Open Geospatial Consortium Inc.

Date: 2007-05-8

Reference number of this OGC® project document: **OGC 07-041r1**

Version: 0.3.0

Category: OGC® Discussion Paper

Editors: Ilya Zaslavsky, David Valentine, Tim Whiteaker

CUAHSI WaterML

Copyright notice

Warning

This document is not an OGC Standard. This document is an OGC Discussion Paper and is therefore not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, an OGC Discussion Paper should not be referenced as required or mandatory technology in or mandatory technology in procurements

Document type:	OGC® Publicly Available Draft
Document subtype:	Candidate Discussion Paper
Document stage:	Draft
Document language:	English

.

Contents

Figures

Tables

i. Preface

WaterOneFlow is a term for a group of web services created by and for the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) community. CUAHSI is an organization representing more than 100 US universities that is supported by the National Science Foundation to develop infrastructure and services for the advancement of hydrologic science. CUAHSI web services facilitate the retrieval of hydrologic observations information from online data sources using the SOAP protocol. CUAHSI WaterML (below referred to as WaterML) is an XML schema defining the format of messages returned by the WaterOneFlow web services.

This document was produced as part of the NSF-supported CUAHSI HIS (Hydrologic Information System Project), and describes the initial version of the WaterML schema in the context of the WaterOneFlow services implementation. CUAHSI is in discussions with OGC about further standardization of the schema and the service signatures, and aligning them with OGC specifications.

Suggested additions, changes, and comments on this discussion paper are welcome and encouraged. Such suggestions may be submitted by OGC portal message, email message, or by making suggested changes in an edited copy of this document.

The changes made in this document version, relative to the previous version, are tracked by Microsoft Word, and can be viewed if desired. If you choose to submit suggested changes by editing this document, please first accept all the current changes, and then make your suggested changes with change tracking on.

ii. Submitting organizations

The following organizations submitted this Implementation Specification to the Open Geospatial Consortium Inc. as a Request For Comment (RFC):

a) University of Texas at Austin (UT-Austin)

b) San Diego Supercomputer Center (SDSC)

iii. Submission contact points

All questions regarding this submission should be directed to the editor or the submitters:

CONTACT	COMPANY	EMAIL
Ilya Zaslavsky	SDSC	zaslavsk [at] sdsc.edu
David Valentine	SDSC	valentin [at] sdsc.edu
Tim Whiteaker	UT-Austin	twhit [at] mail.utexas.edu

iv. Revision history

Date	Release	Author	Paragraph modified	Description
2007-03-20	0.1.1	Ilya Zaslavsky, David Valentine, Tim Whiteaker	Baseline version	Specification of WaterML 1.0 as implemented in WaterOneFlow 1.0 web services
2007-05-08	0.3	Carl Reed	Various. Added future work	Get document ready for posting as DP
2007-05-08	0.3	Simon Cox	Various	Future work content and edits

v. Changes to the OGC® Abstract Specification

The OGC® Abstract Specification does not require changes to accommodate this OGC® standard.

vi. Future work

In future versions of this specification, we intend to harmonize WaterML and WaterOneFlow with relevant OGC specifications. WaterML is most closely related to Observations and Measurements, and might be re-cast as a formal profile of O&M. WaterOneFlow is related to WCS/SOS/SAS and both might be interpreted as implementations of some conceptual observation service.

Foreword

This document is being provided to the OGC for review and discussion by the OGC membership. There may be potential harmonization work to align WaterML with both GML and Observations and Measurements.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights. However, to date, no such rights have been claimed or identified.

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 specification set forth in this document, and to provide supporting documentation.

Introduction

Beginning in 2005, the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI), as part of its Hydrologic Information System (HIS) project, implemented a variety of web services providing access to large repositories of hydrologic observation data, including the USGS National Water Information System (NWIS), and the US Environmental Protection Agency's STORET (Storage and Retrieval) database of water quality information. The services provide access to station and variable metadata, and observations data stored at these sites. As these services were without any formal coordination, their inputs and outputs were different across data sources. Linking together services developed separately in an ad hoc manner does not scale well. As the number and heterogeneity of data streams to be integrated in CUAHSI's hydrologic data access system increased, it would become more and more difficult to develop and maintain a growing set of client applications programmed against the different signatures and keep track of data and metadata semantics of different sources. As a result, WaterML was developed to provide a systematic way to access water information from point observation sites.

In parallel, CUAHSI was also developing an information model for hydrologic observations that is called the Observations Data Model (ODM). Its purpose is to represent observation data in a generic structure that accommodates different source schemas.

While based on the preliminary set of CUAHSI web services, WaterML was further refined through standardization of terminology between WaterML and ODM, and through analysis of access syntax used by different observation data repositories, including USGS NWIS, EPA STORET, NCDC ASOS, Daymet, MODIS, NAM12K, etc. WaterML and ODM, at present, are not identical. WaterML includes detailed information that is not incorporated in ODM, for example source information. Designed to be maximally uniform across both field observation sources and observations made at points, and interoperate with observation data formats common in neighbouring disciplines, it accommodates a variety of spatial types and time representations. WaterML incorporates structures that support on-the-fly translation of spatial and temporal characteristics, and includes structures for SOAP messaging.

1 Scope

This document describes the initial version of the WaterML messaging schema as implemented in version 1 of WaterOneFlow web services. It also lays out strategies for harmonizing WaterML with OGC specifications, the Observations and Measurement specification in particular.

The CUAHSI WaterOneFlow Application Programming Interface (API) is a simple set of methods that can be called to discover and retrieve hydrologic observations data. The core web services API is described in Clause 6.3. WaterOneFlow web services may contain additional methods specific to a given data source; these extended methods are reviewed in Annex B.

The services are available from http://water.sdsc.edu.

2 Conformance

Not applicable at this time.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this document are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

ISO 1000:1994, *SI units and recommendations for the use of their multiples and of certain other units.*

ISO 8601:2004, *Data elements and interchange formats — Information interchange Representation of dates and times*

ISO 19101:2003, *Geographic Information—Reference Model*

ISO/TS 19103:2006, *Geographic Information — Conceptual schema language*

ISO 19110:2006 , Geographic Information – Feature cataloguing methodology

IETF RFC 2396, *Uniform Resource Identifiers (URI): Generic Syntax. (August 1998)*

OGC Observations and Measurements. OpenGIS® Best Practice document, OGC 05-087r4 http://portal.opengeospatial.org/files/?artifact_id=17038

OGC Sensor Observation Service. OpenGIS® Implementation Specification, OGC 07-0009r5 http://portal.opengeospatial.org/files/?artifact_id=20994&version=1

W3C XLink, *XML Linking Language (XLink) Version 1.0. W3C Recommendation (27 June 2001)*

W3C XML, *Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation (6 October 2000)*

W3C XML Namespaces, *Namespaces in XML. W3C Recommendation (14 January 1999)*

W3C XML Schema Part 1, *XML Schema Part 1: Structures. W3C Recommendation (2 May 2001)*

W3C XML Schema Part 2, *XML Schema Part 2: Datatypes. W3C Recommendation (2 May 2001)*

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1
application schema
conceptual schema for data required by one or more applications

[ISO 19101]

4.2
attribute <XML>
name-value pair contained in an **element**

4.3
child element <XML>
immediate descendant **element** of an **element**

4.4
coordinate reference system
coordinate system that is related to the real world by a datum [ISO 19111]

4.5
coverage
feature that acts as a function to return values from its range for any direct position within its spatiotemporal domain

[ISO 19123]

4.6
data type
specification of a value domain with operations allowed on values in this domain

[ISO/TS 19103]

EXAMPLE Integer, Real, Boolean, String, Date (conversion of a data into a series of codes).

Data types include primitive predefined types and user-definable types. All instances of data types lack identity.

4.7
domain
well-defined **set**

[ISO/TS 19103]

1 A mathematical function may be defined on this set, i.e. in a function f:A→B A is the domain of the function f.

2 A domain as in domain of discourse refers to a subject or area of interest.

4.8
element <XML>
basic information item of an XML document containing **child elements**, **attributes** and character data

From the XML Information Set: “Each XML document contains one or more elements, the boundaries of which are either delimited by start-tags and end-tags, or, for empty elements, by an empty-element tag. Each element has a type, identified by name, sometimes called its ‘generic identifier’ (GI), and may have a set of attribute specifications. Each attribute specification has a name and a value.”

4.9
feature
abstraction of real world phenomena

[ISO 19101]

A feature may occur as a type or an instance. Feature type or feature instance should be used when only one is meant.

4.10
feature association
relationship that links instances of one feature type with instances of the same or different feature type

[ISO 19110]

4.11
grid
network composed of two or more sets of **curves** in which the members of each set intersect the members of the other sets in an algorithmic way

[ISO 19123]

The curves partition a space into grid cells.

4.12
namespace <XML>
collection of names, identified by a URI reference, which are used in XML documents as element names and attribute names [W3C XML Namespaces]

4.13
observation (noun)
an act of observing a property or phenomenon, with the goal of producing an estimate of the value of the property.

4.14
property <General Feature Model>
characteristic of a feature type, including attribute, association role, defined behaviour, feature association, specialization and generalization relationship, constraints

[ISO 19109]

4.15
property <GML>
a **child element** of a GML object

It corresponds to feature attribute and feature association role in ISO 19109. If a GML property of a feature has an xlink:href attribute that references a feature, the property represents a feature association role.

4.16
schema
formal description of a model

[ISO 19101]

In general, a schema is an abstract representation of an object's characteristics and relationship to other objects. An XML schema represents the relationship between the attributes and elements of an XML object (for example, a document or a portion of a document)

4.17
schema <XML Schema>
collection of schema components within the same target **namespace**

EXAMPLE Schema components of W3C XML Schema are types, elements, attributes, groups, etc.

4.18
schema document <XML Schema>
XML document containing schema component definitions and declarations

The W3C XML Schema provides an XML interchange format for schema information. A single schema document provides descriptions of components associated with a single XML namespace, but several documents may describe components in the same schema, i.e. the same target namespace.

4.19
semantic type
category of objects that share some common characteristics and are thus given an identifying type name in a particular domain of discourse

4.20
tag <XML>
markup in an XML document delimiting the content of an **element**

EXAMPLE <Road>

A tag with no forward slash (e.g. <Road>) is called a start-tag (also opening tag), and one with a forward slash (e.g. </Road> is called an end-tag (also closing tag).

4.21
UML application schema
application schema written in UML according to ISO 19109

4.22
Uniform Resource Identifier (URI)
unique identifier for a resource, structured in conformance with IETF RFC 2396

The general syntax is <scheme>::<scheme-specific-part>. The hierarchical syntax with a namespace is <scheme>://<authority><path>?<query> - see [RFC 2396].

4.23
value
member of the value-space of a datatype. A value may use one of a variety of scales including nominal, ordinal, ratio and interval, spatial and temporal. Primitive datatypes may be combined to form aggregate datatypes with aggregate values, including vectors, tensors and images [ISO11404].

5 Conventions

5.1 Symbols (and abbreviated terms)

API Application Program Interface

ASOS Automated Surface Observing System

COTS Commercial Off The Shelf

CUAHSI	Consortium of Universities for the Advancement of Hydrologic Science, Inc.
DAYMET	Daily meteorological surfaces modeled at NCAR
EPA	Environmental Protection Agency
GML	Geography Markup Language
ISO	International Organization for Standardization
MODIS	Moderate Resolution Imaging Spectroradiometer
NCAR	National Center for Atmospheric Research
NCDC	National Climatic Data Center
NWIS	National Water Information System
O&M	Observations and Measurements
ODM	Observation Data Model
OGC	Open Geospatial Consortium
OWS	OGC Web Services
STORET	Storage and Retrieval, an information system at EPA
UML	Unified Modeling Language
USGS	United States Geological Survey
WXS	W3C XML Schema Definition Language
WaterML	CUAHSI Water Markup Language
XML	Extensible Markup Language
1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional

5.2 XML conventions

To describe the parts of an XML file in text, this document uses the following conventions:

- Element names are enclosed in brackets, e.g. <element>
- Attributes are prefixed with the @ symbol, e.g. @attribute
- Element text (text content of an element) is enclosed in quotes, e.g. "element text"

The following example XML illustrates these conventions. The example shows a <siteCode> element, with a @network and siteID attributes, and a text value of "010324500".

```
<siteCode network="NWIS" siteID="4622895">010324500</siteCode>
```

6 WaterML Core Concepts and Implementation Context

6.1 Introduction

In this clause, we discuss the conceptual model behind the design of WaterML. For the XML details of each element please refer to Clause 7.

The CUAHSI Water Markup Language (WaterML) is an XML schema defining the elements that are designed for WaterOneFlow messaging, in support of the transfer of water data between a server and a client. WaterML generally follows the information model of ODM (Observation Data Model) described at http://www.cuahsi.org/his/odm.html. WaterML generally shares terminology with ODM, while providing additional terms to further document aspects of both the data retrieved and the retrieval process itself.

The WaterML schema is defined at http://water.sdsc.edu/waterOneFlow/documentation/schema/cuahsiTimeSeries.xsd

The goal of the first version of WaterML was to encode the semantics of hydrologic observations discovery and retrieval and implement WaterOneFlow services in a way that creates the least barriers for adoption by the hydrologic research community. In particular, this implied maintaining a single common representation for the key constructs returned on web service calls. Conformance with OGC specifications was not the goal of this initial version. Hence, throughout this document we accompany WaterML description with notes on possible harmonization of WaterML with the specifications listed below in section 3.

