture of the GWIE strives to optimize usability for clients, maximize interoperability, and re-use existing infrastructure. Important guiding principles include:

- one web-services access point for clients: this eliminates heterogeneity in service delivery that is often present in distributed systems;
- a single integrated result returned to clients: this frees clients from integrating query responses potentially from multiple clients; otherwise clients would need to deal with multiple simultaneous and disjoint streams of data;
- semantic and schematic interoperability: this eliminates the need for schematic and semantic translation to be carried out by the data provider or web client.

6.2 Design and Implementation

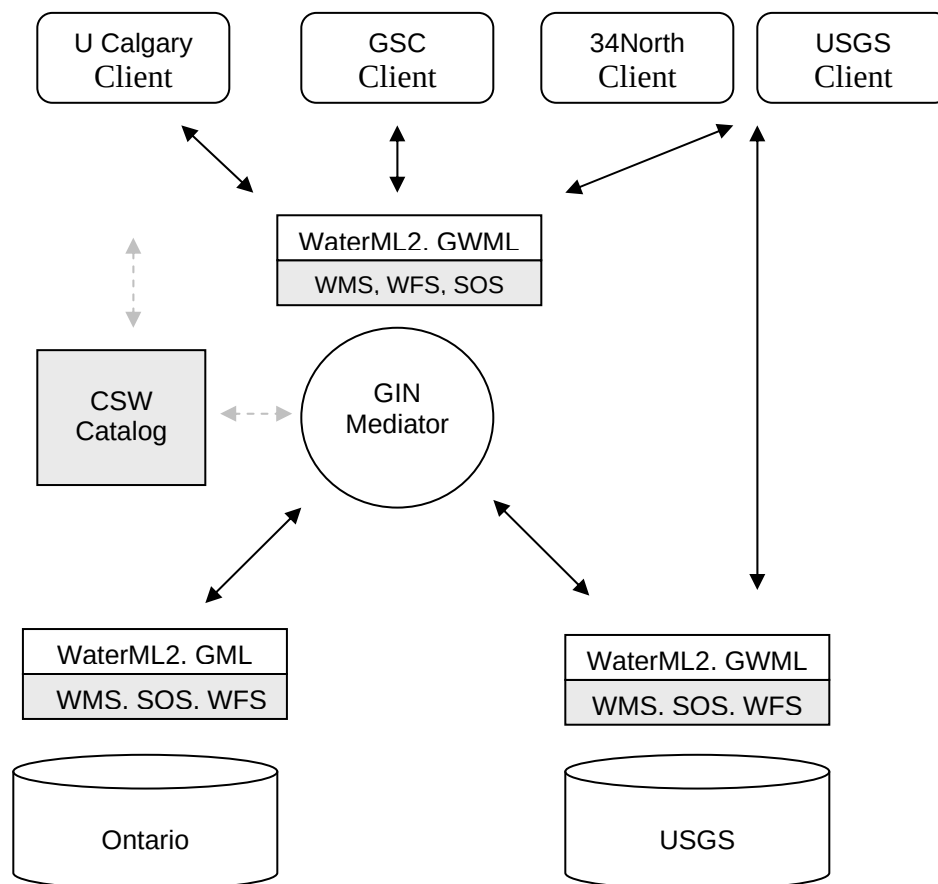


Figure 1: GWIE architecture.

The GWIE adopted a mediated architecture as shown in Figure 1. In this architecture a central mediator brokers requests by clients as well as responses by data providers: the mediator carries out query distribution, semantic and schematic translation, and response integration. Access is provided by OGC WMS, WFS, SOS interfaces, which wrap feature and observation data served by the data providers. A CSW catalog potentially provides discovery services for: the data servers, the mediator services, and for metadata about each sensor (e.g. time and value of last reading) to facilitate execution of queries across data providers.

The GWIE implementation consists of two data providers (USGS, GSC), four clients (U Calgary, GSC, 34North, USGS), and an existing mediator (GIN Mediator) adapted from GSC work. This design is implemented in a hybrid manner: not all clients used the mediator exclusively, as 34North and USGS clients accessed USGS services directly, and the public catalog services (CSW) were not implemented, due to timing and resource constraints.

Implementation of the architecture requires a series of agreements between participants, or a "profile", to constrain service behavior in cases where the specification is inadequate. The GWIE profile is described in section 7.

6.3 Architectural Issues and Recommendations

Implementation of the GWIE architecture identified two key issues. The first is a gap in service discovery, and the second is the absence of a standard way to link various OGC service components.

ISSUE 1. Sensor observations metadata

The OGC GetCapabilities response does not provide sufficient information to enable discovery of relevant sensors in a large collection of sensors / features-of-interest. One approach suggests this information should be provided by a catalog, however, this would require atypical catalog functions which go beyond managing the location of services to include the handling of:

- Metadata about properties such as which sensors measure water levels;
- Metadata about sensor data: period of record, most recent observations, etc.

ISSUE 2. Relationships between services, e.g. between WMS and WFS and SOS

The GWIE requires several OGC services to work together in a coherent way: the client application shows a network of sensors / features-of-interest as WMS layers that need to interact with WFS and SOS services, and SOS services also need to bind a WFS service to a feature-of-interest. The current implementation uses various ways to achieve this and we feel that a standard way to convey the interconnections would be beneficial. Web Mapping Context (OGC 05-005) partially deals with the problem by packaging the initial stack of relevant WMS layers. However, WMC does not provide instructions about how to link to features or observations from a map, inasmuch the "DataURL" is a simple HTTP URI link with no constraints.

- There is a need to adapt the WMC approach by packaging into a single file several aspects of the application. For example linking a WMS layer with a WFS and SOS: at present there is no standard method by which a client can know which WFS or SOS serves data associated with a specific WMS layer. This is particularly relevant to “identify” requests in which a feature is selected on-screen by a client, and where the client issues a `getFeatureInfo` request to a WFS, or a `getObservation` request to a SOS, as carried out in the GWIE.

7 GWIE BEST PRACTICES AND OUTSTANDING ISSUES

This section documents mandatory and optional practices for service delivery in the GWIE—i.e. it describes the “GWIE Services Profile”. It also describes outstanding problems in existing OGC specifications, as well as proposed solutions. The solutions will be submitted to the appropriate OGC groups for consideration.

7.1 GWIE Services Profile

SOS1. GWIE-compliant SOS services SHALL use the SOS 1.0 specification, even if the resulting Observation collection is GML 3.2.

By default, the GWIE SOS services will respond with GML 3.2.1, so service support for GML 3.2.1 responses is mandatory. GWIE SOS services may in addition also optionally respond in GML 3.1.1 via the responseFormat parameter.

For example,

```
<sos:responseFormat>text/xml;subtype="om/1.0.0"</sos:responseFormat>
```

must be used to request GML 3.1.1 (OM 1.0.0 is based on GML 3.1.1) while

```
<sos:responseFormat>text/xml;subtype="om/2.0.0"</sos:responseFormat>
```

can be used to explicitly request GML 3.2 (OM 2.0 is based on GML 3.2), but this is unnecessary since GML 3.2 is the default.

This will also apply to the use of filter encoding in the SOS service request. The filter encoding specification uses GML 3.1.1 and the domain feature must be expressed in 3.1.1 even if the generated response is in 3.2.x.

SOS2. GWIE-compliant SOS service SHALL provide both gml:Coverage and sweCommon encoding for WaterML2 TimeSeries.

The SOS service shall encode a time series (as defined in the WaterML2 UML) either as a wml2:TimeSeries (the default), which is a type of gml:Coverage, or a SWE block. The responseFormat parameter shall be used to explicitly request encoding:

- GML 3.1 using coverage:

```
<sos:responseFormat>text/xml;subtype="om/1.0.0"</sos:responseFormat>
```

- GML 3.1 using swe block encoding:

```
<sos:responseFormat>text/xml;subtype="om/1.0.0";enc=block</sos:responseFormat>
```

- GML 3.2 using coverage:

```
<sos:responseFormat>text/xml;subtype="om/2.0.0"</sos:responseFormat>
```

- GML 3.2 using swe block encoding:

```
<sos:responseFormat>text/xml;subtype="om/2.0.0";enc=block</sos:responseFormat>
```

SOS3. GWIE-compliant SOS SHALL support resultType=inline by default.

The current GWIE architecture is restricted to inline results.

The GWIE recognizes that working with large datasets is an outstanding issue. Several options were considered to deal with this issue, but none were implemented. The options fall into two complementary categories, the first through optimizing the transaction by preventing unnecessary transfer of large datasets, and the second by compressing the payload. This document proposes some options of the first type (See Issues 3, 4, 6 and 7).

Additional variants of the second category include:

- Serialize results offline and point to compact representations of large datasets (for example NetCDF).
- Use of compressed XML formats and APIs such as Fast Infoset (so the whole response would be compressed).

SOS4. The feature-of-interest SHALL be by reference in a SOS response.

Any reference to a feature-of-interest SHALL be expressed as a fully resolvable URL in the xlink:href attribute. This overcomes the problem of having features-of-interest expressed variously in GML.

SOS5. Large collections of feature-of-interest in a [GetCapabilities](#) Offering SHALL be advertised as a nested FeatureCollection.

Instead of a long list of features-of-interest, the GetCapabilities offering section should report a reference to a nested FeatureCollection. The reference itself as well as references within the nested collection SHALL conform to SOS4.

```
<sos:featureOfInterest xlink:href="http://ngwd-bdnes.cits.nrcan.gc.ca:8080/Reference/uri-cgi/feature/Ontario/WaterWellNetwork" xlink:title="Ontario Water well network"/>
```

If the number of features-of-interest is large, the reference shall point to a gml:FeatureCollection. The referenced document shall be a gml:FeatureCollection with mandatory child elements: gml:description, gml:name, and gml:boundedBy containing gml:EnvelopeWithTimePeriod with gml:lowerCorner, gml:upperCorner, and gml:beginPosition and gml:endPosition. When the envelope has only one temporal element, it must be repeated in both beginPosition and endPosition. For an example see ISSUE 3 below. The data access method is analogous to tiling in WMS.

Collections SHALL not overlap (ie, the same feature shall not appear in two different collections). This impacts client behavior: when SOS4 is applied, the client should parse the feature collection to discover the features-of-interest when a feature id is required by related services. The client will inspect the feature-of-interest in case it is a collection before using the reported id in a GetFeatureOfInterest. Note that because of SOS4, the reference to the feature-of-interest is already explicit and does not need a specific operation to be extracted.