While addressing both point observation and coverage sources, WaterOneFlow web services are primarily built around a Point Observation Information Model illustrated below. This model is further described in Clause 6 of this document.

According to the model, a Data Source operates one or more observation networks; a Network is a set of observation sites; a Site is a point location where water measurements are made; a Variable describes one of the types of measurements; and a time series of Values contains the measured data, wherein each value is characterized by its time of measurement and possibly by a qualifier which supplies additional information about the data, such as a < symbol for interpreting water quality measurements below a detection limit. The WaterOneFlow services GetNetworkInfo, GetSiteInfo and GetVariableInfo describe the networks, sites and variables individually, and the service GetValues is the one that actually goes to the data source and retrieves the observed data.

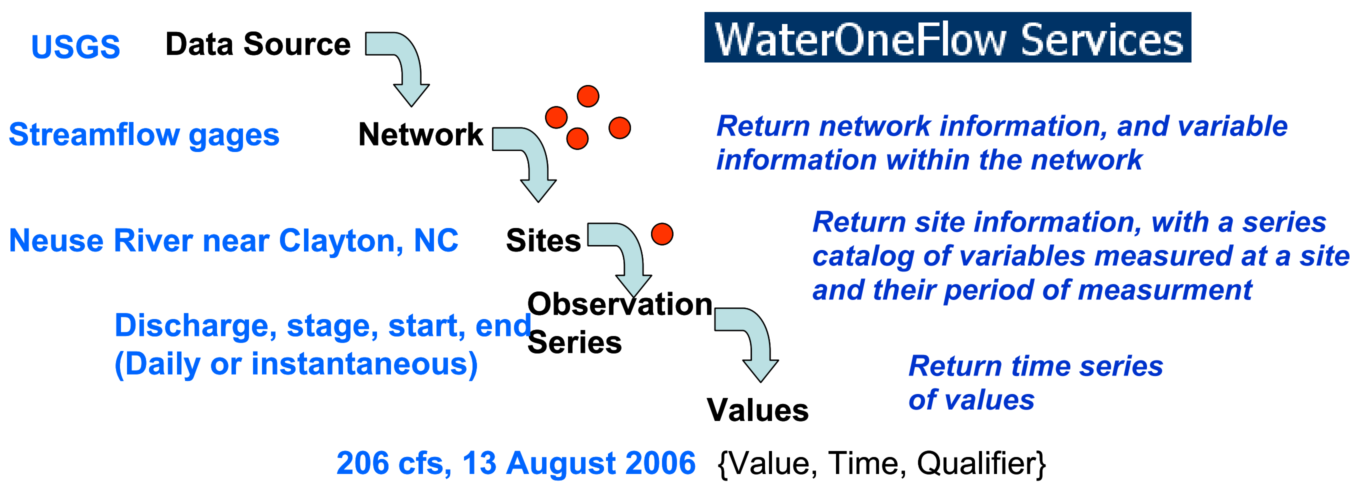

Figure 1. CUAHSI Point Observation Information Model.

6.2 Core concepts

6.2.1 Space, Time, Variable

An *observation* is considered an act of assigning a number, term or other symbol to a phenomenon; the number, term or symbol is the *result* of the action. For the purposes of this document, the terms *observation* and *measurement* are essentially equivalent, the only difference being that a measurement has a quantitative result, while an observation is generic (see OGC® 05-087r4 "Observations and Measurements"). Hydrologic observations are performed against many different phenomena (*properties* of different *features of interest*), and are related to specific times (time points or time intervals.

The *features of interest* common in hydrologic observations may include points (gauging stations, test sites), linear features (streams, river channels), or polygon features (catchments, watersheds).Spatial properties of the features of interest may be further expressed in 2D or 3D, in particular via vertical offsets against common reference features. The observations are made in a particular *medium* (water, air, sediments) using a *procedure.* The procedure may represent a multi-step processing chain including an instrument (sensor), algorithms for transforming the initially measured property (e.g. "partial pressure of oxygen in the water" may be transformed into a measure of "dissolved oxygen concentration"), and various techniques for aggregating, averaging, interpolating and extrapolating censoring and quality-controlling of the value assignment, including multiple scenarios for assignment of no value. As in OGC® 05-087r4, one of the key ideas is that "the observation *result* is an estimate of the value of some property of the feature of interest, and the other observation properties provide context or metadata to support evaluation, interpretation and use of the result.".

The practice of hydrologic observations provides ample evidence of complications beyond this concept. These complications are related to huge, complex and incompatible vocabularies used by several federal hydrologic observation systems, to different and not always documented contexts of measurement and value assignment, to often ambiguously defined features of interest, to complex organizational contexts of hydrologic measurement, transformation and aggregation, etc. Some of them are reviewed in the Annex B (Informative). It is in response to this complexity that the CUAHSI WaterML is primarily designed. Note that some of this complexity may be captured within the Sensor Web standards being developed under the OGC's Sensor Web Enablement (SWE) activity. However, the flexibility inherent in such standards may itself be a barrier to adoption when the target audience is not computer scientists.

At the fundamental level hydrologic observations are identified by the following characteristics:

- The location at which the observations are made (*space*);
- The variable that is observed, such as streamflow, water surface elevation, water quality concentration (*variable*);
- The date and time at which the observations are made (*time*).

Accordingly, elements in WaterML cover those three characteristics, using *sites* and *datasets* to model spatial characteristics, using *variables* to express the variable characteristic; and describing observation *values* via lists of datetime-value pairs representing the temporal dimension of observations.

One of the foundations from which WaterML derives its information model is the CUAHSI Observations Data Model (ODM), as described in the current ODM documentation available at (http://www.cuahsi.org/his/documentation.html). Within this model, the following represent properties of an observation (Table 1).

Table 1. Observation properties in ODM

Property	Definition	Corresponding O&M (as an XPath)
Value	The observation value itself	Observation/result
Accuracy	Quantification of the measurement accuracy associated with the observation value	Observation/observationMetadata/ MD_Metadata/dataQualityInfo/ DQ_DataQuality/report

Date and Time	The date and time of the observation (including time zone offset relative to UTC and daylight savings time factor)	Observation/samplingTime (or possibly Observation/procedureTime)
Variable Name	The name of the physical, chemical, or biological quantity that the value represents (e.g. streamflow, precipitation, water quality)	Observation/observedProperty
Location	The location at which the observation was made (e.g. latitude and longitude)	Observation/featureOfInterest/ SamplingPoint /position
Units	The units (e.g. m or m3/s) and unit type (e.g. length or volume/time) associated with the variable	Observation/result/@uom (where result/@xsi:type=”gml:MeasureType”)
Interval	The interval over which each observation was collected or implicitly averaged by the measurement method and whether the observations are regularly recorded on that interval	Observation/samplingTime/ TimePeriod/duration
Offset	Distance from a reference point to the location at which the observation was made (e.g. 5 meters below water surface)	
Offset Type/ Reference Point	The reference point from which the offset to the measurement location was measured (e.g. water surface, stream bank, snow surface)	
Data Type	An indication of the kind of quantity being measured (currently: instantaneous, continuous, cumulative, incremental, average, maximum, minimum, categorical, constant over interval)	Observation/procedure details
Organization	The organization or entity providing the measurement	Observation/observationMetadata/ MD_Metadata/identificationInfo/ MD_DataIdentification/pointOfContact Or Observation/observationMetadata/ MD_Metadata/distributoinInfo/ MD_Distribution/distributor
Censoring	An indication of whether the observation is censored or not	Observation/ procedureParameter(“censored”,true|false) Or Observation/quality/DQ_ThematicAccuracy Or Observation/result (define a special result type that allows censoring to be indicated)

Data Qualifying Comments	Comments accompanying the data that can affect the way the data is used or interpreted (e.g. holding time exceeded, sample contaminated, provisional data subject to change, etc.)	Observation/quality/DQ_ThematicAccuracy
Analysis Procedure	An indication of what method was used to collect the observation (e.g. dissolved oxygen by field probe or dissolved oxygen by Winkler Titration) including quality control and assurance that it has been subject to	Observation/procedure
Source	Information on the original source of the observation (e.g. from a specific instrument or investigator 3rd party database)	Observation/observationMetadata/ MD_Metadata/dataQualityInfo/ DQ_DataQuality/lineage
Sample Medium	The medium in which the sample was collected (e.g. water, air, sediment, etc.)	Observation/featureOfInterest/ …
Value Category	An indication of whether the value represents an actual measurement, a calculated value, or is the result of a model simulation	Observation/procedure details

Note that WaterML is broader in scope than ODM. ODM is defined over observations made at, or aggregated for, point locations referenced as *sites*, while WaterML is extended to incorporate other spatial feature types.

6.2.2 Observation network, observation series

Individual observations are organized into an *observation series* (a regular sequences of observations of a specific variable made at a specific site), which are in turn referenced in a *series catalog*. The SeriesCatalog table or view in ODM lists each unique site, variable, source, method and quality control level combination found in ODM's Values table, and identifies each by a unique series identifier, SeriesID.

A *series catalog* is an element of an *observation network,* which represents a collection of sites where a particular set of variables is measured. A responsible *organization* can maintain one or more *observation networks*.

In addition to point measurements described in the ODM specification, hydrologic information may be available as observations or model outcomes aggregated over user-defined regions or grid cells. While USGS NWIS and EPA STORET exemplify the former case, sources such as MODIS and Daymet are examples of the latter. In this latter case, as in the case of other remote sensing products or model-generated grids, the observation or model-generated data are treated as *coverages*, and sources of such data are referenced in WaterML as *datasets*, as opposed to *sites*. In other words, WaterML's *dataset* element refers to a type of observations data *source* that is queried by specifying a rectangular region of interest, and the returned time series typically represent some aggregation over the region of interest.

6.2.3 Types in WaterML

WaterML makes extensive use of *polymorphic typing* to support schema flexibility [BUTEK]. As an example, consider that the time series for a given variable is associated with a location in space. If the variable is measured at a stream gage, then the location can be defined by a point in space. However, the variable may also represent the average of an observed phenomenon over a given area, in which case the location may be defined by a collection (aggregation) zone or a box. To allow for both of these representations of space, the initial version of WaterML specifies that spatial location must be described by a generic <GeogLocationType>. This element has only one property, @srs, which indicates the spatial reference system to which the coordinates for the location apply. Thus, the element does not include a means of storing the actual coordinates themselves; the coordinate information is included in elements that extend the initial <GeogLocationType>.

The key to using XML polymorphism is to create additional elements which extend those types. In WaterML, the <LatLonPointType> extends the <GeogLocationType> to include child elements providing the latitude and longitude for a point. Because <LatLonPointType> extends <GeogLocationType>, it must also include the @srs attribute. However, <LatLonPointType> is free to add its own child elements and attributes, which it does to include <latitude> and <longitude> child elements.

Similarly, the first version of WaterML defines a <LatLonBoxType>, which extends <GeogLocationType> by adding four child elements defining the four sides of a bounding box for an area. Thus, by specifying that a location must be defined by a <GeogLocationType>, what WaterML is really saying is that *location* may be defined by a <LatLonBoxType> or <LatLonPointType>. If other means of defining spatial location were to be added to WaterML, the schema and applications built off of the schema would not be broken, so long as the new elements extended the <GeogLocationType> element.

Note that the XML type elements themselves are not returned in a WaterML document. The XML types are like blueprints, and what is actually returned are the objects created from the blueprints. For example, to specify the location of an observation site, the WaterML document returned from a WaterOneFlow web service uses a <geogLocation> element, which is an instance of the <LatLonPointType>. The example XML below shows a <geogLocation> element, which has an @srs attribute from <GeogLocationType>, and <latitude> and <longitude> from <LatLonPointType>. Also notice that it has an @xsi:type attribute that specifies the type of element that <geogLocation> is.

```xml
<geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
  <latitude>30.24</latitude>
  <longitude>-97.69</longitude>
</geogLocation>
```

To help distinguish between XML types (which are abstract) and elements which are instances of those types, XML type names begin with an uppercase letter (e.g. <LatLonPointType>) while instances of those types begin with a lowercase letter (e.g. <geogLocation>).

Note: Location descriptions adopted in the initial version of WaterML do not follow OGC's GML specification. However, many WaterML constructs can be aligned with OGC specifications, as described below in Clause 7.

6.2.4 The basic content, and extensibility

WaterML is primarily designed for relaying fundamental hydrologic time series data and metadata between clients and servers, and to be generic across different data providers. Different implementations of WaterOneFlow services may add supplemental information to the content of messages. However, regardless of whether or not a given WaterML document includes supplemental information, the client shall be sure that the portion of WaterML pertaining to space, time, and variables will be consistent across any data source.

XML Schema is inherently extendable by allowing users to add additional elements in their own namespaces. Creating mixed-content composite documents is convenient in exchanging multi-domain information. However, adding namespaces can be problematic for clients that may not be designed to handle unanticipated information. Schema developers who extend an existing schema must have clear expectations for how a client application should respond to content from unknown namespaces.

WaterML attempts to restrict extensions to clearly defined extensibility points. In some cases, a given source of hydrologic observations data may include additional information, such as the instrument used, the Hydrologic Unit Code (HUC), or the responsible party. The use of these elements is up to the organization maintaining the web service which is making use of WaterML. Advanced clients or customized clients will be able to make use of the supplemental information blocks. All clients shall be able to gracefully handle such information blocks.

6.3 Implementation context

WaterML is currently used as a message format in CUAHSI's WaterOneFlow web services. Depending on the type of information that the client requested, a WaterOneFlow web service will assemble the appropriate XML elements into a WaterML response, and deliver that to the client. The core WaterOneFlow methods include:

• **WaterOneFlow**	•	• **OGC SOS**
• GetSiteInfo	• – for requesting information about an observations site. The	• GetFeatureOfInterest

	returned document has a root element of *SiteInfoResponse* type.	
• GetVariableInfo	• – for requesting information about a variable. The returned document has a root element of *VariableResponse* type.	• GetObservedProperty
• GetValues	• – for requesting a time series for a variable at a given site or spatial fragment of a dataset. The returned document has a root element of *TimeSeriesResponse* type.	• GetResult

The GetValues and GetVariableInfo methods are implemented for all *observation networks* and *datasets* currently covered by WaterOneFlow services. The GetSiteInfo method is implemented over observation networks only. In addition, the GetSites method is implemented over ODM instances containing user-contributed observations datasets. In the current implementation, the initial discovery of sites is done via an online mapping interface, thus a detailed formulation of the GetSites method is left to the next release.

The response types and the respective structure of returned documents, are described in Clause 7.7.

6.4 General issues of bridging with OGC specifications and best practices

There are several directions for connecting the above concepts with the relevant OGC specifications.

- Aligning spatial feature descriptions, e.g. using gml:Point for describing location of *sites* and gml:Envelope for describing rectangular regions of interest.

- Aligning service signatures, in particular, implementing the getCapabilities request to return general information about services, including service identification, service provider, and operations metadata (e.g. as described in WFS Simple profile).

- Aligning the terminology of the ODM (*sites, variables, observation series* and *networks*, etc.) with terms adopted by the O&M specification (*procedureParameter, observedProperty, ObservationCollection*, featureOfInterest, procedure, result, etc.

Some examples of these alignments are given in Clause 7. We will appreciate further ideas and recommendations from OGC membership on this.

7 WaterML element descriptions

7.1 Element naming conventions in WaterML

WaterML terminology has been synchronized, to the maximum possible extent, with the CUAHSI Observations Database Model. Following this model, the adopted naming convention scheme is as follows:

- xxID - internal to application codes that uniquely identify a term/site/unit. These are optional, and are assigned by the database or web service creator.
- xxCode - Alphanumeric. These are the codes that are used to retrieve the sites/variables from the data source, and generally match up with public identifiers for sites/variables within a given network.
- xxType - an element block that is used as a type definition. These are used in the development of the XML schema to differentiate object types, and elements that are reused.

Standards used in the element descriptions:

- Element names have the first letter lower-cased;
- XML parent-type have the first letter upper-cased;
- Extension elements can contain any XML content, and are the location where data providers should place any supplemental information, beyond the basic WaterML content

Some confusion may occur when dealing with the term “Type.” Type is used in multiple ways. The first is when referring to an XML information structure that is inherited. These have the first letter capitalized and are suffixed with Type; eg VariableInfoType , UnitsType. Second is when referring to an element name that is often an enumerated reference. These have the first letter lower case, and are suffixed with Type: valueType, unitsType

RELAX NG compact notation (http://relaxng.org/compact-tutorial-20030326.html#id2814737) is used to outline the element structure of an information set.

```
element sites{
  element siteInfo {SiteInfoType},
  element seriesCatalog {seriesCatalogRecord}+
  }
```

The above structure says that element <sites> contains two elements, <siteInfo>, and <seriesCatalog>. The {}+ say that the element <seriesCatalog> is repeatable. Element <siteInfo> is of SiteInfoType. There can be multiple seriesCatalog elements containing seriesCatalogRecord. In addition to modifier of "+", "?" or optional, and "*" zero or more elements.

For clarity, the details of an included element are often expanded, for example:

```
siteInfo = SiteInfoType

element siteInfo {
  element siteName {string},
  element siteCode {
   attribute network {string},
   attribute siteID {xsd:int}?
  }+,
  element timeZoneInfo {
      . . .
  },
  element geoLocation {
      . . .
  }?,
   . . .
}
```

7.2 Namespace

The namespace should be: http://www.cuahsi.org/waterML/1.0/, as in:

```
default namespace = "http://www.cuahsi.org/waterML/1.0/"
```

7.3 Elements dealing with space

7.3.1 General description

As mentioned in Clause 6.2.2, CUAHSI WaterML currently supports the return of information from two types of sources: collections of observation sites (e.g. stream gages) and datasets where data are typically requested over user-defined region of interest. These spatial components are represented with the <SiteInfoType> and <DatasetInfoType> elements, respectively. Each of these elements has a child element that extends the <GeogLocationType> to express the location of the element in geographic coordinates. The two possible extensions of <GeogLocationType> are <LatLonPointType> for point locations, and <LatLonBoxType> for locations defined by a box in latitude and longitude.

Because <SiteInfoType> represents a site at a discrete location in space, it will have a child element of the <LatLonPointType> type. For <DatasetInfoType>, some datasets return information for a single point, while others return data aggregated over an area.

Thus, elements of either <LatLonPointType> type or <LatLonBoxType> type may be child elements of <DatasetInfoType>.

Any element that extends the <GeogLocationType> element will also have an attribute that defines the coordinate system (e.g., vertical datum, spheroid, etc.) that applies to the latitude and longitude coordinates. Note that all <GeogLocationType> elements represent location in geographic (latitude and longitude) coordinates, assuming WGS84 by default (EPSG code 4326). If elevation information is present in site description, then the default datum and coordinate system definition refers to EPSG code 4979, which specifies the (latitude, longitude, altitude) triplet. In OGC specifications, the coordinate systems are referred to by URNs "urn:ogc:def:crs:EPSG::4326" and "urn:ogc:def:crs:EPSG::4979" respectively. For other coordinate systems and datums, both horizontal and vertical datum information will be included.

In addition to its location, the <SiteInfoType> element also includes data about the site itself, such as <siteName> and <siteCode>. The <DatasetInfoType> includes a <dataSetIdentifier> element that specifies the name of the dataset, e.g. "Daymet". The <SiteInfoType> and <DatasetInfoType> elements themselves are extensions of the generic <SourceInfoType> element. Thus, when WaterML returns information about the location of a site or measurements, the location is returned with an element that is of the <SourceInfoType> type. The figure below shows the possible ways of expressing location in the current version of WaterML.

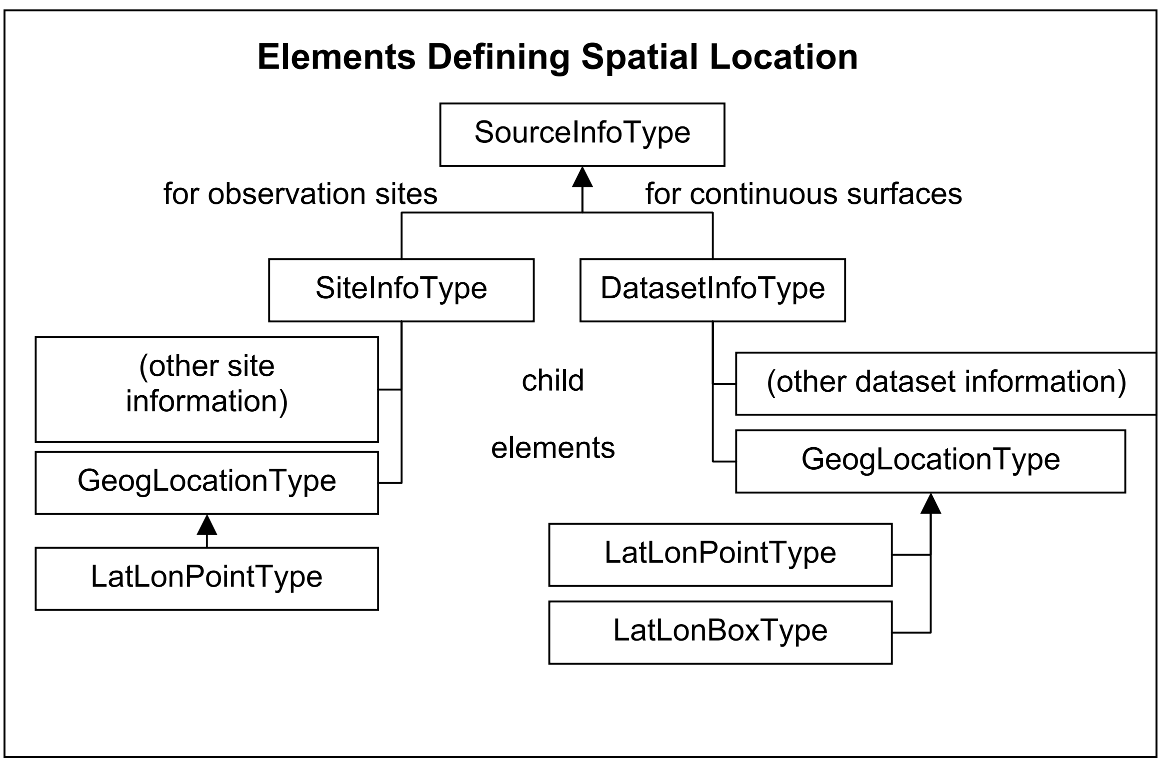

Figure 2. Conceptual Diagram of Elements Defining Spatial Location in WaterML

7.3.2 The SiteInfoType type

7.3.2.1 Annotated structure

```
siteInfo = SiteInfoType

element siteInfo {
  element siteName {string},
  element siteCode {
   attribute network {string},
   attribute siteID {xsd:int}
  }+,
  element timeZoneInfo {
   element defaultTimeZone {
    element zoneAbbreviation{string},
    element zoneOffset {string}
    }?,
   element daylightSavingsTime {
    element zoneAbbreviation{string},
    element zoneOffset {string}
    }?
  },
  element geoLocation {
   element geogLocation {LatLonPointType|LatLonBoxType}
   element localSiteXY {
    element X {double},
    element Y {double},
    element Z {double} ?,
    element projectionInformation {string} ?
   }?,
 },
  element note {
   attribute type {string},
   attibute href {string},
   attribute title {string}
  }*,
  element extension {any}?
  element property {xlink} *
 }
```

Notes:

- Element <siteInfo> is of type <SiteInfoType>.
- The <siteInfo> describes site information, and not the observations at a site. This is done in order to make the <siteInfo> element a reusable object that can be used in multiple messages.
- Element <siteInfo> is used as part of a <timeSeries>, and <site> elements, which themselves are part of the <timeSeriesRespone> and <sitesResponse> messages.

This separation of structure matches the design choices made in ODM which specifies separate tables for sites and series.

- When <siteInfo> is used in a <sitesResponse>, a <site> element includes both the <siteInfo> and one or more <seriesCatalog> elements.
- When used in a <timeSeries>, polymorphism is used, so the <sourceInfo> element will have an xsi:type="SiteInfoType".
- The optional <timeZoneInfo> element, with its <dafaultTimeZone> <daylightSavingsTime> components, uses strings to specify time zone and daylight savings time information for a site. If present, this information may be used for local time conversions at the server.

7.3.2.2 Examples

```
<!-- Generic as used in sitesReponse -->
<siteInfo >
  <siteName>ROCK CK NR BATTLE MOUNTAIN, NV</siteName>
  <siteCode network="NWIS" siteID="4622895">10324500</siteCode>
  <geoLocation>
    <geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
      <latitude>40.83040556</latitude>
      <longitude>-116.5883417</longitude>
    </geogLocation>
  </geoLocation>
</siteInfo>
```

```
<!-- polymorphic as used on timeSeriesReponse/timeSeries -->
<sourceInfo xsi:type="SiteInfoType">
  <siteName>ROCK CK NR BATTLE MOUNTAIN, NV</siteName>
  <siteCode network="NWIS" siteID="4622895">10324500</siteCode>
  <geoLocation>
    <geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
      <latitude>40.83040556</latitude>
      <longitude>-116.5883417</longitude>
    </geogLocation>
  </geoLocation>
</sourceInfo
```

7.3.3 The DataSetInfoType type

7.3.3.1 Annotated structure

```
DataSetInfoType {
  element dataSetIdentifier,
  element dataSetLocation {LatLonPointType |LatLonBoxType},
  element note,
  element extension {any}?
  element property {xlink}*
```

}

Notes:

- DataSetInfoType is only used in a polymorphic type inside of <sourceInfo>.
- <dataSetIdentifier> is the name or reference to the dataset.
- <datasetLocation> contains geometry that was returned. <datasetLocation> is polymorphic and will have xsi:type="LatLonPointType" or xsi:type="LatLonBoxType".

7.3.3.2 Example

```
<!-- polymorphic as used in timeSeriesResponse/timeSeries/ -->
<sourceInfo xsi:type="DataSetInfoType">
  <dataSetIdentifier>DAYMET</dataSetIdentifier>
  <dataSetLocation xsi:type="LatLonPointType" srs="EPSG:4269">
    <latitude>45</latitude>
    <longitude>-113</longitude>
  </dataSetLocation>
</sourceInfo>
```

7.3.4 The site element

7.3.4.1 Annotated structure

```
element site {
  element siteInfo {SiteInfoType},
  element seriesCatalog {seriesCatalogRecord}+
  }
```

Notes:

- <site> is an element within the <sites> element, which is returned on a GetSiteInfo or GetSites API call.
- A site contains two parts: a required <siteInfo> element, and zero or more <seriesCatalog> elements.
- In response to GetSiteInfo request, the number of <seriesCatalog> elements is 1 or more. In response to GetSites request, the number of <seriesCatalog> elements is zero or more (potentially reducing the size of the response).