Therefore a GWIE compliant client should use the explicit reference instead of using GetFeatureOfInterest. This architecture actually questions the necessity of having a GetFeatureOnInterest request since items can be accessed directly from the observation (by resolving the om:featureOfInterest/xlink:href).

SOS6. An EPSG code shall follow the URN pattern `urn:ogc:def:crs:EPSG:<code>`.

SOS7. The axis order for EPSG 4326 SHALL be x,y when the response is serialized in GML 3.1, but y,x when serialized in versions above 3.1.x.

As specified in <http://www.ogcnetwork.net/node/491> (08-038r5). This requirement has been suggested in order to progressively transition from x,y to y,x without breaking legacy applications that rely on the older axis order. It is also needed to enable client applications to be aware of both possibilities and to determine which one is used.

This is consistent with existing GML documentation where GML 3.1.1 shows the "wrong" (x,y) order and GML 3.2.x shows the "correct" (y,x) order.

We can explicitly express the order of axes by using the following CRS ids:

- urn:ogc:def:crs:epsg::4326 for "old" axes order (x,y)
- urn:ogc:def:crs:epsg:6.5:4326 for new axes order

SOS8. A GWIE-compliant SOS service SHALL support specified spatial and temporal filters.

The service SHALL support this short list of filter operations:

Supported temporal filters

- TM_During
- TM_Before
- TM_After

Supported spatial filters

- BBOX

SOS9. A GWIE-compliant SOS service SHALL ignore the property name in a spatial request and use whatever makes sense for the location of an Observation.

A spatial request, where a series of observations can be filtered using a spatial operator (most commonly a BBOX), should, according to the spec, express a ogc:Filter where the spatial property is explicit. This profile will use whichever spatial property makes sense for the data provider. For consistency, ISSUE5 proposes a formal way to convey this behavior in SOS.

This example uses the property name proposed in ISSUE5:

```
<sos:GetObservation xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:sos="http://www.opengis.net/sos/1.0"
xmlns:om="http://www.opengis.net/om/1.0"
xsi:schemaLocation="http://www.opengis.net/sos/1.0
http://schemas.opengis.net/sos/1.0.0/sosAll.xsd"
service="SOS" version="1.0.0" srsName="EPSG:4326">
...
  <sos:featureOfInterest>
    <ogc:BBOX>

<ogc:PropertyName>http://www.opengis.net/req/omxml/2.0/data/samplingGeometry</ogc:
PropertyName>
      <gml:Envelope>
        <gml:lowerCorner>38.11 -78.6</gml:lowerCorner>
        <gml:upperCorner>38.14 -78.4</gml:upperCorner>
      </gml:Envelope>
    </ogc:BBOX>
  </sos:featureOfInterest>
...
</sos:GetObservation>
```

SOS10. GetObservationResponse shall use gml:FeatureCollection as response container, but in the interim om:ObservationCollection is acceptable.

A valid SOS 1 response cannot be structured because the default response must validate with WaterML2, which uses GML 3.2 (see SOS1 above). At the time of the conclusion of the GWIE, we were not able to determine a satisfactory solution and therefore wrapped the response using (non valid) om:ObservationCollection.

SOS11. GetObservation SHALL report spatial location using om:parameter.

As a convenience for applications that want to plot the location of the observation without having to resolve the feature-of-interest (including all the related schema and performance implications), all Observations SHALL comply with sub –clause 7.13 of OGC 10-025 (Cox, 2010b), where the sampling geometry is serialized with an Observation parameter:

```
<?xml version="1.0" encoding="UTF-8"?>
<om:OM_Observation>
...
  <om:parameter>
    <om:NamedValue>
      <om:name
xlink:href="http://www.opengis.net/req/omxml/2.0/data/samplingGeometry"/>
      <om:value>
        <gml:Point gml:id="SamplingPoint">
          <gml:pos srsName="urn:ogc:def:crs:EPSG:4326">52.9 7.52</gml:pos>
        </gml:Point>
      </om:value>
    </om:NamedValue>
  </om:parameter>
</om:OM_Observation>
```

The complete example can be found in SOS 2.0 10-037.

SOS12. GWIE compliant SOS service shall implement GetObservationById.

See clause 10 of OGC 06-009r6. GetObservationById shall also report location following SOS11

SOS13. GWIE-compliant SOS results SHALL be encoded as time series.

Even when only a single value is returned, it shall be encoded as a time series with one record.

WFS1. GWIE WFS services shall use WFS 1.1.0.

WFS2. GWIE WFS shall provide water well information using gwml:WaterWell.

WFS3. External references SHALL use HTTP URI and be fully resolvable.

External references (ie, `xlink:href`) must be expressed using a HTTP URI as defined in Cox 2010a and suggested in http://www.cabinetoffice.gov.uk/media/308995/public_sector_uri.pdf.

During resolution, the correct mime type should be returned. For example, a WaterML2 Observation pointing to a feature-of-interest would use

```
<om:featureOfInterest xlink:href="http://ngwd-bdnes.cits.nrcan.gc.ca:8080/Reference/uri-cgi/feature/gsc/waterwell/ca.on.gov.wells.1100001" xlink:title="Water well 1100001"/>
```

It is strongly recommended that `xlink:title` be used along with `xlink:href` as an aid to client applications.

7.1.1 Optional GWIE Profile Elements

OPT1. It is strongly recommended that a GWIE-compliant WFS return gwml:logElement with well construction information within gwml:WaterWell .

OPT2. It is recommended that a GWIE-compliant WFS and SOS implement a (nonstandard) form using the parameter "xml" and HTTP/POST.

For an HTML-based client, posting via a form is easier than posting with XML in the HTTPBody. This is especially useful when providing example invocations of the service. The enctype shall be "application/x-www-form-urlencoded"

```
<form action="../../../GetObservation" method="post" enctype="application/x-www-form-urlencoded" name="sos">
<textarea rows="15" cols="75" name="request">
<lt;sos:GetObservation
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:sos="http://www.opengis.net/sos/1.0"
  xmlns:om="http://www.opengis.net/om/1.0"
```

```

xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:sa="http://www.opengis.net/sampling/1.0"
xsi:schemaLocation="http://www.opengis.net/sos/1.0
http://schemas.opengis.net/sos/1.0.0/sosAll.xsd"
service="SOS" version="1.0.0" srsName="EPSG:4326">
  <sos:offering>urn:GIN:offering:groundwaterLevel:1</sos:offering>
  <sos:observedProperty>urn:ogc:def:property:OGC:GroundWaterLevel</sos:obs
ervedProperty>
  <sos:featureOfInterest>
    <ogc:BoundingBox>
      <ogc:PropertyName>sa:position</ogc:PropertyName>
      <gml:Envelope>
        <gml:lowerCorner>-79 44</gml:lowerCorner>
        <gml:upperCorner>-78 45</gml:upperCorner>
      </gml:Envelope>
    </ogc:BoundingBox>
  </sos:featureOfInterest>
  <sos:responseFormat>text/xml;
subtype="om/1.0.0" </sos:responseFormat>
  <sos:resultModel
xmlns:wml="http://www.wron.net.au/waterml2">wml:WaterMonitoringObservation</
sos:resultModel>
  <sos:responseMode>inline</sos:responseMode>
  <sos:GetObservation>
</textarea>
<input type="submit"/>
</form>

```

OPT3. It is strongly recommended that a GWIE-compliant SOS implement comparison operations under the result parameter.

Although the GWIE did not implement this use case, we recommend that filter on result shall be implemented and follow recommendations proposed in ISSUE 4

```

<?xml version="1.0" encoding="UTF-8"?>
<sos:GetObservation service="SOS" version="1.0.0"
xmlns:sos="http://www.opengis.net/sos/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"
xmlns:ogc="http://www.opengis.net/ogc">
  <sos:offering>OntarioWaterWell</sos:offering>
  <sos:observedProperty>WaterLevel</sos:observedProperty>
  <sos:result>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>water level</ogc:PropertyName>
      <ogc:Literal>12.4</ogc:Literal>
    </ogc:PropertyIsGreaterThan>
  </sos:result>
  <sos:responseFormat>text/xml;subtype="om/2.0.0"
  </sos:responseFormat>
</sos:GetObservation>

```

7.2 Issues and Recommendations

ISSUE 3. SOS GetCapabilities

The SOS 1 specification requires that the list of features-of-interest be explicitly serialized in the [GetCapabilities](#) document. This list is either used for discovery (harvesting by a catalog) or to provide a valid list of feature identifiers to be used in GetObservation (the feature-of-interest id being one of the allowed parameters

of this operation). In the current WaterML2 model, it has been decided that the feature-of-interest should be the monitoring station (or any other relevant sampling feature). But, some systems can contain large quantities of features-of-interest, so it is impractical for performance reasons to serialize them individually. Nonetheless, the service must still publish a collection of features-of-interest that can be used in GetObservation to extract observation related to a specific feature-of-interest.

Proposed Solution

We propose that the profile element SOS5 be formally part of the SOS specification. The capabilities document should be allowed to provide a composite feature as feature-of-interest. The composite feature shall be a `gml:FeatureCollection` that nests other `gml:FeatureCollections` and/or features-of-interest. The collection would be composed of a list of sub-collections. If the sub-collection contains a reasonable amount of features-of-interest, the list would be serialized explicitly; otherwise a reference to a collection shall be serialized. The nesting logic within collections is up to the provider. It can follow a purely geometric partitioning (e.g. quad-tree) or follow an administrative structure (e.g. state/county/city).

Proposed Specification Modification: SOS 2

Specify that the related feature CAN be a feature collection and if so, the client SHALL consider that the feature is actually within this collection, considering that the feature collection:

- can have nested feature collections,
- the feature-of-interest must report a `xlink:role` indicating that the reference is a collection or a single feature-of-interest, and
- those nested feature collections can be provided by reference. It's then up to the client application to resolve the references and drill down into the collection.

Note that this can also be used in SOS 1.0. It can be implemented as a specific profile. The referenced document shall be a `gml:FeatureCollection` with child elements:

- `gml:description` (optional),
- `gml:name` (mandatory),
- `gml:boundedBy` (mandatory)

The `boundedBy` property shall contain a `gml:EnvelopeWithTimePeriod` with all its mandatory properties (`gml:lowerCorner`, `gml:upperCorner`, `gml:beginPosition`, and `gml:endPosition`).