7.3.4.2 Example

```
<site>
  <siteInfo>
    <siteName>BIG ROCK C NR VALYERMO CA</siteName>
    <siteCode network="NWIS" siteID="4622637">10263500</siteCode>
    <!-- removed  for clarity -->
  </siteInfo>
  <seriesCatalog menuGroupName="USGS Daily Values" serviceWsdl=" ">
    <!-- removed  for clarity -->
  </seriesCatalog>
  <seriesCatalog menuGroupName="USGS Unit Values" serviceWsdl=" ">
    <!-- removed  for clarity -->
  </seriesCatalog>
</site>
```

7.3.5 The LatLonPointType type

7.3.5.1 Annotated structure

latLonPoint = LatLonPointType

element latLonPoint {
 attribute srs {text},
 element latitude {xsd:double},
 element longitude {xsd:double}
}

Notes:

- The @srs should be either an EPSG coded value specified as “EPSG:4326” or a projection string.
- In the current implementation, all services are required to return locations in latitude and longitude, and the clients are not expected to have a projection engine. Coordinate transformations shall be handled at the server, following a coordinate system specified by @srs.

7.3.5.2 Examples

```
<!--Generic Example -->
<latLonPoint srs="EPSG:4326">
  <latitude>35.64722220</latitude >
  <longitude>-78.40527780</longitude >
</latLonPoint>
```

```
<!-- As used in siteInfo source -->
<geogLocation xsi:type="LatLonPointType" srs="EPSG:4326">
  <latitude>35.64722220</latitude >
  <longitude>-78.40527780</longitude >
</geogLocation>
```

7.3.6 The LatLonBoxType type

7.3.6.1 Annotated structure

```
latLonBox = LatLonBoxType

element latLonBox {
attribute srs {text},
  element south {xsd:double},
  element west {xsd:double},
  element north {xsd:double},
  element east {xsd:double}
}
```

Notes:

- A <latLonBox> describes a bounding box. This is defined in terms of North, East, South and West, so that box can cross the international date line (+/-180).
- The @srs should be either an EPSG coded value specified as "EPSG:4326" or a projection string.

7.3.6.2 Examples

```
<!-- Generic Example -->
<latLonBox xsi:type="LatLonBoxType" srs="EPSG:4326">
  <south>45</south>
  <west>-108</west>
  <north>46</north>
  <east>-107</east>
</latLonBox >

<!-- as used in dataset source -->
<dataSetLocation xsi:type="LatLonPointType" srs="EPSG:4326">
  <latitude>35.64722220</latitude >
  <longitude>-78.40527780</longitude >
</dataSetLocation>
```

7.3.7 Notes on compatibility with OGC specifications

7.3.7.1 The <geogLocation> element:

A fairly simple change would align this element with GML best practices as used in the OGC Point Profile, GeoRSS GML, GML OASIS Profile, and the OGC GML IETF GeoShape Best Practices document. Following these specifications, <geogLocation> can be transformed from:

```
<geoLocation>
  <geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
    <latitude>40.83040556</latitude>
    <longitude>-116.5883417</longitude>
  </geogLocation>
</geoLocation>
```

Into:

```
<geoLocation>
  <gml:Point srsName="urn:ogc:def:crs:EPSG:6.6:4269>"
    <gml:pos>40.83040556 -116.5883417</gml:pos>
  </gml:Point>
</geoLocation>
```

Similarly, other OGC shapes (as defined, for example, in the GML 3.1.1 PIDF-LO Shape Application Schema document, 06-142) can be encoded inside the <geoLocation> element, e.g. for 2D polygon:

```
<gml:Polygon srsName="urn:ogc:def:crs:EPSG::4326"
xmlns:gml="http://www.opengis.net/gml">
  <gml:exterior>
    <gml:LinearRing>
      <gml:pos>42.556844 -73.248157</gml:pos>
      <gml:pos>42.549631 -73.237283</gml:pos>
      <gml:pos>42.539087 -73.240328</gml:pos>
      <gml:pos>42.535756 -73.254242</gml:pos>
      <gml:pos>42.542969 -73.265115</gml:pos>
      <gml:pos>42.553513 -73.262075</gml:pos>
      <gml:pos>42.556844 -73.248157</gml:pos>
    </gml:LinearRing>
  </gml:exterior>
</gml:Polygon>
```

The specification of latLonBoxType can also be changed to follow GML's Envelope element, e.g.:

```
<gml:Envelope srsName="urn:x-ogc:def:CRS:EPSG:27354">
  <gml:lowerCorner>-73.933217 40.78587</gml:lowerCorner>
  <gml:upperCorner>-73.768722 40.914404</gml:upperCorner>
</gml:Envelope>
```

Note that this in the above example the inclusion of srsName is mandatory because it specifies a different sequence of coordinate axes than in the default (here, longitude is first and latitude is second). The srsName may be omitted which would assume the default WGS-84 2D SRS (where the first value is latitude).

Alternately (and this is the flavor adopted in WaterML), the template can be borrowed from ISO 19115, as:

```
<westBoundLongitude>...</westBoundLongitude>
<eastBoundLongitude>...</eastBoundLongitude>
<southBoundLatitude>...</southBoundLatitude>
<northBoundLatitude>...</northBoundLatitude>
```

7.3.7.2 Datums

OGC's Datums schema is at http://schemas.opengeospatial.net/gml/3.1.1/base/datums.xsd. We expect to align both horizontal and vertical datum description to the schema.

7.4 Elements dealing with variables

7.4.1 General description

WaterML categorizes information about variables in two ways: information about the variable in general, and period of record information about the variable as observed at a given site. Information about the variable in general is given by an element of the <VariableInfoType> type. This XML type defines a number of possible child elements including, but not limited to:

- <variableCode> - Identifier for the variable within an observation network, e.g. "00010"
- <variableName> - Name of the variable, e.g. "Water Temperature"
- <sampleMedium> - E.g. "Soil"
- <units> - An element with child elements giving information about the units, such as the units abbreviation (e.g. "cfs") and units type (e.g. "length")

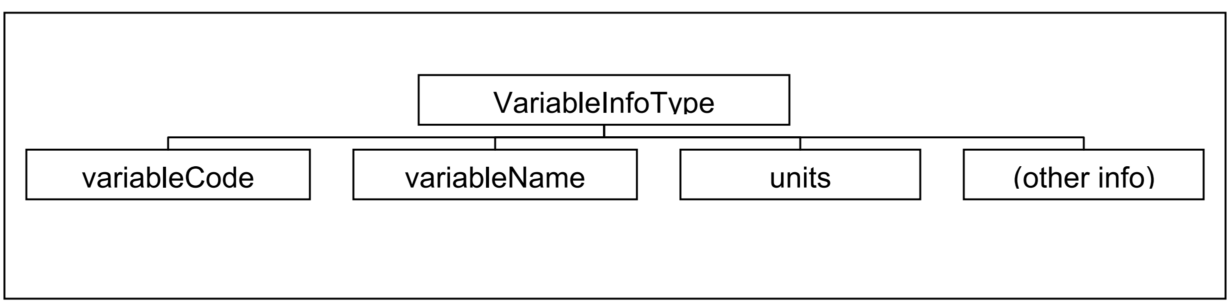

Figure 3. Conceptual Diagram of Elements Defining Variables in WaterML

While it cannot be guaranteed that the variableCode is unique, the combination of variableCode, sampleMedium, units, and perhaps other properties of the variable is shown to uniquely identify a variable in the federal observations data repositories we have worked with. A <UniqueVariableReference> will be an element constructed from several variable properties, in a way that is specific for an observation network, with the purpose to be a unique URN as is common in OGC specifications.

7.4.2 The variable element

7.4.2.1 Annotated structure

```
variable = VariableInfoType

variable = {
  element variableCode {
   attribute variableID {xsd:int}?,
   attribute vocabulary {string},
   attribute network {string}?,
   attribute default {boolean}?,
   }+,
  element variableName {string},
  element variableDescription {string}?,
  element valueType {ValueTypeEnumeration}?,
  element sampleMedium {SampleMediumEnumeration}?,
  element generalCategory {GeneralCategoryEnumeration}?,
  element units {UnitsType},
  element isRegular {Boolean},
  element dataType {DataTypeEnumeration}?,
  element timeSupport {float}?,
  element timeUnits {UnitsType}?,
  element extension {any}?
  element property {xlink}
 }
```

Notes:

- Element <variable> contains the descriptive information about a variable that has been observed.
- The <variableName> element should be an abbreviated name, and not contain <units> information, if possible.
- More detail about a variable can be contained in the <variableDescription> element (Note: this element does not have an equivalent field in the ODM).
- The attributes @vocabulary is for reference to the source of the variableCode. If there are multiple <variableCode> elements, then an attribute of @default=”true”

should be on the element which is the code that should be used to refer to the variable. Often, a network can be associated with a <variableCode>, in which case, the @network attribute can be used. Multiple <variableCode> elements are supported to allow direct mapping of variableCodes between different vocabularies.

- Elements <valueType>, <generalCategory>, <sampleMedium> are controlled vocabularies, as defined by the Observations database (Annex A)

- Extension elements can contain any XML content, at the discretion of data providers.

7.4.2.2 Example

```
<variable>
  <variableCode network="NWIS" vocabulary="NWIS" default="true"
      variableID="#####">00060</variableCode>
  <variableCode vocabulary="CUAHSI"
      variableID="#####">Discharge</variableCode>
  <variableName>Discharge [Time support should be separate element or
attribute, not part of variable name]</variableName>
  <variableDescription>Discharge[units should not be part of variable
description]</variableDescription>
  <!-- controlled vocabulary from OD -->.
  <valueType> </valueType>
  <generalCategory> </generalCategory>
  <sampleMedium> </sampleMedium>
  <units unitsCode="35" unitsAbbreviation="cfs" unitsType="flow">
    cubic feet per second
  </units>
</variable>
```

7.4.3 Units element

7.4.3.1 Annotated structure

```
UnitsType = {
  element units {string,
   attribute unitsID {xsd:int}?,
   attribute unitsCode {string}?,
   attribute unitsAbbreviation {string},
   attribute unitsType {string}
   }
  }
```

Notes:

- The <units> element is used within the variables, and values elements.

- The <values> element in timeseries also uses attributes @unitsCode, and @unitsAbbreviation.

- @unitsAbbreviation is the CUAHSI controlled vocabulary abbreviation for this unit
- @unitsID is an internal identifier.
- @unitsType is the CUAHSI controlled vocabulary for unitsType

7.4.3.2 Example

```
<units unitsID="35" unitsAbbreviation="cfs" unitsType="flow">
  cubic feet per second
</units>
```

7.4.4 Notes on compatibility with OGC specifications

7.4.4.1 Units of measure

GML v 3.1 includes a Units of Measure (UOM) dictionary (http://schemas.opengeospatial.net/gml/3.1.1/base/units.xsd) which may be used as a template for the handling of units in WaterML.

7.4.4.2 Modeling variables.

The O&M discusses feature-centric (ISO 19101 and 19109), coverage-centric (ISO 19123) and observation-centric viewpoints on representing measured properties in spatio-temporal domain. Treatment of variables in WaterML shall be placed in this context, and potentially aligned with the treatment of O&M's observedProperty and its components. At the moment, WaterML's treatment is more prescriptive, compared to O&M where observedProperty is specified as a link to an entity in a dictionary of phenomenon definitions defined by URN, e.g.

```
<om:observedProperty xlink:href="urn:x-ogc:def:phenomenon:OGC:Precipitation"/>
```

This difference in treatment stems from WaterML's intent to align with parameter description and identification practices of the federal hydrologic repositories we worked with.

7.5 Elements dealing with time and measured values

7.5.1 General description

The elements described in this section focus on time, and in particular on the time series of hydrologic observations that a user requests from WaterOneFlow. All of the timestamps, values, and supplemental time series information are stored under a single element of type <TsValuesSingleVariableType>. This element contains one or more <value> elements, as well as optional elements defining such things as the quality control level or qualifiers used.

Each <value> element represents a single time series value. The value is stored as the element's text (the data between the opening and closing tags of the element), while the timestamp at which that value occurred is stored in the @dateTime attribute of <value>. The <value> element may contain other attributes to further qualify the value.

The figure below illustrates that one <TsValuesSingleVariableType> element may have any <value> elements, and that each <value> element has a @dateTime attribute.

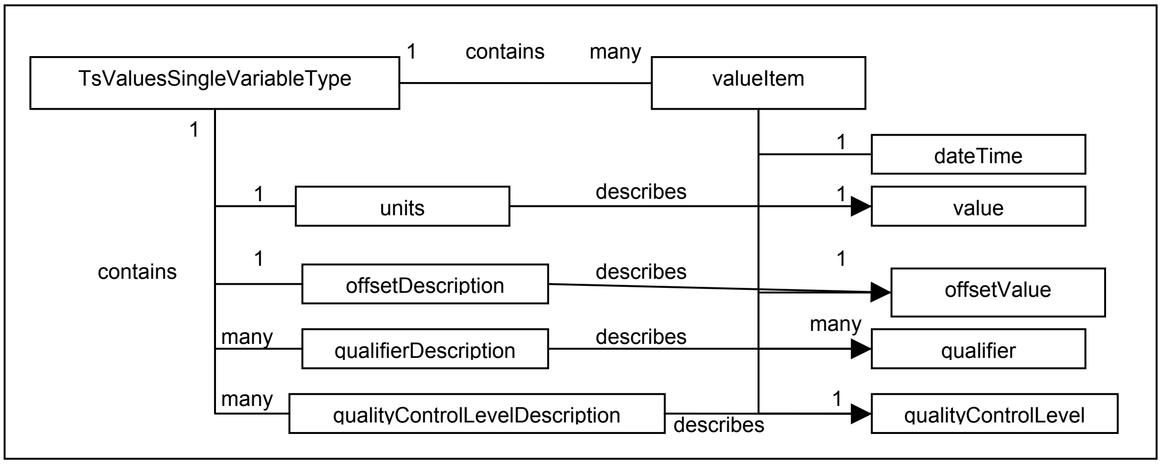

Figure 3. WaterML elements representing a set of values

7.5.2 Values

7.5.2.1 Annotated structure

```
element values {
   element units {string,
     attribute unitsID {xsd:int}?,
     attribute unitsCode {string}?,
     attribute unitsAbbreviation {string},
     attribute unitsType {string}
    }
   element valueItem {
     attribute codedVocabularyTerm {text}?,
     attribute codedVocabulary {text}?,
     element value{xsd:real},
     element dateTime {xsd:DateTime},
     element valueAccuracy {xsd:real}?,
     element censorCode {censorCodeCVEnnumeration}?,
     element qualfierCode {text}*,
     element offsetValue {xsd:decimal}?,
```



```
        element qualityContolLevelID
                     {QualityControlLevelEnumeration}?,
      }*,
      element offsetDescription {
        element units {string,
          attribute unitsID {xsd:int}?,
          attribute unitsCode {string}?,
          attribute unitsAbbreviation {string},
          attribute unitsType {string}
         }
         element description {text}
      }?,

      element qualifierDescription {text,
        attribute qualiferCode
      }*
      element qualityControlDescription {text,
        attribute qualityControlLevelID
      }*
  }
```

Notes:

- Element <values> contains a list of <valueItem> elements that are repeated for the entire series followed by offsetDescription and <qualifierDescription> elements.

- Element <valueItem> contains element <value> that holds the actual data values and elements <dateTime>, <valueAccuracy>, <censorCode>, <qualifier>, <offsetValue>, and <qualityControlLevel> provide value level information about the data value. <dateTime> is required, but the other elements are optional. There may be multiple <qualifier> elements for data values that are qualified by multiple qualifer descriptions.

- Element <units> gives unit information.

  The elements <offsetValue> within <valueItem> and <offsetDescription> within values communicate offset information

- The handling of coded vocabularies for categorical variables has significant limitations in the current version. We have discovered that some services mix text and numeric content in the returned values. In order to accommodate such services, two codedVocabulary attributes are used. If value is a coded vocabulary, then set @codedVocabulary='true', place the vocabulary term in @codeVocabularyTerm, and place a numeric value in the <value>. Often such terms can and should be recoded as either @censorCode, or @qualifiers. If @qualifiers is used, then insert <qualifier> elements at the end of the <values> element

7.5.2.2 Examples

Example 1. Here is a minimum set of elements and attributes. A <values> element with @unitsAbbreviation (drawn from the controlled vocabulary, UnitsTypeEnumeration – see Annex A) .

```
<values>
  <units unitsID="35" unitsAbbreviation="cfs" unitsType="flow">cubic
feet per second</units>
  <valueItem>
    <value>5.6</value>
    <dateTime>1977-04-04T11:45:00</dateTime>
  </valueItem>
  <!-- snip -->
  <valueItem>
    <value>0.38</value>
    <dateTime>1990-08-29T11:45:00</dateTime>
  </valueItem>
  <valueItem>
    <value>2.6</value>
    <dateTime>1991-11-01T13:30:00</dateTime>
  </valueItem>
</values>
```

Example 2. Values returned with <censorCode> and <qualifier> elements.

If a <censorCode> is present, then the data should be examined, and used only is appropriate. Censor codes are:

- "lt" – less than,
- "gt" - greater than,
- "nc" - no code

The two-letter codes are used rather than traditional symbols to ensure that the user creates and decodes the XML messages properly.

```
<values>
  <units unitsID="35" unitsAbbreviation="cfs" unitsType="flow">cubic
feet per second</units>
  <valueItem>
    <value>5.6</value>
    <dateTime>1977-04-04T11:45:00</dateTime>
    <qualifierCode>A</qualifierCode>
  </valueItem>
  <!-- snip -->
  <valueItem>
    <value>0.38</value>
    <dateTime>1990-08-29T11:45:00</dateTime>
    <qualifierCode>A</qualifierCode>
    <qualifierCode>e</qualifierCode>
```

```
    <censorCode>lt</censorCode>
  </valueItem>
  <valueItem>
    <value>2.6</value>
    <dateTime>2007-11-01T13:30:00</dateTime>
    <qualifier>p</qualifier>
  </valueItem>
  <!-- qualifier Description -->
  <qualifierDescription qualifiedCode="A" >Approved.
USGS</qualifierDescription>
  <qualifierDescription qualifierCode="e">Value was estimate.
USGS</qualifierDescription>
  <qualifierDescription qualifierCode="p">Preliminary Value.
USGS</qualifierDescription>
</values>
```

Example 3: Values returned with <offsetValue> and <offsetDescription> elements.

```
<values>
  <units unitsID="35" unitsAbbreviation="cfs" unitsType="flow">cubic
feet per second</units>
  <valueItem>
    <value>5.6</value>
    <dateTime>1977-04-04T11:45:00</dateTime>
    <offsetValue>10</offsetValue>
  </valueItem>
  <!-- snip -->
  <valueItem>
    <value>0.38</value>
    <dateTime>1990-08-29T11:45:00</dateTime>
    <offsetValue>20</offsetValue>
  </valueItem>
  <valueItem>
    <value>2.6</value>
    <dateTime>1991-11-01T13:30:00</dateTime>
    <offsetValue>10</offsetValue>
    <qualifier>p</qualifier>
  </valueItem>
  <offsetDescription>
    <units unitsAbbreviation="m" unitsType="length">
      meters
    </units>
    <description>Depth below surface</description>
  </offsetDescription>
  <qualifierDescription qualifierCode="p">Preliminary Value.
USGS</qualifierDescription>
</values>
```

Example 4: Values returned with <qualityContolLevel> :

```
<values>
  <units unitsID="35" unitsAbbreviation="cfs" unitsType="flow">cubic
feet per second</units>
  <valueItem>
    <value>5.6</value>
```

```
    <dateTime>1977-04-04T11:45:00</dateTime>
    <qualityControlLevel>1</qualityControlLevel>
    <qualifier>A</qualifier>
    <qualifier>e</qualifier>
  </valueItem>
  <!-- snip -->
  <valueItem>
    <value>0.38</value>
    <dateTime>1990-08-29T11:45:00</dateTime>
    <qualityControlLevel>1</qualityControlLevel>
    <qualifier>A</qualifier>
    <qualifier>e</qualifier>
  </valueItem>
  <valueItem>
    <value>2.6</value>
    <dateTime>2007-11-01T13:30:00</dateTime>
    <qualityControlLevel>0</qualityControlLevel>
    <qualifier>p</qualifier>
  </valueItem>
  <qualifierDescription qualifierCode="A" >Approved.
USGS</qualifierDescription>
  <qualifierDescription qualifierCode="e">Value was estimate.
USGS</qualifierDescription>
  <qualifierDescription qualifierCode="p">Preliminary Value.
USGS</qualifierDescription>
  <qualityControlDescription qualityControlLevel="0">Raw
Date</qualityControlDescription>
  <qualityControlDescription qualityControlLevel="1">Quality Controlled
Data</qualityControlDescription>
</values>
```


7.5.3 Elements of TimePeriodType

In treatment of time values, CUAHSI WaterML generally follows GML (http://schemas.opengis.net/gml/3.1.1/base/temporal.xsd). We distinguish between a time range data is available for, a single instant where data was collection, and a floating time range, where the data available for a specified duration counting back from the present. These are common time representations encountered in time series descriptions available from several federal agencies. Two elements designed to be compatible with GML, are used to represent these cases. A base type, TimeIntervalType, has two children, TimePeriodType, and TimeInstantType. TimePeriodType can be used to describe both a time range, and a floating time period. Restricting the elements to those outlined above will simplify client implementation.

7.5.3.1 Annotated structure

```
timePeriod = TimePeriodType

element timePeriod {
  element begin {dateTime,
```

```
  attribute indeterminatePosition {IndeterminatePositionEnum}
},
  element end {dateTime,
   attribute indeterminatePosition  {IndeterminatePositionEnum}
   }
  element timeLength {real,
    attribute unit {string}
 }
}
}

timeInstant = TimeInstantType

element timeInstant {
   element timePosition {dateTime}
}
```

Notes:

- In the databases we examined, data series appear in three different forms: a time range specified by begin time and end time, a single observation specified by a single time stamp, and a floating time range extending backward, from the current date and time, by a specified number of days (the latter being common when referring to real time observations kept for a limited time). The two XML elements used to describe these situations are derived from the parent type TimeIntervalType. They are TimePeriodType (for a time range, and floating real time data), and TimeInstantType (for a single observation).

- Polymorphism is used in the <series> element of <seriesCatalog>. The measurement time interval element *variableTimeInterval* can be describe in two ways:
    - *TimePeriodType* is a time range containing a *begin* and *end*
    - *TimeInstantType* is a single event, containing one element, *timePosition*

- *The TimePeriodType* is flexible, so we can describe real time information with a floating time period.

- The polymorphic type is determined by setting an @xsi:type on <variableTimeInterval>:

    <variableTimeInterval xsi:type="TimePeriodType">

7.5.3.2 Examples

Example 1: XML representation of a time range:

```
<!--Generic -->
<timeInterval>
  <begin>1982-12-09T00:00:00</begin>
  <end>1982-12-09T00:00:00</end>
</timeInterval>
```

```
<!--as used in seriesCatalog -->
<variableTimeInterval xsi:type="TimePeriodType">
  <begin>1982-12-09T00:00:00</begin>
  <end>1982-12-09T00:00:00</end>
</variableTimeInterval>
```

Example 2: XML representation of a single observation:

```
<!-- generic -->
<timeInstant>
  <timePosition>1982-12-09T00:00:00</timePosition >
</timeInstant>
```

```
<!--as used in seriesCatalog -->
<variableTimeInterval xsi:type=" TimeInstantType ">
  <timePosition>1982-12-09T00:00:00</ timePosition >
</variableTimeInterval>
```

Example 3: XML representation of a real time observations series where data are only available for a limited time.

```
<timeInterval>
  <end indeterminatePosition=”now” />
    <timeLength unit="day">-31</timeLength>
</timeInterval>
```

```
<!-- as used in seriesCatalog -->
<variableTimeInterval xsi:type="TimePeriodType">
    <end indeterminatePosition=”now” />
    <timeLength unit="day">-31</timeLength>
</variableTimeInterval>
```

Note. If <site> with a <series> containing the <variableTimeInterval xsi:type="TimePeriodType"> is stored locally or cached, then it will be necessary to recalculate the data availability begin/end date and time.

7.5.4 Notes on compatibility with OGC specifications

Time treatment, for the three types of time specification common in hydrologic data systems, is generally aligned with the GML approaches, sans the syntax.

7.6 Series and series catalogs

7.6.1 General description

The <seriesCatalog> contains a list of unique combinations of site, variable and time intervals that specify a sequence of observations. Multiple <seriesCatalog> elements can be included where multiple dataSeries are available for a site. This treatment is different from the ODM, where data in a single database instance are served via a single web service. For some data providers, the same variable codes are utilized for different services. For example the USGS has a daily values service, where values are for a 24 hour period, and real-time observations, where data is available in 15 minute increments. A common siteCode, and variableCode are used between the data services. Hence inclusion of multiple <seriesCatalog> elements, reflecting series with different time scales or method within the same organization, or from different source organizations, is allowed in WaterML. See the ODM document for a discussion of the support, spacing and extent of observations that define time scale and for how series are identified based on a unique combination of site, variable, method, source, quality control level.

As stated in the ODM documentation, the notion of data series used in WaterML does not distinguish between different series of the same variable at the same site but measured with different offsets. If for example temperature was measured at two different offsets by two different sensors at one site, both sets of data would fall into one data series for the purposes of the series catalog. In these cases, interpretation or analysis software will need to specifically examine and parse the offsets by examining the offset associated with each value. The series catalog does not do this because the principal purpose of the series catalog is data discovery, which we did not want to be overly complicated.

7.6.2 Series

7.6.2.1 Annotated structure

```
element series {
  element variable {VariableInfoType},
  element valueCount {xsd:int
    attribute countIsEstimate {boolean}
  },
  element variableTimeInterval {
    TimeIntervalType|TimeInstantType
  }
  element sampleMedium {string}?,
  element valueType {string}?,
  element generalCategory {string}?,
  element method {MethodType}?,
  element qualityControlLevel {string}?,
```

```
    element source {SourceType}?,
    element property {xlink}
  }
```

Notes:

- A series contains summary information about a set of observations at a site. The observations have a <variable>, and are observed over a time interval specified by <variableTimeInterval>. In addition, they have a count of values, <valueCount> which in some cases may be an estimate, in which case @countIsEstimated="true"
- The relevant use of polymorphism in the <series> element of <seriesCatalog> is described in Clause 7.5.3.1.