Below is an example of a reference to a large collection. The collection has 4 root `featureMembers` (nw, ne, sw, and se). The first and the last `featureMembers` are

external references to other collections. The second and third contain collections have an explicit list of features-of-interests.

```
<?xml version="1.0" encoding="UTF-8"?>
<gml:FeatureCollection xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xlink="http://www.w3.org/1999/xlink" gml:id="network.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml/3.2
http://schemas.opengis.net/gml/3.2.1/gml.xsd">
  <gml:boundedBy>
    <gml:EnvelopeWithTimePeriod srsName="urn:ogc:def:crs:EPSG:4326">
      <gml:lowerCorner>49.260000000000008 -95.19</gml:lowerCorner>
      <gml:upperCorner>56.92 -84.69</gml:upperCorner>
      <gml:beginPosition frame="urn:ogc:property:time:iso8601">1899-
05-16T00:00:00.0</gml:beginPosition>
      <gml:endPosition frame="urn:ogc:property:time:iso8601">2009-12-
30T00:00:00.0</gml:endPosition>
    </gml:EnvelopeWithTimePeriod>
  </gml:boundedBy>
  <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca/service/gin/sos2/network/OntarioNetwork?BBOX=-
95.19,49.260000000000008,-89.94,53.09" xlink:title="sw"
xlink:role="http://www.opengis.net/gml/FeatureCollection"/>
    <gml:featureMember>
      <gml:FeatureCollection gml:id="nw">
        <gml:boundedBy>
          <gml:EnvelopeWithTimePeriod
srsName="urn:ogc:def:crs:EPSG:4326">
            <gml:lowerCorner>54.59 -95.19</gml:lowerCorner>
            <gml:upperCorner>56.92 -89.94</gml:upperCorner>
            <gml:beginPosition
frame="urn:ogc:property:time:iso8601">1899-05-16T00:00:00.0</gml:beginPosition>
            <gml:endPosition
frame="urn:ogc:property:time:iso8601">2009-12-30T00:00:00.0</gml:endPosition>
          </gml:EnvelopeWithTimePeriod>
        </gml:boundedBy>
        <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca:8080/Reference/uri-
cgi/feature/gsc/waterwell/ca.on.gov.wells.3101985" xlink:title="3101985"
xlink:role="http://www.opengis.net/gml/Feature"/>
          <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca:8080/Reference/uri-
cgi/feature/gsc/waterwell/ca.on.gov.wells.3100506" xlink:title="3100506"
xlink:role="http://www.opengis.net/gml/Feature"/>
          <!-- ... -->
        </gml:FeatureCollection>
      </gml:featureMember>
    </gml:featureMember>
    <gml:featureMember>
      <gml:FeatureCollection gml:id="ne">
        <gml:boundedBy>
          <gml:EnvelopeWithTimePeriod
srsName="urn:ogc:def:crs:EPSG:4326">
            <gml:lowerCorner>54.59 -89.94</gml:lowerCorner>
            <gml:upperCorner>56.92 -84.69</gml:upperCorner>
            <gml:beginPosition
frame="urn:ogc:property:time:iso8601">1899-05-16T00:00:00.0</gml:beginPosition>
            <gml:endPosition
frame="urn:ogc:property:time:iso8601">2009-12-30T00:00:00.0</gml:endPosition>
          </gml:EnvelopeWithTimePeriod>
        </gml:boundedBy>
        <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca:8080/Reference/uri-
cgi/feature/gsc/waterwell/ca.on.gov.wells.3100636" xlink:title="3100636"
xlink:role="http://www.opengis.net/gml/Feature"/>
          <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca:8080/Reference/uri-
cgi/feature/gsc/waterwell/ca.on.gov.wells.3100638" xlink:title="3100638"
xlink:role="http://www.opengis.net/gml/Feature"/>
        </gml:FeatureCollection>
      </gml:featureMember>
    </gml:featureMember>
  </gml:FeatureCollection>
</gml:FeatureCollection>
```

```

        <gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca:8080/Reference/uri-
cgi/feature/gsc/waterwell/ca.on.gov.wells.3100640" xlink:title="3100640"
xlink:role="http://www.opengis.net/gml/Feature"/>
        <!-- ... -->
    </gml:FeatureCollection>
</gml:featureMember>
<gml:featureMember xlink:href="http://ngwd-
bdnes.cits.nrcan.gc.ca/service/gin/sos2/network/OntarioNetwork?BBBOX=-
89.94,49.2600000000000008,-84.69,53.09" xlink:title="se"
xlink:role="http://www.opengis.net/gml/FeatureCollection"/>
</gml:FeatureCollection>

```

OGC Comment Status

Comment #24 of OGC 10-208.

Deferred to future work on the "SOS Large Dataset Extension".

ISSUE 4. Complex Thematic Queries in SOS

Filtering on the result requires the client to have some mechanism to discover the properties that can be filtered. A use case addressed by the GWIE is a request for all observations that have a water level above or below some threshold. This requires the filter expression to be written against the content of the result element (which is a time series containing pairs of water levels and time stamps).

SOS 1 provides two operations that allow a client to inspect the structure of a response: DescribeObservationType and DescribeResultModel (OGC 06-009r6 sections 10.6 and 10.7). The second operation would normally provide the response as a W3C XSD schema. The XSD schema of the expected result is not sufficient for several reasons:

- The SOS service can provide the result using multiple SWE encodings. Since the filtering encoding uses ogc:Filter, it assumes that the target property can be expressed using XPath. A result of using SWE encoding is that it does not express the structure in XSD, but uses a SWE meta schema. This meta-schema is not expressed in the schema, but rather in the instance of a dataset.
- The SOS service can serialize the result using many encodings (XML or Block or gml:Coverage or any other GML domain model) for the same observed property. The current specification does not provide a way to specify to the client which "schema" it should use to express the filter. We feel that the user should not have to adapt the way he expresses the filter based on the serialization of the output.
- Even if the result is expressed in explicit GML XSD (as opposed to SWE), and even if the schema is actually known, the wide usage of polymorphism and substitutions in the result model can lead to multiple options.
- Observable properties are actually advertised as SWE Common which does not have a direct mapping to GML/XML elements. When parsing the SWE description of the observed property, we don't explicitly know

how this would materialize in the filter expression, i.e. which observation element matches a GML element.

While it is possible to extract the property definition in SWE and even get the details about the units of measure used by the service, it is not possible to craft an XPath representation of the structure of this property, simply because there are no systematic mappings between a SWE structure and a GML structure.

Proposed Solution

We therefore propose that result queries SHALL be written against the SWE meta schema instead of GML XSD schema. It is possible in both SOS 1 and SOS 2 to discover the properties that are serialized (in one way or another) in the result and write a filter that unambiguously expresses the filtering criteria to the service. The client SHALL use the identifier (the name of the field in the swe:DataRecord) of the property as a proxy for the filter property, regardless of the result serialization.

The service should have an internal mechanism to do what "makes sense" in this particular case. It should also handle the cases when the structure of the value can be complex (range instead of single value) and remove the complexity of writing a filter for the user. This pattern has several benefits:

- It provides a way to document and advertise properties that are part of the result set at the phenomenon level.
- Properties are advertised using a conformant OGC approach.
- Properties are well described (SWE provides a certain level of metadata that XSD does not).
- The query structure is decoupled from the result encoding, which is consistent with a CSW pattern and a proposed "stored query" mechanism in WFS 2.0 (ISO-19142). The client can rely on a consistent way to get the parameters, regardless of the response encoding.
- This mechanism can be used by both SOS 1.0 and SOS 2.0 clients. The difference being only in the steps required to access the portion of SWE required to build the query.

The GWIE members recognize that more work needs to be done to achieve a coherent pattern.

A third option would explicitly introduce a list of "queryable" properties in the Offering.

```
<sos:ResultParameters>
  <sos:Parameter identification="urn:x-ogc:parameter:depth"
    uom="urn:ogc:def:uom:UCUM:m" type="xs:double">
    <sos:name xml:lang="fr">Profondeur de la nappe</sos:name>
    <sos:name xml:lang="eng">Aquifer depth</sos:name>
  </sos:Parameter>
  <sos:Parameter
    identification="urn:ogc:parameter:time" uom="urn:ogc:property:time:iso8601"
    type="xs:DateTime">
    <sos:name xml:lang="fr">Date de la mesure</sos:name>
```

```

        <sos:name xml:lang="eng">Measurement date</sos:name>
      </sos:Parameter>
    </sos:ResultParameters>

```

Proposed Specification Modification: SOS 2

We therefore propose the following modifications to the SOS specification:

- A SOS service SHALL take SWE references as proxies for property names in a result filter expression.
- A SOS service SHALL handle numerical representation and perform the necessary adjustment internally for complex numbers. Therefore, if the result is made of numerical values that are a range, or have uncertainties, it should handle comparison with input parameters that are single values automatically.
- A SOS service COULD explicitly provide a list of queryable properties in the Offering section. This means that the offering tag SHALL be modified to include the result parameters and those parameter names shall match SWE property names.

```

<sos:observedProperty>
  <sos:name>urn:ogc:def:property:OGC:GroundWaterLevel</sos:name>
  <sos:ResultParameters>
    <sos:Parameter identification="urn:x-ogc:parameter:depth"
      uom="urn:ogc:def:uom:UCUM:m" type="xs:double">
      <sos:name xml:lang="fr">Profondeur de la nappe</sos:name>
      <sos:name xml:lang="eng">Aquifer depth</sos:name>
    </sos:Parameter>
    <sos:Parameter
      identification="urn:ogc:parameter:time" uom="
      urn:ogc:property:time:iso8601" type="xs:DateTime">
      <sos:name xml:lang="fr">Date de la mesure</sos:name>
      <sos:name xml:lang="eng">Measurement date</sos:name>
    </sos:Parameter>
  </sos:ResultParameters>
</sos:observedProperty>

```

OGC Comment Status

Comment #25 of OGC 10-028.

Deferred to future work by SOS 2.0 SWG.

GWIE invited to propose an extension draft.

ISSUE 5. Spatial Queries in SOS

This is similar to the previous issue, but without the benefit of having a SWE description of the spatial property. SOS 1 allows only spatial filters on the feature-of-interest parameter. A proper spatial filter requires a PropertyName to identify the geometry property which might not be readily or unambiguously extracted from the schema. Even if the SOS actually provides a schema location for the domain of interest, the following issues must be dealt with:

- The service might refer to a heterogeneous collection of feature-of-interest types, therefore not a single property but multiple spatial properties must be identified (forcing a complex “OR” expression).

- The service might refer to a feature type that is the head of a substitution group where the spatial property is implemented by derived types (SF_SamplingFeatureType is a good example), forcing an “OR” expression.
- The spatial property might not be unique for a feature type, forcing an “OR” expression or resulting in incoherent filter expressions.
- The spatial property might not be conveniently located at the root of the feature-of-interest type and could require a deep analysis of the schema to locate it. This risks finding more than one property, again forcing an “OR” or incoherent expressions.

Several implementations have chosen to ignore the property name and apply the spatial query in a way that makes sense for the service (similar to the WFS 1.0.0 KVP BBOX parameter that did not require explicit naming of the geometry property). One proposal is to systematically use the gml:boundedBy property, but this essentially forces all spatial operations to be expressed on an envelope.

SOS 2.0 (OGC 10-037, clause 12.1) proposes a Spatial Filtering Profile to expose a geometry as a parameter of the Observation. They therefore suggest that a unique identifier shall represent this parameter (OGC 10-025r1 sub clause 7.13 states that the identifier of this parameter SHALL be <http://www.opengis.net/req/omxml/2.0/data/samplingGeometry>). The client then uses the following as a property name:

```
om:parameter/om:NamedValue[om:name/@xlink:href='http://www.opengis.net/req/omxml/2.0/data/samplingGeometry']/om:value
```

Proposed Solution

We propose that the long XPath, which essentially is a constant for the spatial property, be replaced by the HTTP URI identifier. proposed in OGC 10-025r1: <http://www.opengis.net/req/omxml/2.0/data/samplingGeometry>. We feel this is a more generic way to represent a "well-known property" or a relevant samplingGeometry. The proposal is similar to ISSUE2 as it hides the schematic details and potential heterogeneity of the feature(s) of interest and does not bind the meaning to an encoding artifact of O&M. We also feel that this approach is more portable to other OGC service interfaces.

Proposed Specification Modification: SOS2

A SOS service SHALL consider "<http://www.opengis.net/req/omxml/2.0/data/samplingGeometry>" as a proxy to a sampling geometry property of the feature-of-interest in a spatial filter expression.

OGC Comment Status

Comment # 26 of OGC 10-028.

Accepted.

ISSUE 6. SOS resultType parameter with "metadata" and "results"

In dealing with large datasets, the SOS operations GetObservation, GetResult, GetObservationById, and GetFeatureOfInterest can return many results (for example, there are time series in USGS groundwater data with 30000+ measurements). Clients which naively make routine queries of large datasets can be overwhelmed by the size of the response document. We think there needs to be mechanisms that allow a client to query the SOS to get an estimate of the size of the dataset and to adopt a different workflow if the size is large.

Proposed Solution

The WFS specification (both 1.0 and 2.0) uses a resultType parameter with a value of "result" or "hits" in 09-025r1 section 7.6.3.6. We feel this is a good feature to add to SOS 2.0 to deal with large datasets, as it allows a client to get a count of the number of results returned before deciding whether to deal with the entire result set. We think we should further expand this functionality to add more result information, including the number of observations for each site. A smart client, using resultType (in concert with another proposal to add @maxObservations and @maxResults) can then decide to make a limited query.

We propose to include a "metadata" result type where the server would have the opportunity to report on several aspects of the result collection. The content of that metadata section is to be refined in future work.

As for the parameter name, however, we cannot directly import the resultType functionality directly into SOS 2.0 because the resultType parameter is already taken in SOS 2.0 for other purposes. Thus, we have two choices:

- Associate the WFS resultType functionality with a different parameter in SOS 2.0, or
- Move the current SOS 2.0 resultType to a different parameter to allow resultType to be used as in WFS.

We consider that it was a mistake for SOS to reuse a pre-existing parameter name and assign it a different behavior. Therefore, we think that the second choice is better, as it maintains consistency with WFS. The first choice is potentially confusing to those less familiar with OGC standards because it causes the resultType parameter to have different meanings under the WFS and SOS specifications.

Proposed Specification Modification: SOS 2

- Change the current SOS 2.0 functionality which uses the "resultType" parameter to instead use the "responseResultType" parameter.
- Add an optional "resultType" parameter with two values -- "result" or "metadata" -- with "result" being the default if unspecified.
- Specify the behavior of a SOS response to the "resultType" parameter: it may either generate a complete response document containing resources

that satisfy the operation or it may simply generate an empty response container including a metadata section containing various parameters of interest, including the total number of resources that the operation would return. Which of these two responses a SOS generates is determined by the value of the optional resultType parameter.

OGC Comment Status

Comment #27 of OGC 10-028.

Acceptance of alignment of the parameter name resultType to WFS.

Deferred to future work on the “SOS Large Dataset Extension”.

ISSUE 7. maxObservations, maxResults, maxFeatures, sortBy, sortOrder in SOS

In dealing with large datasets behind the SOS, clients (especially browser clients) can be overwhelmed by the large size of the results . WFS 1.0 provides a way to limit the number of features returned from a GetFeature request via the @maxFeatures attribute. We feel this is easy to implement and worth emulating in SOS. This feature, combined with a sorting instruction, could also solve the problem of getting the latest value of a result, by retrieving the first item of a series sorted in descending temporal sequence.

Proposed Solution

We propose that SOS GetObservation, GetObservationByID and GetFeatureOfInterest add support for maxObservations, maxResults, and maxFeatures, to limit the number of observations and the number of elements of a complex result. We also propose that parameters be added to control the sorting order of observations and also result elements. The query semantics should be able to control observation limits and order, independently of the results. The semantics of these parameters are to be defined by others, such as the SOS SWG. These changes will enable the extraction of the latest Observation from a SOS as follows (parameter names are not official proposals here, but are examples to demonstrate the concept):

```
GetObservation sortBy="phenomenonTime" sortOrder="DESC" maxObservation="1"
```

It would also be possible to get the latest element of a time series in a result by performing a similar request:

```
GetObservation result sorted by "time" result sort Order = "DESC" maxResults = "1"
```

Proposed Specification Modification: SOS 2

To be determined.

OGC Comment Status

Comment #28 of OGC 10-028.

Deferred to future work on the "SOS Large Dataset Extension".

ISSUE 8. Units of Measure

Of all the OGC filter encodings, only the <Distance> element has a @units attribute. This is a problem for clients because physical quantities need units to be meaningful. The client is unable to create a proper filter encoding without knowing the implicit units associated with the numbers in the filter encoding. Furthermore, even if the client knows the proper units, it must do the conversion to the units on the server. Generally, amongst relevant OGC specifications the responsibility to perform unit conversion rests with the client. We think there are two problems with this approach:

- Even if the service wanted to provide unit conversion functionality, it cannot do so.
- There does not exist an easy way for the client to discover units of measure for fields in order to properly construct a filter encoding.

The only available method to discover units of measure is to parse the SWE definition of the property.

Proposed Solution

We propose that the unit description be explicitly referenced in the offering as a SWE dataset. The suggestion below accomplishes the goal of allowing the service to share the responsibility of unit conversion. Furthermore, it has an additional beneficial side effect of making the filter encoding request more self-contained. Existing filter encoding requests make use of implicit units, so may return different kinds of results when submitted to different service implementations.

Proposed Specification Modification: SOS 2, Filter Encoding

Add a <ConversionCapabilities> section to the GetCapabilities response. The service uses this section to announce all the unit conversions it supports.

```
<ConversionCapabilities>
  <ConversionGroup name="urn:ogc:def:length">
    <uom default="true" defaultUnits
id="urn:ogc:def:uom:UCUM:ft">feet</unitsuom>
    <units uom id="http://usgs.gov/mile">miles</unitsuom>
      <units uom
id="http://nrcan.gc.ca/murn:ogc:def:uom:UCUM:m">meters</unitsuom>
    </ConversionGroup>
    <ConversionGroup name="urn:ogc:def:surface">
      <uomnits id="urn:ogc:def:uom:UCUM:sqfhttp://usgs.gov/sqfoot">square
feet</uomnits>
      <uomnits id="urn:ogc:def:uom:UCUM:sqmhttp://usgs.gov/sqmile">square
```



```

miles</uomnits>
    <uomnits id="urn:ogc:def:uom:UCUM:m2http://nrcan.gc.ca/sqm">square
meters</uomnits>
    </ConversionGroup>
</ConversionCapabilities>

```

Furthermore, units of the same dimension should be grouped in ConversionGroup elements. A ConversionGroup may have at most one optional <defaultUnits>.

For each <Offering>/<observedProperty> in GetCapabilities, the server announces the SWE description of the dataset for the observedProperty via a @structure attribute. If no property-specific default unit is declared in observedProperty, then the default unit is assumed to be the default unit of the appropriate ConversionGroup, if there is one:

```

<sos:observedProperty xlink:href="urn:ogc:def:property:OGC:GroundWaterLevel"
structure="http://usgs.gov/sos/gwie/datastructure.xml"/>

```

The structure attribute points to a document compliant with SWE that describes the actual structure of the observed property:

```

<swe:DataArray gml:id="GroundwaterLevel">
  <swe:elementCount>
    <swe:Count>
      <swe:constraint>
        <swe:AllowedValues>
          <swe:min>0</swe:min>
        </swe:AllowedValues>
      </swe:constraint>
    </swe:Count>
  </swe:elementCount>
  <swe:elementType name="Components">
    <swe:SimpleDataRecord>
      <swe:field name="time">
        <swe:Time definition="urn:ogc:property:time:iso8601"/>
      </swe:field>
      <swe:field name="depth">
        <swe:Quantity definition="urn:x-ogc:def:phenomenon:OGC:Depth">
          <swe:uom code="m"/>
        </swe:Quantity>
      </swe:field>
    </swe:SimpleDataRecord>
  </swe:elementType>
  <swe:values/>
</swe:DataArray>

```

For all the Comparison Operators, i.e. PropertyIsEqualTo, ...PropertyIsBetween, add an optional @uom attribute. If no value for @uom is given in a filter request, then the numeric value is assumed to be measured using the default units for the property.

```

<PropertyIsLessThan units="urn:ogc:def:uom:UCUM:ft">
  <PropertyName>DEPTH</PropertyName>
  <Literal>30</Literal>
</PropertyIsLessThan>

```

When the service receives a request containing a filter encoding component with a unit that it either does not recognize or that it cannot convert, it throws a `UnitNotSupportedException` with a message identifying:

- the property at question,
- the acceptable units for that property.

This exception allows the client to easily adjust the filter encoding in case of a unit error without having to parse through a heavy `GetCapabilities` section.

OGC Comment Status, OGC Change Request

Comment #29 of OGC 10-028.