7.6.2.2 Examples

Example 1: where element variableTimeInterval = TimeIntervalType

```
element TimePeriodType = {
  element begin {dateTime},
  element end {dateTime}
}
```

```
<series>
  <variable>
    <variableCode vocabulary="NWIS" default="true"
      variableID="7597">00065</variableCode>
    <variableName>Stage</variableName>
    <variableDescription>
      Water level stage.
      USGS Parameter Group:physical property USGS Subgroup:Gage height
    </variableDescription>
    <units unitsAbbreviation="ft" unitsType="length">international
foot</units>
  </variable>
  <valueCount >14237</valueCount>
  <variableTimeInterval xsi:type="TimePeriodType">
    <begin>1967-10-01T00:00:00</begin>
    <end>2006-09-25T00:00:00</end>
  </variableTimeInterval>
</series>
```

Example 2: where element variableTimeInterval = TimeInstantType

```
element TimeInstantType = {
  element timePosition {dateTime}
}
```

```
<series>
  <variable>
```

```
    <variableCode vocabulary="NWIS" default="true"
       variableID="7579">72019</variableCode>
    <variableName>Depth to water</variableName>
    <variableDescription>
      Depth to water below land surface.
      USGS Parameter Group:physical property USGS Subgroup:Depth to
water level
    </variableDescription>
    <units unitsAbbreviation="ft" unitsCode="48">international
foot</units>
  </variable>
  <valueCount >1</valueCount>
  <variableTimeInterval xsi:type="TimeInstantType">
      <timePosition>1972-06-16T00:00:00</timePosition>
  </variableTimeInterval>
</series>
```

Example 3: where the data is real-time, using element variableTimeInterval = TimePeriodType (a subset of TimePeriod applicable to real-time information is shown)

```
timePeriod = TimePeriodType

element timePeriod {
  element end {
    attribute indeterminatePosition
     {IndeterminatePositionEnum}
   }
  element timeLength {real,
    attribute unit {string}
 }
}
}
```

```
<series>
  <variable>
    <variableCode vocabulary="NWIS" default="true"
          variableID="7579">72019</variableCode>
    <variableName>Depth to water</variableName>
    <variableDescription>
      Depth to water below land surface.
      USGS Parameter Group:physical property USGS Subgroup:Depth to
water level
    </variableDescription>
    <units unitsAbbreviation="ft" unitsCode="48">international
foot</units>
  </variable>
  <valueCount countIsEstimated="true">2976</valueCount>
  <variableTimeInterval xsi:type="TimePeriodType">
    <end indeterminatePosition=”now”/>
    <timeLength unit="day">-31</timeLength>
  </variableTimeInterval>
</series>
```

Note. If <site> with a <series> containing the <variableTimeInterval xsi:type="TimePeriodRealTimeType"> is stored locally or cached, the it will be necessary to recalculate the <beginDateTime> and <endDateTime>..

7.6.3 SeriesCatalog

7.6.3.1 Annotated structure

```
element seriesCatalog =
   attribute menuGroupName,
   attribute serviceWSDL,
   element note {string},
   element series {
     element variable {VariableInfoType},
     element valueCount {xsd:int
        attribute countIsEstimate {boolean}
     },
     element variableTimeInterval {
      TimePeriodType|TimeInstantType
     }
   }+
 }
```

Notes:

- <seriesCatalog> is an element within the <site>, which is returned in a GetSiteInfo response.
- The attributes of <seriesCatalog> are intended as hints to applications:
  - @serviceWSDL provides where this service's GetValues method is located. This GetValues method must exactly match the input paramters of the WaterOneFlow web services; location, variable, beginDateTime,endDateTime.
  - @menuGroupName is for the name to be displayed in an HTML select list group.
- Multiple <seriesCatalog> elements are allowed. This is useful when a location uses the same descriptive codes (site and variable) for different data services. Each <seriesCatalog> can contain multiple <series> elements. The details of <series> are discussed below. This is discussed earlier in Clause 7.6.1.

7.6.3.2 Examples

The example below includes two <seriesCatalog> elements, each with one <series>. In the first series, the <valueCount>of 14327 is flagged as estimated by setting the

@countIsEstimated="true". This could be the case if the system being accessed does not directly provide full details of the measured variables.

In the second <seriesCatalog> no @countIsEstimated is seen. This means that this is an exact count.

The polymorphic character of variableTimeInterval is also demonstrated. The first series has an @xsi:type= TimePeriodType", and represents a range. The second series has an @xsi:type="TimeInstantType" because only a single measurement has been observed for that variable.


```
<seriesCatalog menuGroupName="USGS Daily Values"
   serviceWsdl="http://water.sdsc.edu/waterOneFlow/NWISDV/Service.asmx
   ">
  <note
    type="sourceUrl">http://waterdata.usgs.gov/nwis/dv?[snip]</note>
  <series>
    <variable>
      <variableCode vocabulary="NWIS" default="true"
        variableID="7597">00065</variableCode>
      <variableName>Stage</variableName>
      <variableDescription>
        Water level stage.
        USGS Parameter Group:physical property USGS Subgroup:Gage
height
      </variableDescription>
      <units unitsAbbreviation="ft" unitsCode="48">international
foot</units>
    </variable>
    <valueCount countIsEstimated="true">14237</valueCount>
    <variableTimeInterval xsi:type="TimePeriodType">
      <begin>1967-10-01T00:00:00</begin>
      <end>2006-09-25T00:00:00</end>
    </variableTimeInterval>
  </series>
</seriesCatalog>
<!-- NOTE series catalog is repeatable -->
<seriesCatalog menuGroupName="USGS Instantaneous Irregular Data"
  serviceWsdl="http://water.sdsc.edu/waterOneFlow/NWISIID/Service.asmx">
  <series>
    <variable>
      <variableCode vocabulary="NWIS" default="true"
         variableID="1369">01056</variableCode>
      <variableName>Manganese, , filtered</variableName>
      <variableDescription>
        Manganese,Manganese concentration in filtered water.
        USGS Parameter Group:minor and trace inorganics USGS
         Subgroup:Manganese
      </variableDescription>
      <units unitsAbbreviation="mg/L" unitsCode="199">milligrams per
liter</units>
      <note type="nwis:ParameterDescription">Manganese, water,
filtered, micrograms per liter</note>
    </variable>
```

```
    <valueCount>1</valueCount>
    <variableTimeInterval xsi:type="TimeInstantType">
      <timePosition>1972-06-16T00:00:00</timePosition>
    </variableTimeInterval>
  </series>
</seriesCatalog>
```

7.6.4 Notes on compatibility with OGC specifications

O&M discusses discrete time coverages as a model for time series measured at point locations such as monitoring stations. Consider Listing 33 from the O&M specification:

```
<?xml version="1.0" encoding="UTF-8"?>
<om:ObservationCollection gml:id="coll1"
xmlns:om="http://www.opengeospatial.net/om/0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengeospatial.net/om/0.0 ../om.xsd">
  <gml:description>Collection of observations</gml:description>
  <gml:name>Observation Collection 1</gml:name>
  <om:time>
    <gml:TimePeriod gml:id="op1t">
      <gml:beginPosition>2005-01-11T17:22:25.00</gml:beginPosition>
      <gml:endPosition>2005-01-11T17:24:25.00</gml:endPosition>
    </gml:TimePeriod>
  </om:time>
  <om:member>
    <om:Observation gml:id="o1">
      <om:time>
        <gml:TimeInstant gml:id="ot1t">
          <gml:timePosition>2005-01-11T17:22:25.00</gml:timePosition>
        </gml:TimeInstant>
      </om:time>
      <om:procedure xlink:href="urn:x-
ogc:object:feature:Sensor:OGC:scales"/>
      <om:observedProperty xlink:href="urn:x-
ogc:def:phenomenon:OGC:mass"/>
      <om:featureOfInterest
xlink:href="http://some.interested.org/vegetables/subiMarkets/banana1"/
>
      <om:result xsi:type="gml:MeasureType" uom="urn:x-
ogc:def:uom:OGC:kg">0.28</om:result>
    </om:Observation>
  </om:member>
  <om:member>
    <om:Observation gml:id="o2">
      <om:time>
        <gml:TimeInstant gml:id="ot2t">
          <gml:timePosition>2005-01-11T17:24:25.00</gml:timePosition>
        </gml:TimeInstant>
      </om:time>
      <om:procedure xlink:href="urn:x-
ogc:object:feature:Sensor:OGC:scales"/>
```

```
      <om:observedProperty xlink:href="urn:x-
ogc:def:phenomenon:OGC:mass"/>
      <om:featureOfInterest
xlink:href="http://some.interested.org/vegetables/subiMarkets/banana2"/
>
      <om:result xsi:type="gml:MeasureType" uom="urn:x-
ogc:def:uom:OGC:kg">0.27</om:result>
    </om:Observation>
  </om:member>
</om:ObservationCollection>
```

Alternately, CompactDiscreteTimeCoverage format can be used for returning observation results, as, for example, in Listing 43 of O&M. This structure is similar to WaterML:

```
...
<om:result>
  <swe:CompactDiscreteTimeCoverage>
    <swe:element>
      <swe:CompactTimeValuePair>
        <swe:geometry>2005-06-17T09:00+08:00</swe:geometry>
        <swe:value xsi:type="gml:MeasureType" uom="mm">10.1</swe:value>
      </swe:CompactTimeValuePair>
    </swe:element>
    <swe:element>
      <swe:CompactTimeValuePair>
        <swe:geometry>2005-06-18T09:00+08:00</swe:geometry>
        <swe:value xsi:type="gml:MeasureType" uom="mm">15.7</swe:value>
      </swe:CompactTimeValuePair>
    </swe:element>
   <!-- . . . -->
  </swe:CompactDiscreteTimeCoverage>
</om:result>
...
```

O&M allows for specifying the result as a data stream, or, as in this case, as an observation collection. Note that the values of the procedure, observedProperty and featureOfInterest are all given as URN references. For implementation efficiency, WaterML includes additional variable properties to ensure that a variable is uniquely identified in a repository, and one web service call returns sufficient information for common clients. Also, WaterML accommodates different variable vocabularies and codes used across repositories. Schemas suggested in O&M shall be tested for efficiency and completeness against the USGS, EPA and NCDC repositories, to decide on adjustments.

7.7 Elements dealing with web method queries

7.7.1 General description

In addition to elements describing hydrologic information, WaterML also defines elements which keep track of the queries that the user made to the WaterOneFlow web service. This provides a means of quality control, so that the user can check to see which inputs a given web method actually received from the client application. This information is stored in an element of the <QueryInfoType> type. For example, if the client asked for information about site "147", the element would return information essentially saying, "you have requested information about site 147". All of the parameters that the user sent to the web service are stored in a child element of <QueryInfoType> called <criteria>.

In some cases, a WaterOneFlow web service retrieves information from a data source by navigating to a single URL, and then parsing the information that is returned from that URL. When this scenario occurs, the service may return the URL that it used to retrieve the information. This provides another level of quality control. If the client does not receive the information it expects from the web service, it can navigate to the URL directly to see what information is being returned from the original data source, before being reformatted into WaterML by the web service. When present, the URL is stored in an element named <queryURL>.

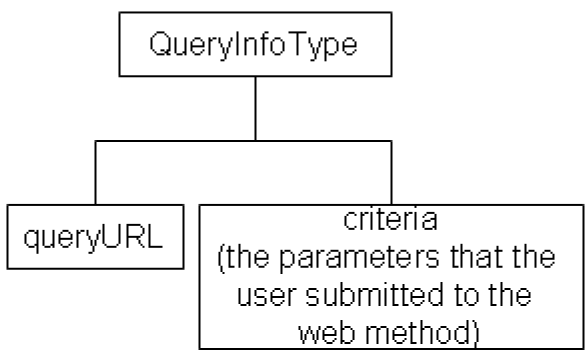

Figure 4. Conceptual Diagram of Elements Dealing with Web Method Queries in WaterML

The GetSiteInfo, GetVariableInfo and GetValues methods of WaterOneFlow services, return, respectively, documents of SiteInfoResponse, VariableResponse, and TimeSeriesResponse types. Each of the response types includes the queryInfo element, and the information about sites, variables, and time series respectively. The returned content is described in the following clauses:

- For GetSiteInfo: Clauses 7.3.2 (siteInfo) and 7.6.3 (seriesCatalog)
- For GetVariableInfo: Clause 7.4.2 (variableInfo)
- For getValues: Clause 7.6.2 (series)

The three basic response types are described below

7.7.2 SiteInfoResponse Type

7.7.2.1 Annotated structure

The GetSiteInfo method returns a WaterML element called <sitesResponse> of the <SiteInfoResponseType> type. This element includes information about a site, such as site name and location, and also a catalog of the variables that are measured at the site.

The <SiteInfoResponseType> element contains a <queryInfo> element of type <QueryInfoType>, and a <site> element. The <site> element contains a <siteInfo> element of type <SiteInfoType> which gives the basic information about a site such as name and location, and a <seriesCatalog> element that lists the variables measured at the site.

The <seriesCatalog> element contains one or more <series> elements, where each is associated with a single variable at a site. Within the <series> element is a <variable> element of type <VariableInfoType>, and <variableTimeInterval> element of type <TimePeriodType>. Other elements may also be present to further qualify the series.