This proposal was deferred to future work for on an extension.

ISSUE 9. SOS 1 Typo

Sos4ogc.xsd (<http://schemas.opengis.net/sos/1.0.0/ogc4sos.xsd>) which provides the filter specification has a misspelled "TM_Overalps" (should be TM_Overlaps). This prevents validating queries from the published schemas (unless the misspelled element is used).

Proposed Specification Modification: SOS 1

Correct spelling of "TM_Overlaps" in Sos4ogc.xsd and in the SOS 1 specification.

OGC Change Request Status

Accepted.

7.3 GWIE WaterML2 Profile

The WaterML2 Profile used by the GWIE is shown below in Figure 2. The following profile elements essentially constrain some property values to a controlled list, or force the cardinality of other properties.

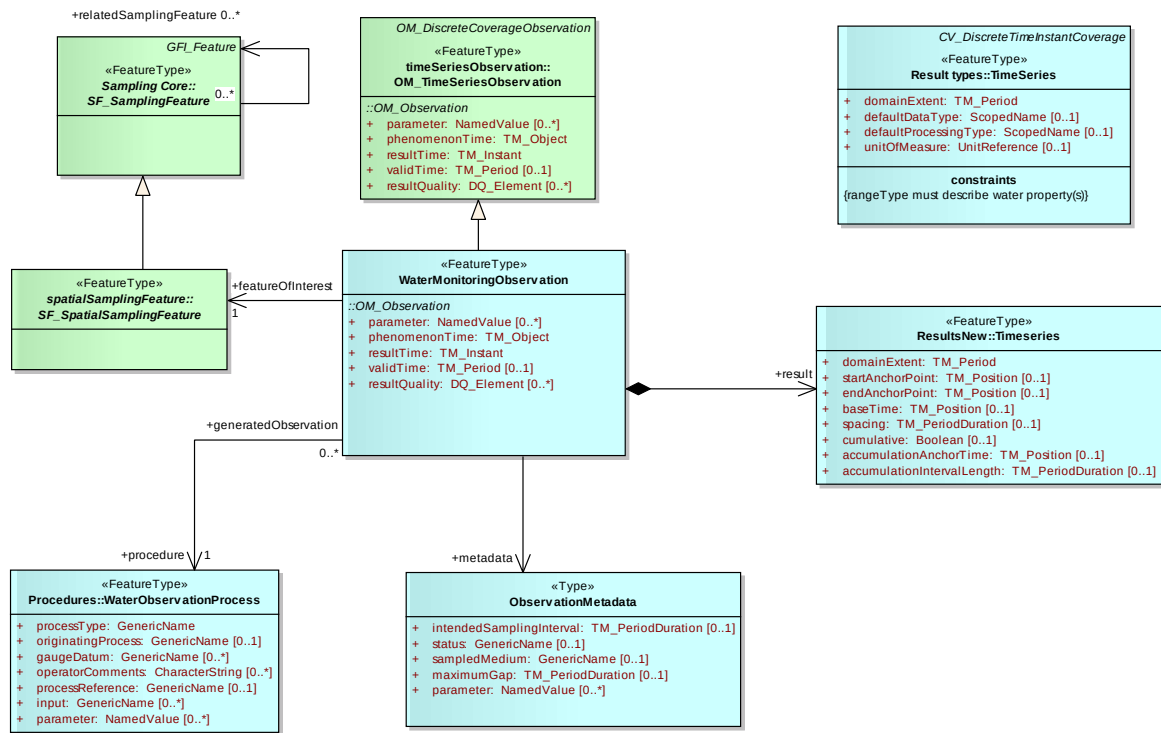


Figure 2: WaterML2 schema used by the GWIE.

WML2-1. wml2:WaterMonitoringObservation/gml:identifier is mandatory.

While gml:identifier is not mandatory in GML, a GWIE compliant service MUST serialize a gml:identifier for each observation. This property must provide a unique identifier for this observation which can eventually be used to identify the observation in a subsequent GetObservationById .request

WML2-2. om:parameter SHALL report spatial location.

As a convenience for applications that want to plot the location of the observation without having to resolve the feature-of-interest (including all the related schema and performance implications), an Observation SHALL comply with sub-clause 7.13 of OGC 10-025 (Cox, 2010b):

```

<?xml version="1.0" encoding="UTF-8"?>
<wml2:WaterMonitoringObservation>
...
  <om:parameter>
    <om:NamedValue>
      <om:name
xlink:href="http://www.opengis.net/req/omxml/2.0/data/samplingGeometry"/>
      <om:value>
        <gml:Point gml:id="SamplingPoint">
          <gml:pos srsName="urn:ogc:def:crs:EPSG:4326">52.9 7.52</gml:pos>
        </gml:Point>
      </om:value>
    </om:NamedValue>
  </om:parameter>
...
</wml2:WaterMonitoringObservation>

```

WML2-3. om:procedure shall be a reference to the water level sensor at the monitoring well.

```

<?xml version="1.0" encoding="UTF-8"?>
<wml2:WaterMonitoringObservation>
...
  <om:procedure xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="urn:ogc:object:Sensor:usgs-gw:USGS-393504074205101"/>
...
</wml2:WaterMonitoringObservation>

```

WML2-4. om:observedProperty SHALL be GroundWaterLevel.

```

<?xml version="1.0" encoding="UTF-8"?>
<wml2:WaterMonitoringObservation>
...
  <om:observedProperty xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="urn:ogc:def:property:OGC:GroundWaterLevel"/>
...
</wml2:WaterMonitoringObservation>

```

WML2-5. om:featureOfInterest SHALL be a resolvable link to the groundwater well in GWML.

```

<?xml version="1.0" encoding="UTF-8"?>
<wml2:WaterMonitoringObservation>
...
  <om:featureOfInterest xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://infotrek.er.usgs.gov/ogc-
ie/wfs?request=GetFeature&featureId=USGS-
393504074205101&typeName=gwml:WaterWell" xlink:title="Groundwater monitoring
well: USGS-393504074205101"/>
...
</wml2:WaterMonitoringObservation>

```

WML2-6. om:result SHALL be wml2:TimeSeries.

The result must conform to the WaterML2.0 time series schema. Because WaterML2 was evolving during the GWIE, we froze the WaterML2 schema for the GWIE to the June 2010 version (which is available in the WaterML2.0 Subversion repository)

WML2-7. Within wml2:TimeSeries, wml2:element/wml2:TimeValuePair/wml2:value/swe:Quantity are scalar values with units that represent distance below land-surface datum.

7.3.1 GWIE WaterML2 instance document example

```
<wml2:WaterMonitoringObservation
xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:om="http://www.opengis.net/om/2.0"
xmlns:sa="http://www.opengis.net/sampling/2.0"
xmlns:swe="http://www.opengis.net/swe/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wml2="http://www.wron.net.au/waterml2"
xmlns:md="http://www.isotc211.org/2005/gmd"
xmlns:gco="http://www.isotc211.org/2005/gco"
xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:wml2="http://www.wron.net.au/waterml2" gml:id="usgs.gw.obs.USGS-
393504074205101">
  <gml:identifier codeSpace="urn:x-usgs:wellname">obs.USGS-
393504074205101</gml:identifier>
  <om:metadata>
    <wml2:ObservationMetadata>
      <md:contact xmlns:md="http://www.isotc211.org/2005/gmd"
xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="http://cida.usgs.gov"/>
      <md:dateStamp xmlns:md="http://www.isotc211.org/2005/gmd">
        <gco:Date xmlns:gco="http://www.isotc211.org/2005/gco">2010-11-
24</gco:Date>
      </md:dateStamp>
      <md:identificationInfo xmlns:md="http://www.isotc211.org/2005/gmd"
xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="urn:OGC:unknown"/>
      <wml2:intendedSamplingInterval>
        <gml:TimePeriod gml:id="invalid.isi.USGS-393504074205101">
          <gml:beginPosition>1986-02-15T00:00-05:00</gml:beginPosition>
          <gml:endPosition>2008-12-16T05:00-05:00</gml:endPosition>
        </gml:TimePeriod>
      </wml2:intendedSamplingInterval>
    </wml2:ObservationMetadata>
  </om:metadata>
  <om:phenomenonTime>
    <gml:TimePeriod gml:id="invalid.pt.USGS-393504074205101">
      <gml:beginPosition>1986-02-15T00:00-05:00</gml:beginPosition>
      <gml:endPosition>2008-12-16T05:00-05:00</gml:endPosition>
    </gml:TimePeriod>
  </om:phenomenonTime>
  <om:resultTime>
    <gml:TimeInstant gml:id="invalid.rt.USGS-393504074205101">
      <gml:timePosition>1986-02-15T00:00-05:00</gml:timePosition>
    </gml:TimeInstant>
  </om:resultTime>
  <om:procedure xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="urn:ogc:object:Sensor:usgs-gw:USGS-393504074205101"/>
  <om:parameter>
    <om:NamedValue>
      <om:name xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://www.opengis.net/req/omxml/2.0/data/samplingGeometry"/>
      <om:value>
        <gml:Point srsName="urn:ogc:def:crs:EPSG:4326"
gml:id="invalid.loc.USGS-393504074205101">
          <gml:pos>39.5859515546095 -74.3467007946351</gml:pos>
        </gml:Point>
      </om:value>
    </om:NamedValue>
  </om:parameter>
  <om:observedProperty xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="urn:ogc:def:property:OGC:GroundWaterLevel"/>
  <om:featureOfInterest xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://infotrek.er.usgs.gov/ogc-
ie/wfs?request=GetFeature&featureId=USGS-
393504074205101&type=usgs:gwml:WaterWell" xlink:title="Sensor site USGS-
393504074205101"/>
  <om:result>
    <wml2:TimeSeries>
      <wml2:domainExtent>
```

```

        <gml:TimePeriod gml:id="invalid.de.USGS-393504074205101">
          <gml:beginPosition>1986-02-15T00:00-05:00</gml:beginPosition>
          <gml:endPosition>2008-12-16T05:00-05:00</gml:endPosition>
        </gml:TimePeriod>
      </wml2:domainExtent>
      <wml2:defaultDataType
codeSpace="http://usgs.gov/water/">InstVal</wml2:defaultDataType>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>1986-02-15T00:00-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">
              <swe:uom code="feet"/>
              <swe:value>22.5</swe:value>
            </swe:Quantity>
          </wml2:value>
          <wml2:comment>Observed.</wml2:comment>
        </wml2:TimeValuePair>
      </wml2:element>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>1988-11-30T14:44-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">
              <swe:uom code="feet"/>
              <swe:value>23.07</swe:value>
            </swe:Quantity>
          </wml2:value>
          <wml2:comment>Steel-tape measurement.</wml2:comment>
        </wml2:TimeValuePair>
      </wml2:element>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>1998-11-19T16:41-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">
              <swe:uom code="feet"/>
              <swe:value>18.56</swe:value>
            </swe:Quantity>
          </wml2:value>
          <wml2:comment>Steel-tape measurement.</wml2:comment>
        </wml2:TimeValuePair>
      </wml2:element>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>2003-12-11T15:30-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">
              <swe:uom code="feet"/>
              <swe:value>16.54</swe:value>
            </swe:Quantity>
          </wml2:value>
          <wml2:comment>Calibrated electric-tape
measurement.</wml2:comment>
        </wml2:TimeValuePair>
      </wml2:element>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>2008-11-26T16:55-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">
              <swe:uom code="feet"/>
              <swe:value>26.46</swe:value>
            </swe:Quantity>
          </wml2:value>
          <wml2:comment>Steel-tape measurement.</wml2:comment>
        </wml2:TimeValuePair>
      </wml2:element>
      <wml2:element>
        <wml2:TimeValuePair>
          <wml2:time>2008-12-16T05:00-05:00</wml2:time>
          <wml2:value>
            <swe:Quantity xmlns:swe="http://www.opengis.net/swe/2.0">