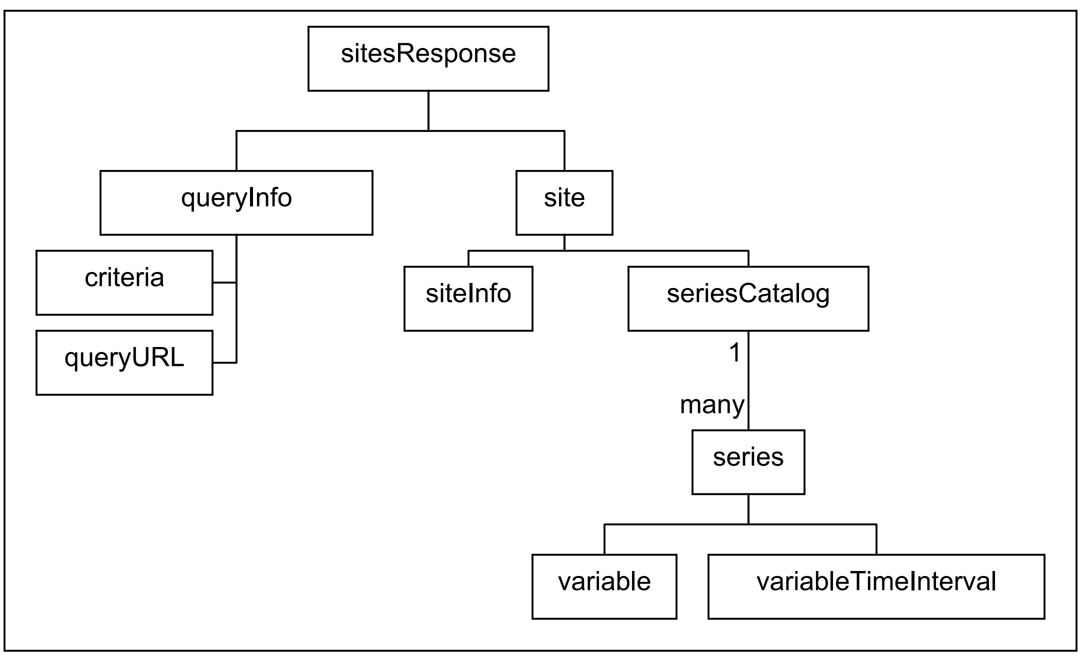

Figure 5. WaterML sitesReponse

```
sitesResponse = {
  element queryInfo {}?,
```

```
element sites {
 element siteInfo {SiteInfoType},
 element seriesCatalog {seriesCatalogRecord+}*
 }
}
```

Notes:

- The <siteResponse> is returned in two different API methods: GetSiteInfo, GetSites.
- Element <sites>, which is return of a GetSiteInfo response, contains two parts: a <siteInfo> element, and <seriesCatalog>. The content of element <site> is dependent on the API method called as discussed in Clause 7.3.4.1
- While there is presently no method of returning multiple sites in a GetSiteInfo method call, WaterML allows for multiple sites to be returned.

7.7.2.2 Example


```xml
<sitesResponse xmlns:gml="http://www.opengis.net/gml"
xmlns:xlink="http://www.w3.org/1999/xlink "
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance "
xmlns:wtr="http://www.cuahsi.org/waterML/"
xmlns="http://www.cuahsi.org/waterML/1.0/">
   <queryInfo>
    <criteria>
      <locationParam>NWIS:10263500</locationParam>
    </criteria>
  </queryInfo>
   <site>
     <siteInfo>
      <siteName>BIG ROCK C NR VALYERMO CA</siteName>
      <siteCode network="NWIS" siteID="4622637">10263500</siteCode>
       <geoLocation>
         <geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
          <latitude>34.42083115</latitude>
          <longitude>-117.8395072</longitude>
        </geogLocation>
      </geoLocation>
    </siteInfo>
     <seriesCatalog menuGroupName="USGS Daily Values" serviceWsdl="
http://water.sdsc.edu/WaterOneFlowDev/DailyValues.asmx">
      <note type="sourceUrl">
http://waterdata.usgs.gov/nwis/dv[snip]&begin_date=2006-12-
09&site_no=10263500</note>
       <series>
```



```
          <variable>
           <variableCode vocabulary="NWIS" default="true"
              variableID="12578">00060</variableCode>
           <variableName>Discharge, cubic feet per second</variableName>
           <units unitsAbbreviation="cfs" unitsCode="35">cubic feet per
second</units>
        </variable>
        <valueCount countIsEstimated="true">30563</valueCount>
         <variableTimeInterval xsi:type="TimePeriodType">
           <begin>1923-02-01T00:00:00</begin>
           <end>2006-10-07T00:00:00</end>
        </variableTimeInterval>
      </series>
    </seriesCatalog>
    <seriesCatalog menuGroupName="USGS Unit Values"
    serviceWsdl="http://water.sdsc.edu/WaterOneFlowDev/UnitValues.asmx">
      <note type="sourceUrl">
http://waterdata.usgs.gov/nwis/uv?format=rdb[snip]&begin_date=2006-12-
09&site_no=10263500</note>
      <series>
         <variable>
           <variableCode vocabulary="NWIS" default="true"
               variableID="12582">00065</variableCode>
           <variableName>Gage height, feet</variableName>
           <units unitsAbbreviation="ft" unitsCode="48">international
foot</units>
        </variable>
        <valueCount countIsEstimated="true">2976</valueCount>
         <variableTimeInterval xsi:type="TimePeriodType">
           <end indeterminatePosition="now"/>
           <timeLength unit="day">-31</timeLength>
        </variableTimeInterval>
      </series>
       <series>
         <variable>
           <variableCode vocabulary="NWIS" default="true"
               variableID="12578">00060</variableCode>
           <variableName>Discharge, cubic feet per second</variableName>
           <units unitsAbbreviation="cfs" unitsCode="35">cubic feet per
second</units>
        </variable>
        <valueCount countIsEstimated="true">2976</valueCount>
         <variableTimeInterval xsi:type="TimePeriodType">
           <end indeterminatePosition="now"/>
           <timeLength unit="day">-31</timeLength>
        </variableTimeInterval>
      </series>
    </seriesCatalog>
  </site>
</sitesResponse>
```

In the example above, information is derived from multiple sources. A note[@type='sourceUrl'] is used to convey the information about the original source of series information.

7.7.3 VariablesResponse Type

7.7.3.1 Annotated structure

The GetVariableInfo method returns a WaterML element called <variablesResponse> of the <VariablesResponseType> type. This element includes information about a variable, such as the name of the variable and its units of measure.

The <variablesResponse> element contains a <variables> element, which contains one or more <variable> elements which are of the <VariableInfoType> type. These <variable> elements are the same building blocks used as the <variable> elements that are returned as part of the SiteInfoResponseType.

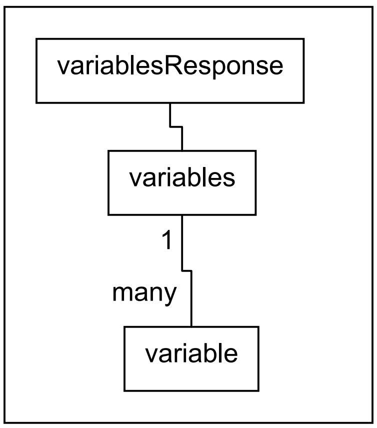

Figure 6. WaterML variablesResponse

```
variablesResponse = {
  element queryInfo { }?,
  element variables {
     element variable {VariableInfoType}+
  }
```

Notes:

- A <variablesResponse> is returned in response to a GetVariables method call.
- If no parameters are passed to GetVariables, then all variables for a given service are returned.
- Note: <variablesResponse> may contain more than one variable returned on a single GetVariables call even though a single variable code was requested. This occurs when a service has multiple medium, time intervals, or other variable characteristics.

7.7.3.2 Example

```
<variablesResponse xmlns:gml="http://www.opengis.net/gml"
xmlns:xlink="http://www.w3.org/1999/xlink "
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance "
xmlns:wtr="http://www.cuahsi.org/waterML/"
xmlns="http://www.cuahsi.org/waterML/1.0/">
  <variable>
    <variableCode vocabulary="NWIS" default="true"
       variableID="1369">01056</variableCode>
    <variableName>water</variableName>
    <variableDescription>water. USGS Parameter Group:minor and trace
inorganics USGS Subgroup:Manganese</variableDescription>
    <units unitsAbbreviation="mg/L" unitsCode="199">milligrams per
liter</units>
    <note type="nwis:ParameterDescription">Manganese, water, filtered,
micrograms per liter</note>
  </variable>
</variablesResponse>
```

7.7.4 TimeSeriesResponse Type

The GetValues method returns a WaterML element called <timeSeriesResponse> of the <TimeSeriesResponseType> type. This element includes a time series of values for a given variable at a given site, as well as information about the variable and the site.

The <timeSeriesResponse> element contains a <queryInfo> element of type <QueryInfoType>, and a <timeSeries> element of type <TimeSeriesType>. The <queryInfo> element serves the same purpose as in the <SiteInfoResponseType> element.

The <timeSeries> element contains three child elements: <sourceInfo> of type <SourceInfoType>, <variable> of type <VariableInfoType>, and <values> of type <TsValuesSingleVariableType>. Each of these XML types were described above as building blocks of WaterML. The <sourceInfo> element provides information about the location to which the time series values apply. The <variable> element provides information about the variable observed, such as units and name. The <values> element contains the time series consisting of datetimes and values.

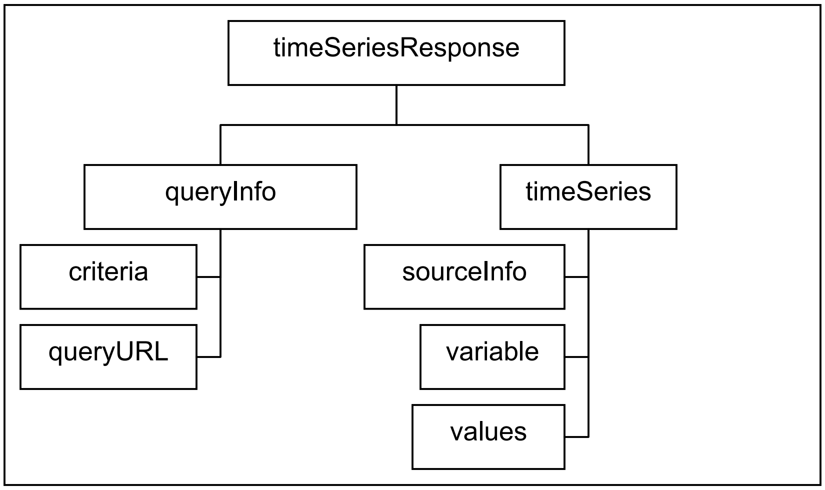

Figure 7. WaterML timeSeriesResponse

```
TimeSeriesRepsonse = {
  element queryInfo { }?,
  element timeSeries {
    element note { }+,
    element sourceInfo {siteInfoType | datasetInfoType }
    element variable {VariableInfoType},
    element values {
    element value {
      attribute dateTime {xsd:dateTime}
      attribute censorCode {CensorCodeEnumeration},
      attribute qualifiers {string},
      attribute offsetValue {double},
      attribute offestUnitsAbbreviation {string},
    }+,
    element qualifier {
      attribute qualifierCode {string},
      attribute qualifierID {xsd:int},
      attribute vocabulary {string}
    }+
  }+
}
```

Notes:

- A call to GetValues returns a <*timeSeriesResponse*>. This is a self-contained call, i.e. not prior or subsequent calls to a web service are needed to utilize the information. Essential variable, source (site or dataset), and values information is returned on this call.

- In <timeSeries>, polymorphism is used on the element <sourceInfo>. The <sourceInfo> element will have an xsi:type=”SiteInfoType” or xsi:type=”datasetInfoType”. If a datasource is site based, then it should return xsi:type=”SiteInfoType”. Examples of site-based services are the CUAHSI ODM web services, USGS NWIS, EPA STORET, and NCDC ASOS. If a dataset is used to generate the time series, then xsi:type=”datasetInfoType” should be used. Examples of such latter type of services are DAYMET, and MODIS.

- Populating <queryInfo> is encouraged though not required. Using <queryInfo allows users to re-examine the source of information if needed. Often, a single URL is not sufficient to define a source of returned data completely. In this case, additional <note> elements with @type=”sourceUrl” are suggested.

- Elements <note> are allowed at the top of the <timeSeries>. This makes <note> elements more visible that if they were at the end of the message, past the <values> element. Elements <note> are useful for other content, such as “all values are preliminary” (note: services should also flag such preliminary <value>’s with @qualifiers and <qualifiers> elements).

- While WaterML allows for multiple <timeSeries> to be returned in a single response, there are no API methods defined for such responses.