```

```

        <swe:uom code="feet"/>
        <swe:value>23.32</swe:value>
      </swe:Quantity>
    </wml2:value>
    <wml2:comment>Electric-tape measurement.</wml2:comment>
  </wml2:TimeValuePair>
</wml2:element>
</wml2:TimeSeries>
</om:result>
</wml2:WaterMonitoringObservation>

```

7.4 GWIE GWML Profile

GWML 1. GWIE uses GWML 1.0.

The Groundwater Markup Language 1.0 (GWML 1) is implemented by the GWIE without modification. Specifications and examples for GWML 1 can be obtained at: http://ngwd-bdnes.cits.rncan.gc.ca/service/api_ngwds/en/gwml.html. The main aspect of GWML used by the GWIE is the water well model, which is summarized in Figure 3.

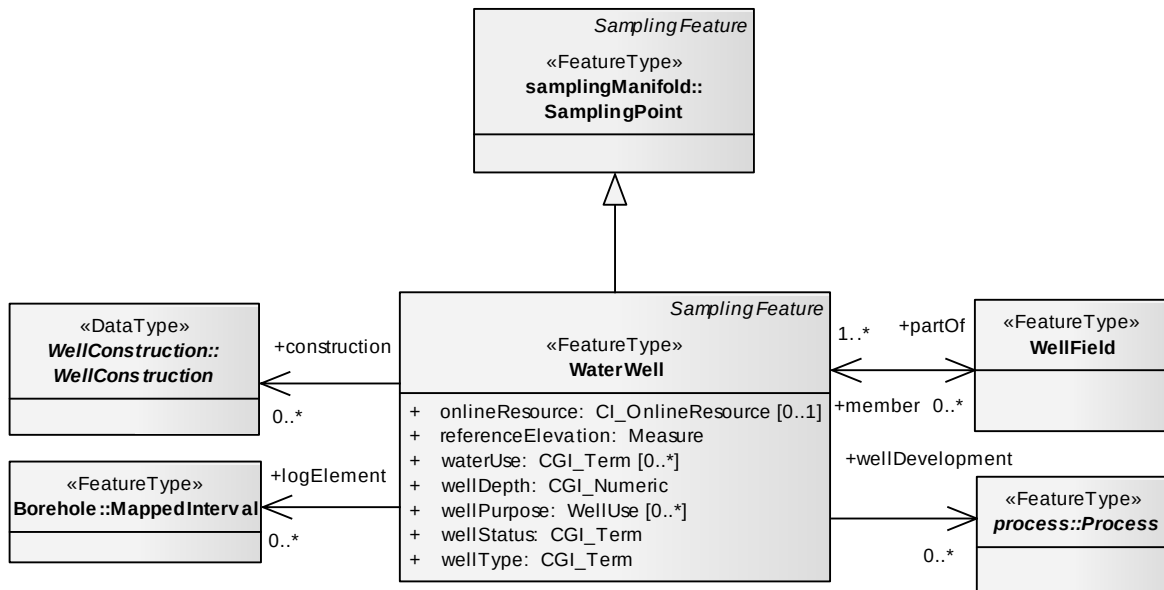


Figure 3: GWML water well model fragment

8 GWIE CLIENT IMPLEMENTATION

8.1 Motivation and Goals

An important evaluation undertaken by the GWIE is assessment of the suitability of the GWIE profile with respect to software clients. Several participants adapt existing web-based client software, or develop new software, as part of the GWIE: GSC, 34North, Kisters, and University of Calgary. This not only helps assess client suitability but also provides testing of the interactions between several OGC interfaces (WMS, WFS and SOS). Of particular note is the independent identification of common issues by multiple client developers.

ISSUE 10. Dense Time Series

A time series can consist of thousands of measurements taken over years by one or more sensors. This large data volume can cause significant performance problems when a time series is returned as a single observation, as is the case in WaterML2. Each GWIE client notes this problem and their proposed solutions are listed below.

Proposed Solutions

- **SOS solution:** the service provides downscaling of data to lower the density, e.g. by serving average values over certain time periods.
- **SOS solution:** implement thematic querying on measured values (GetResult) to constrain SOS results to a portion of the time series, which requires ISSUE 1 above to be resolved and also likely ISSUE 6 and 7. One idea is to use a RESTFUL service to accomplish this.
- **SOS Solution:** provide each sensor's sampling frequency information to clients so that clients can compose reasonable requests to the SOS servers. Sampling rate information can be embedded in either service metadata (e.g., SOS) or sensor metadata (e.g., SensorML). However, each individual sensor in a SOS ObservationOffering might have a different sampling rate. To provide each sensor's sampling rate might result in a very long GetCapabilities XML document. Note: such sampling metadata for the observation can be provided via implementation of the recommendations made in ISSUE 6 above.
- **SOS Solution:** provide a paging mechanism in the SOS GetObservation request. With a paging mechanism, a client can expect a reasonable response time and file size from SOS servers (See ISSUE 7).
- **WaterML solution:** model a time series as an observation collection, where an observation is an atomic measurement (one time point). This allows clients to query for the whole or any part of the time series. A practical downside is the need for each atomic measurement to have an identifier. An

upside is this would allow extraction of single atomic measurements by id, but accompanied by some impacts on querying: it would require a two-step query process to: (1) obtain ids of Results satisfying some condition, and then (2) retrieve those ids.

8.2 GSC Client

The GSC Client is a proof of concept, developed primarily to demonstrate the results of the GWIE. The client operates over both groundwater features, formatted as GWML, and water level observations, formatted as WaterML2. It satisfies the usage scenarios: (1) discovery of data, via WMS viewing of wells; (2) query of single features/observations, via WFS and SOS; (3) display of characteristics for one or more time series observations or one water well feature, and (4) downloading of multiple features/observations also via WFS and SOS. Figure 4 shows the results for usage scenarios (2) and (3): it shows a single USGS groundwater level time series observation associated with a water well feature: the horizontal chart at the bottom shows the time series observation (water level fluctuations over time) and the vertical chart on the right shows the water well feature (the rock log recorded along the length of the well).

Observation 1:

Name: obs.USGS-474917093144601
Feature of interest (GWML): Water Well #
Procedure: urn:ogc:object:Sensor:usgs-gw:USGS-474917093144601
Observed property: urn:ogc:def:property:OGC:GroundWaterLevel

METADATA	
Contact	http://cida.usgs.gov
Date stamp	
Identification Info	urn:OGC:unknown
Intended sampling Interval	Period - Beginning: 1970-09-04T00:00-06:00 - End: 1999-07-24T00:00-06:00

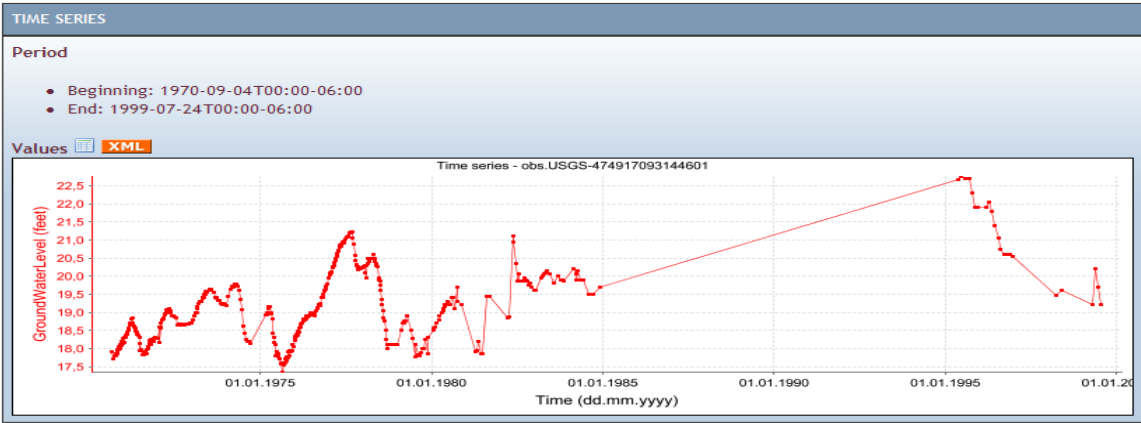


Figure 4: display of time series observation and water well feature rock log.

8.3 University of Calgary (UC)

Sensor Web Browser is a virtual globe-based SWE client developed by the GeoSensor Web Lab at the University of Calgary. Sensor Web Browser allows users to search for and browse distributed and heterogeneous sensing resources within a coherent virtual globe frontend. Currently, Sensor Web Browser is supporting the following OGC standards: WMS, SOS, SPS, O&M, SensorML, and WaterML. In the scope of the GWIE, Sensor Web Browser allows users to navigate the browser's virtual globe frontend, zoom in to users' area of interests, and discover sensors (Figure 5). Users can then access the latest measurement of each sensor by simply moving the mouse cursor over the sensor icons (Figure 5). By right clicking a sensor icon, users can also access further information of the specific sensor, such as time series graph, feature-of-interest (Figure 6), and sensor meta data (the SensorML in Figure 7). Users can also export sensor measurements in CSV format by selecting a bounding box and choosing a time period (Figure 8).

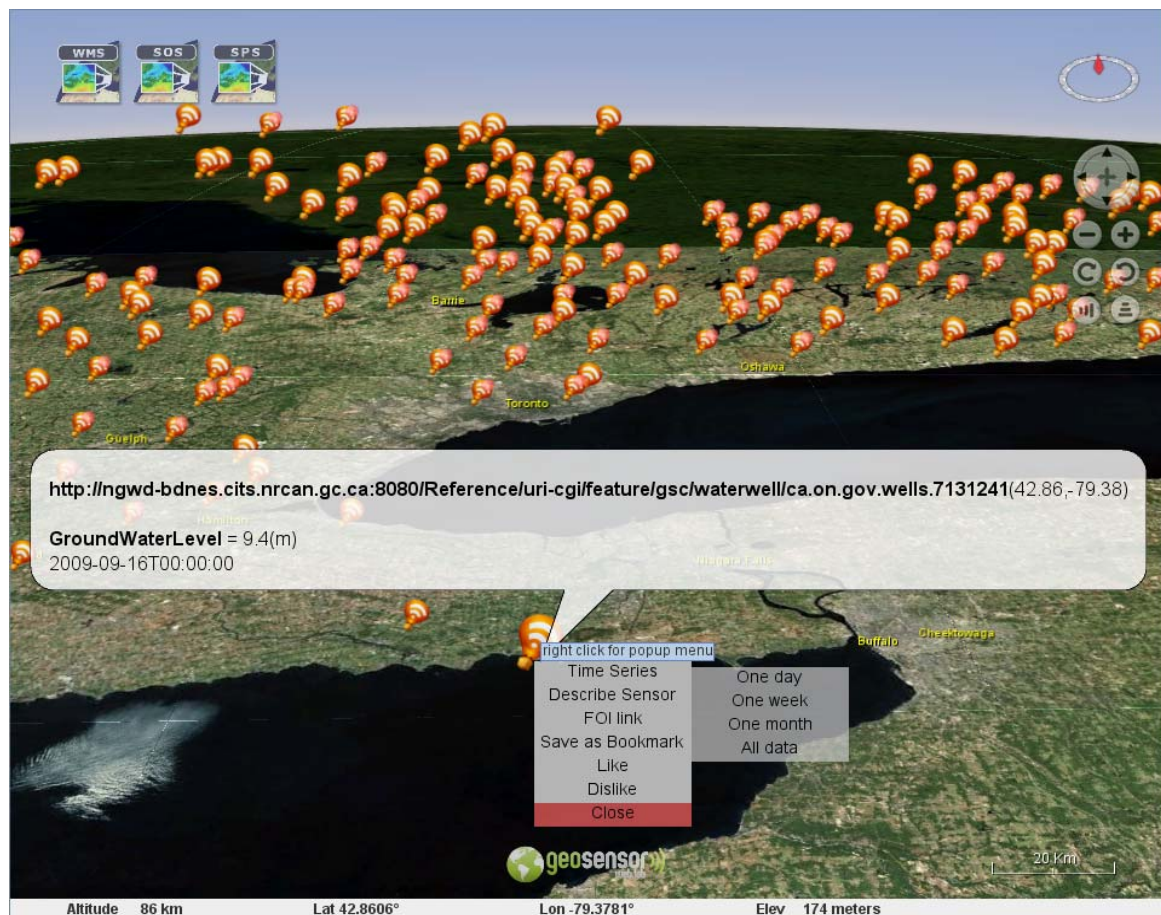


Figure 5: using Sensor web browser to browse sensors

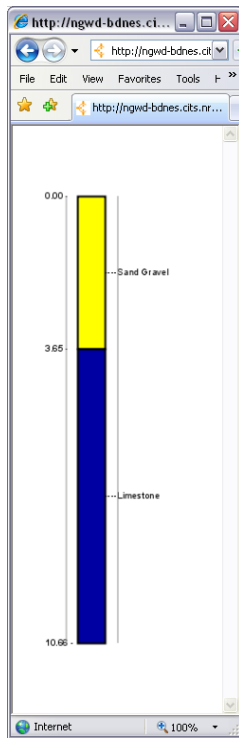


Figure 6: displaying water well info.

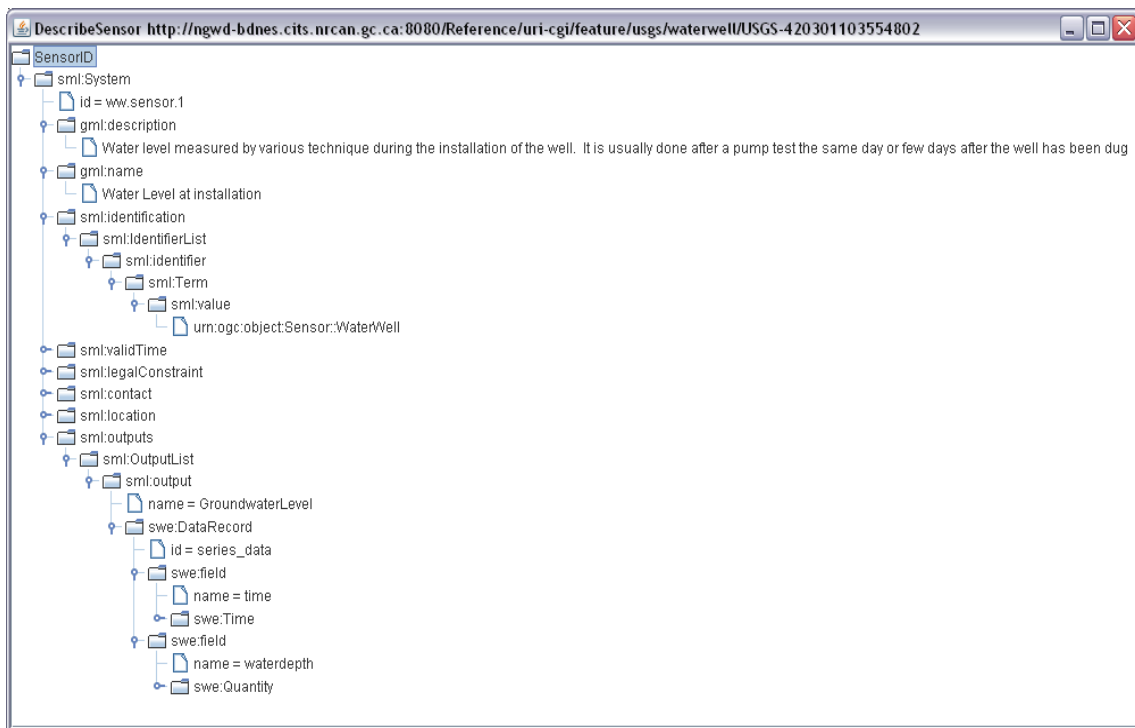


Figure 7: sensor meta data via SOS DescribeSensor interface.

	A	B	C	D	E	F	G
1	Property	foiD	ProcedureID	Latitude	Longitude	Time	Value
2	[OntarioWaterWell]GroundWaterLevel	http://ngwd-b(urn:ogc:object:Se	44.12752291	-77.10300232	2009-09-28T00:00:00		4
3	[OntarioWaterWell]GroundWaterLevel	http://ngwd-b(urn:ogc:object:Se	44.32909762	-79.82968418	2009-09-30T00:00:00		25.2984
4	[OntarioWaterWell]GroundWaterLevel	http://ngwd-b(urn:ogc:object:Se	43.52663918	-80.1188228	2009-09-30T00:00:00		48.768
5	[OntarioWaterWell]GroundWaterLevel	http://ngwd-b(urn:ogc:object:Se	44.86840351	-81.33362658	2009-09-28T00:00:00		14.0208

Figure 8: data export to CSV format

8.4 34 North (34N)

The 34 North client application is a web based collaborative resource management application (OpenNRM) developed in response to the ever increasing need to visualize, manage and record vast amounts of data about our natural environment in a collaborative way. The application is designed to bridge the gap between the growing complexity of data and information available to researchers/academia, scientists, policy makers, resource managers and the public. The 34 North set of utilities allow interested parties to explore, sift, and visualize information for the purpose of saving and sharing the results. The GIS utilities consume several of the OGC interoperable services including but not necessarily limited to WMS, WFS, SOS, and both groundwater features, formatted as GWML, and water level observations, formatted as WaterML2. In response to the GWIE, 34 North has adapted the OpenNRM client to satisfy three usage scenarios identified over the course of the GWIE:

- Data discovery, via WMS viewing of wells;
- Query of single features/observations, via WFS and SOS;
- Download: retrieve data from multiple observations/sensors and water wells.

8.4.1 Discovery of Data via WMS

The 34 North OpenNRM Client software allows for immediate reading and registering of WMS services via the getCapabilities routine basic to the WMS. Figure 9 illustrates the WMS Service of the NRC over the Canadian province of Ontario. Figure 10 illustrates the USGS Center for Integrated Data Analytics WMS 1.1.1 service showing the locations of various water well features throughout the State of Michigan.

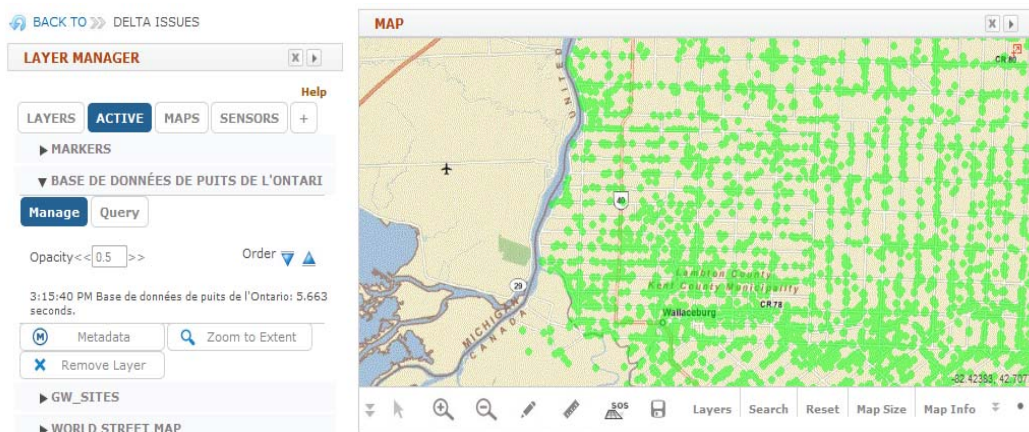


Figure 9: NRC WMS Layer demonstrating the density of the Canadian GW sites.

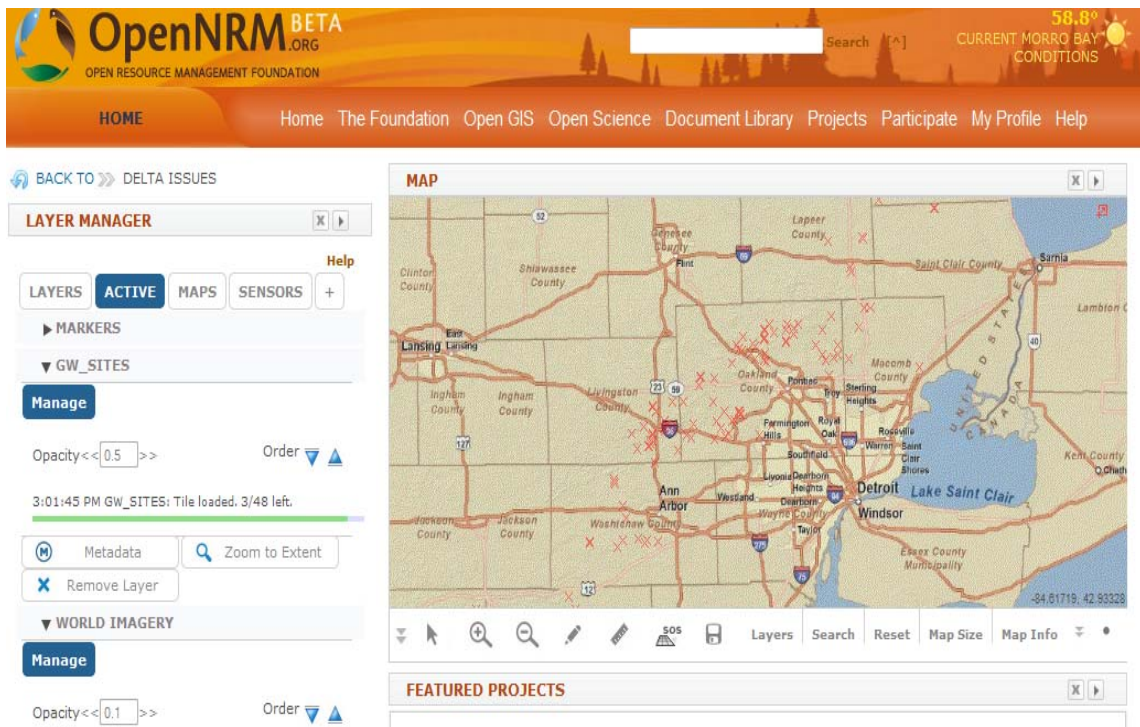


Figure 10: USGS service for water well features throughout the State of Michigan.

8.4.2 Query of single features/observations, via WFS and SOS

Querying of single features and displaying / evaluating the results is accomplished by dragging a bounding box across the map object for any region of interest. Figure 11 shows this process in action for a small region in Ontario.

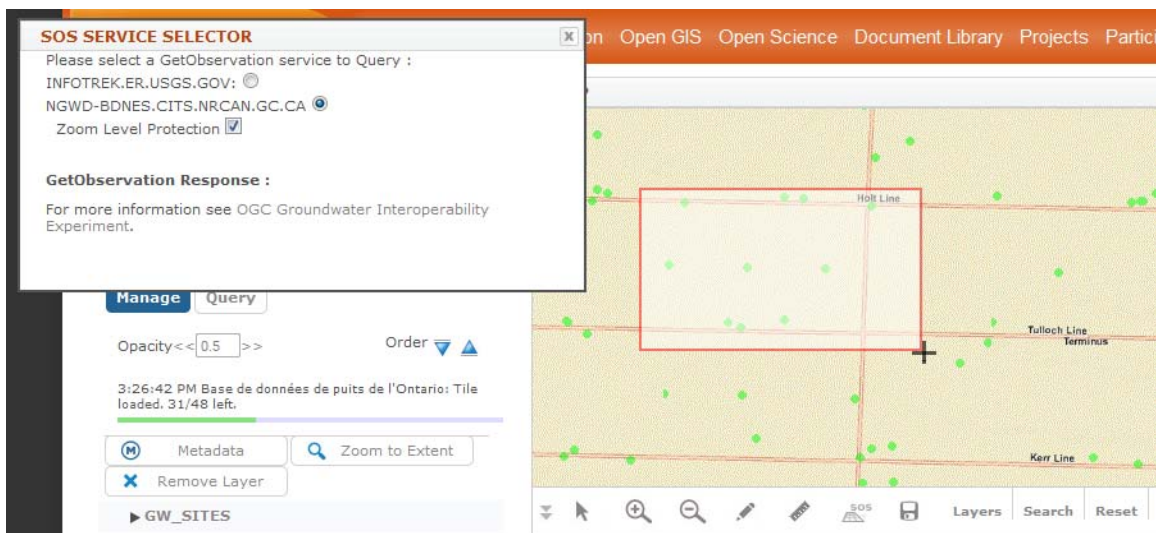


Figure 11: single observation query for Ontario region.

8.5 NCSA GWIE Client and Service

The NCSA implementation uses Illinois groundwater elevation data from 1993 to 2009, provided by the Illinois State Water Survey. The implementation consists of a RESTFUL SOS and a web client to browse the water levels. At the time of this report the service does not yet support WaterML2.

8.5.1 NCSA Client

The web client is live at <http://sensorweb.ncsa.uiuc.edu:8080/OGC/>, and is shown in Figure 12 charting groundwater levels.

8.5.2 NCSA RESTFUL SOS

This web site provides a time-series charting service to visualize the groundwater water elevation data in near real-time (in a streaming data fashion). It also provides various download formats including a SOS service. Specifically, we provide a RESTFUL SOS URL to allow executing any temporal window query that includes a “start” and “end” time.

Here is an example of the RESTFUL SOS URL:

<http://sensorweb.ncsa.uiuc.edu:8080/OGC/stream/window/start/757058400000/end/1265436000000/format/sos/data/true/uri/urn:streams/ogc/well/pnumber/294632&>

Here is a fragment of the response:

```
<om:ObservationCollection xsi:schemaLocation="http://www.opengis.net/om/1.0
http://schemas.opengis.net/om/1.0.0/om.xsd">
  <om:member>
    <om:Observation>
      <om:samplingTime>
        <gml:TimePeriod xsi:type="gml:TimePeriodType">
          <gml:beginPosition>1993-12-28T00:00:00-06:00</gml:beginPosition>
          <gml:endPosition>2010-02-06T00:00:00-06:00</gml:endPosition>
        </gml:TimePeriod>
      </om:samplingTime>
      <om:procedure xlink:href="urn:streams/ogc/well/pnumber/294632"/>
      <om:observedProperty>
        <swe:CompositePhenomenon gml:id="cpid0" dimension="1">
```

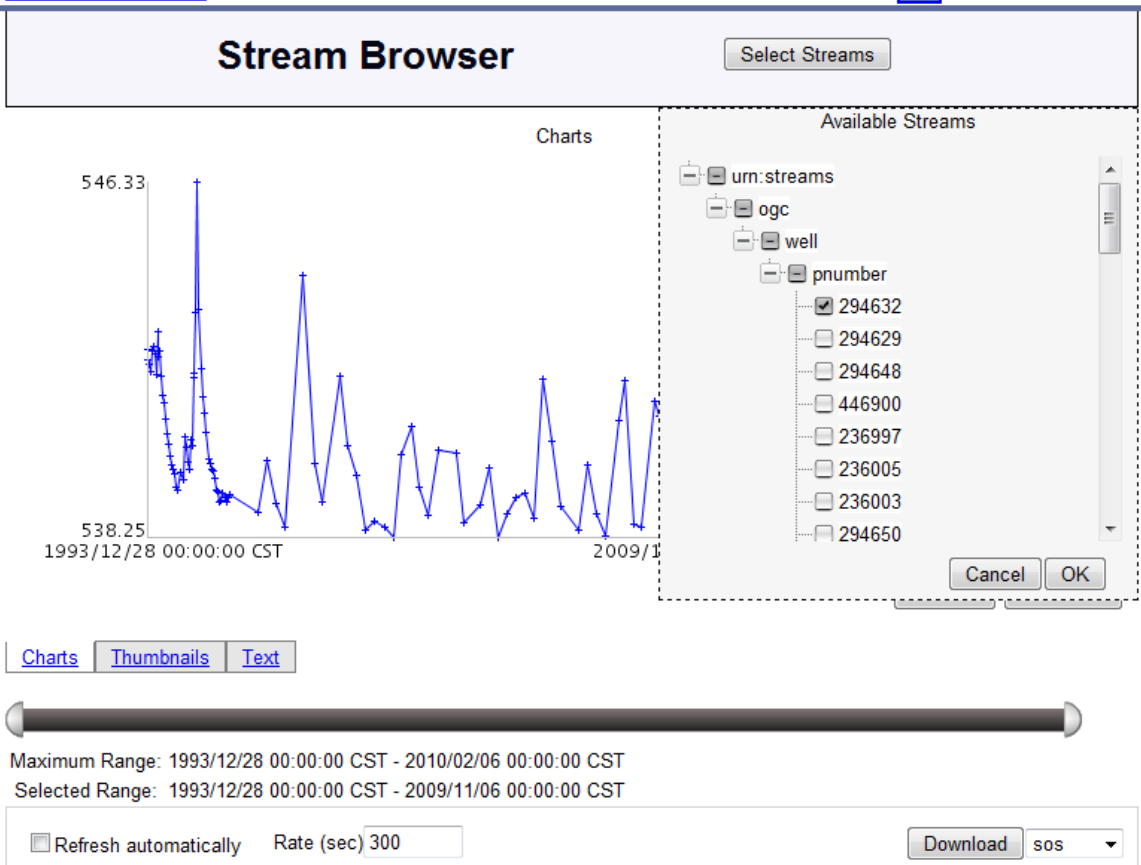


Figure 12: NCSA web client for browsing groundwater levels.

9 ACKNOWLEDGEMENTS

The GWIE is grateful for the invaluable guidance provided by the OGC, and for the commitment and dedication of the GWIE participants. The GSC acknowledges the support of Natural Resources Canada including the Groundwater Geoscience Program, as well as GeoConnections who supplied OGC membership. USGS participants acknowledge the support of the USGS Groundwater Resources Program. CUAHSI acknowledges support from National Science Foundation grant no. EAR-0622374.