7.7.4.2 Example

```
<timeSeriesResponse xmlns:gml="http://www.opengis.net/gml"
xmlns:xlink="http://www.w3.org/1999/xlink "
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance "
xmlns:wtr="http://www.cuahsi.org/waterML/"
xmlns="http://www.cuahsi.org/waterML/1.0/">
  <queryInfo>
    <queryURL>http://nwis.waterdata.usgs.gov/nwis/qwdata?&site_no=015783
    10&parameter_cd=00530&format=rdb[snip] </queryURL>
    <criteria>
      <locationParam>NWIS:01578310</locationParam>
      <variableParam>NWIS:00530</variableParam>
      <timeParam>
        <beginDateTime>2001-01-01T00:00:00</beginDateTime>
        <endDateTime>2001-12-31T00:00:00</endDateTime>
      </timeParam>
    </criteria>
  </queryInfo>
  <timeSeries>
    <sourceInfo xsi:type="SiteInfoType">
      <siteName>SUSQUEHANNA RIVER AT CONOWINGO, MD</siteName>
      <siteCode siteID="4605410">01578310</siteCode>
      <geoLocation>
        <geogLocation xsi:type="LatLonPointType" srs="EPSG:4269">
          <latitude>39.65732996</latitude>
          <longitude>-76.1749532</longitude>
        </geogLocation>
      </geoLocation>
    </sourceInfo>
    <variable>
```

```
      <variableCode vocabulary="NWIS" default="true"
         variableID="12658">00530</variableCode>
      <variableName>Residue, total nonfilterable, milligrams per
liter</variableName>
      <units unitsAbbreviation="mg/L" unitsCode="199">milligrams per
liter</units>
    </variable>
    <values count="28">
      <value censorCode="lt" dateTime="2001-01-03T11:45:00">10</value>
      <!-- snip -->
      <value dateTime="2001-05-01T11:30:00">12</value>
      <value censorCode="lt" dateTime="2001-05-16T08:45:00">10</value>
      <value censorCode="lt" dateTime="2001-06-12T09:00:00">10</value>
      <value dateTime="2001-06-27T09:20:00">16</value>
      <value dateTime="2001-11-07T11:15:00">42</value>
      <value dateTime="2001-12-11T09:45:00">20</value>
      <!-- needs to have qualifier elements added -->
    </values>
  </timeSeries>
</timeSeriesResponse>
```

7.7.5 QueryInfo Element

7.7.5.1 Annotated structure

```
  element queryInfo {
    element creationTime{xsd:dateTime},
    element queryURL,
    element criteria {
      element locationParam,
      element variableParam,
      element timeParam {
        element beginDateTime {xsd:DateTime}?,
        element endDateTime {xsd:DateTime}?
      }
    },
    element note{}*
}
```

Notes:

- Each GetValues response includes additional information about the sources of the information. This is called the <queryInfo> block.
- If the service is scraping a web site, the queryURL should be supplied, so that users can go back to the information source.
- The <beginDateTime> and <endDateTime> need not be specified.

7.7.5.2 Example

```
<queryInfo xmlns="http://www.cuahsi.org/waterML/1.0/">
  <creationTime>2007-04-04T00:00:00</creationTime>
  <queryURL>
    http://nwis.waterdata.usgs.gov/nwis/qwdata?
    &site_no=10324500&parameter_cd=00061&format=rdb&date_format=
    YYYY-MM-DD&begin_date=1977-04-04&end_date=1991-11-01
  </queryURL>
  <criteria>
    <locationParam>NWIS:10324500</locationParam>
    <variableParam>NWIS:00061</variableParam>
    <timeParam>
      <beginDateTime>1977-04-04T00:00:00</beginDateTime>
      <endDateTime>1991-11-01T00:00:00</endDateTime>
    </timeParam>
  </criteria>
  <note type="example">Notes are repeatable and can be used to store
information that is not in the schema</note>
</queryInfo>
```

8 Limitations, and future work

8.1 Multiple siteCodes and variableCodes

It is possible that site and variable codes change over time, or the same site is common between several observation networks. The present WaterOneFlow methods are inflexible, due to web service limitations. Basically, you cannot overload web service methods. In order to accommodate this, we will add query methods in the upcoming WaterOneFlow web services. This will require extensions to WaterML in order to support the submittal of query information. We expect that this will be based on the OGC filter specification. These changes will allow for spatial and temporal query capabilities, and retrieval of information by code or internal ID, and retrieval of multiple site and time series results.

8.2 Categorical Values

If values are all categorical, then it is expected that web services developers should reformat these values to integers, flag the <valueItems> with codedVocabulary attributes, and utilize attribute @codedVocabularyTerm.

Often, real world hydrologic datasets contain mixed-typed values, i.e. one may encounter both numeric and text content in the “values” field (e.g. “1.23”, “no data”, “censored”, “below detection limit”, “less than 10”, “between 5 and 6”, etc.). Most often, text strings in otherwise numeric columns are used to flag a value that is censored. At the moment, WaterML does not handle a mix of numeric and text values in <value>, nor does it handle a mix of categorical and numerical values. If text values communicate data that is censored, or qualified, then web service providers will need to determine what information (@censorCode, <qualityControlLevel> or <qualifier>) a value should be tagged with. If text in a "value" communicates a missing value or a null value, then we suggest that an empty <value/> element be used, with an appropriate qualifier. This may break clients, since nullable primitive types are not supported in some programming languages. The <value> should be a numeric value that is included in local Observations Database following the ODM format, or consistent across the service.

8.3 Adding support for groups

At present, the notion of grouping does not apply to web service messaging. Responses only return information for a single variable. Multi-variable responses are considered for the next version.

8.4 Terminology

We expect that several elements used in the first version of CUAHSI WaterML will be eventually renamed, to align with terms used in OGC specifications, and to better relay the semantics of hydrologic measurement (e.g. the "dataset" element shall be renamed).

8.5 Metadata

Metadata is outside of the scope of a messaging format

ANNEX A (normative)
Controlled Vocabularies (XML Enumerations)

A1 Introduction

Controlled vocabularies for the fields are required to maintain consistency and avoid the use of synonyms that can lead to ambiguity. Controlled vocabularies are implemented as XML schema Enumerations. For the enumerations, we use the standards established for the CUAHSI ODM. The following controlled vocabularies in the ODM are mapped to enumerations.

A2 Censor Code CV: <CensorCodeCVEnumeration>

Term	Definition
(@censored is empty or not present)	not censored
Lt	less than
Gt	greater than
Nc	not censored
Nd	non-detect
pnq	present but not quantified

A3 DataType CV: <DataTypeCVEnumeration>

Term	Definition
Continuous	A quantity specified at a particular instant in time measured with sufficient frequency (small spacing) to be interpreted as a continuous record of the phenomenon.
Sporadic	The phenomenon is sampled at a particular instant in time but with a frequency that is too coarse for interpreting the record as continuous. This would be the case when the

	spacing is significantly larger than the support and the time scale of fluctuation of the phenomenon, such as for example infrequent water quality samples.
Cumulative	The values represent the cumulative value of a variable measured or calculated up to a given instant of time, such as cumulative volume of flow or cumulative precipitation.
Incremental	The values represent the incremental value of a variable over a time interval, such as the incremental volume of flow or incremental precipitation.
Average	The values represent the average over a time interval, such as daily mean discharge or daily mean temperature.
Maximum	The values are the maximum values occurring at some time during a time interval, such as annual maximum discharge or a daily maximum air temperature.
Minimum	The values are the minimum values occurring at some time during a time interval, such as 7-day low flow for a year or the daily minimum temperature.
Constant Over Interval	The values are quantities that can be interpreted as constant over the time interval from the previous measurement.
Categorical	The values are categorical rather than continuous valued quantities. Mapping from Value values to categories is through the CategoryDefinitions table.

A4 General Category CV: <GeneralCategoryCVEnumeration>

Term	Definition
Water Quality	Data associated with water quality variables or processes
Climate	Data associated with the climate, weather, or atmospheric processes
Hydrology	Data associated with hydrologic variables or processes
Biota	Data associated with biological organisms
Geology	Data associated with geology or geological processes

A5 Quality Control Levels CV: <QualityControlLevelEnumeration>

QualityControlLevelID	Definition	Explanation
0	Raw data	Raw data is defined as unprocessed data and data products that have not undergone quality control. Depending on the data type and data transmission system, raw data may be available within seconds or minutes after real-time. Examples include real time precipitation, streamflow and water quality measurements.
1	Quality controlled data	Quality controlled data have passed quality assurance procedures such as routine estimation of timing and sensor calibration or visual inspection and removal of obvious errors. An example is USGS published streamflow records following parsing through USGS quality control procedures.
2	Derived products	Derived products require scientific and technical interpretation and include multiple-sensor data. An example might be basin average precipitation derived from rain gages using an interpolation procedure.
3	Interpreted products	These products require researcher (PI) driven analysis and interpretation, model-based interpretation using other data and/or strong prior assumptions. An example is basin average precipitation derived from the combination of rain gages and radar return data.
4	Knowledge products	These products require researcher (PI) driven scientific interpretation and multidisciplinary data integration and include model-based interpretation using other data and/or strong prior assumptions. An example is percentages of old or new water in a hydrograph inferred from an isotope analysis.

A6 Sample Medium CV: <SampleMediumCVEnumeration>

Term	Definition
Surface Water	Sample taken from surface water such as a stream, river, lake, pond, reservoir, ocean, etc.
Ground Water	Sample taken from water located below the surface of the ground, such as from a well or spring
Sediment	Sample taken from the sediment beneath the water column
Soil	Sample taken from the soil
Air	Sample taken from the atmosphere
Tissue	Sample taken from the tissue of a biological organism
Precipitation	Sample taken from solid or liquid precipitation

A7 Sample Type CV: <SampleTypeCVEnumeration>

Term	Definition
FD	Foliage Digestion
FF	Forest Floor Digestion
FL	Foliage Leaching
LF	Litter Fall Digestion
GW	Groundwater
PB	Precipitation Bulk
PD	Petri Dish (Dry Deposition)
PE	Precipitation Event
PI	Precipitation Increment
PW	Precipitation Weekly
RE	Rock Extraction
SE	Stemflow Event
SR	Standard Reference
SS	Streamwater Suspeneded Sediment
SW	Streamwater
TE	Throughfall Event
TI	Throughfall Increment
TW	Throughfall Weekly
VE	Vadose Water Event
VI	Vadose Water Increment
VW	Vadose Water Weekly
Grab	Grab sample

A8 Topic Category CV: <TopicCategoryCVEnumeration>

Term	Definition
farming	Data associated with agricultural production
Biota	Data associated with biological organisms
boundaries	Data associated with boundaries
climatology/meteorology/atmosphere	Data associated with climatology, meteorology, or the atmosphere
economy	Data associated with the economy
elevation	Data associated with elevation
environment	Data associated with the environment
geoscientificInformation	Data associated with geoscientific information
health	Data associated with health
imageryBaseMapsEarthCover	Data associated with imagery, base maps, or earth cover
intelligenceMilitary	Data associated with intelligence or the military
inlandWaters	Data associated with inland waters
location	Data associated with location
oceans	Data associated with oceans
planningCadastre	Data associated with planning or cadastre
society	Data associated with society
structure	Data associated with structure
transportation	Data associated with transportation
utilitiesCommunication	Data associated with utilities or communication

A9 Units CV: <UnitsCVEnumeration>

UnitsID	UnitsName	UnitsType	UnitsAbbreviation
1	percent	Dimensionless	%
2	degree	Angle	deg
3	grad	Angle	grad
4	radian	Angle	rad
5	degree north	Angle	degN
6	degree south	Angle	degS
7	degree west	Angle	degW
8	degree east	Angle	degE
9	arcminute	Angle	arcmin
10	arcsecond	Angle	arcsec
11	steradian	Angle	sr
12	acre	Area	ac
13	hectare	Area	ha
14	square centimeter	Area	cm2
15	square foot	Area	ft2
16	square kilometer	Area	km2
17	square meter	Area	m2
18	square mile	Area	mi2
19	hertz	Frequency	Hz
20	darcy	Permeability	D
21	british thermal unit	Energy	BTU
22	calorie	Energy	cal
23	erg	Energy	erg
24	foot pound force	Energy	lbf ft
25	joule	Energy	J

26	kilowatt hour	Energy	kW h
27	electronvolt	Energy	eV
28	langleys per day	Energy Flux	Ly/d
29	langleys per minute	Energy Flux	Ly/m
30	langleys per second	Energy Flux	Ly/s
31	megajoules per square meter per day	Energy Flux	MJ/m2 d
32	watts per square centimeter	Energy Flux	W/cm2
33	watts per square meter	Energy Flux	W/m2
34	acre feet per year	Flow	ac ft/yr
35	cubic feet per second	Flow	cfs
36	cubic meters per second	Flow	m3/s
37	cubic meters per day	Flow	m3/d
38	gallons per minute	Flow	gpm
39	liters per second	Flow	l/s
40	million gallons per day	Flow	MGD
41	dyne	Force	dyn
42	kilogram force	Force	kgf
43	newton	Force	N
44	pound force	Force	lbf
45	kilo pound force	Force	kip
46	ounce force	Force	ozf
47	centimeter	Length	cm
48	international foot	Length	ft
49	international inch	Length	in
50	international yard	Length	yd
51	kilometer	Length	km

52	meter	Length	m
53	international mile	Length	mi
54	millimeter	Length	mm
55	micron	Length	um
56	angstrom	Length	Å
57	femtometer	Length	fm
58	nautical mile	Length	nmi
59	lumen	Light	lm
60	lux	Light	lx
61	lambert	Light	La
62	stilb	Light	sb
63	phot	Light	ph
64	langley	Light	Ly
65	gram	Mass	gr
66	kilogram	Mass	kg
67	milligram	Mass	mg
68	microgram	Mass	mg
69	pound mass (avoirdupois)	Mass	lb
70	slug	Mass	slug
71	metric ton	Mass	tonne
72	grain	Mass	grain
73	carat	Mass	car
74	atomic mass unit	Mass	amu
75	short ton	Mass	ton
76	BTU per hour	Power	BTU/hr
77	foot pound force per second	Power	lbf/s

78	horse power (shaft)	Power	hp
79	kilowatt	Power	kW
80	watt	Power	W
81	voltampere	Power	VA
82	atmospheres	Pressure/Stress	atm
83	pascal	Pressure/Stress	Pa
84	inch of mercury	Pressure/Stress	inch Hg
85	inch of water	Pressure/Stress	inch H2O
86	millimeter of mercury	Pressure/Stress	mmHg
87	millimeter of water	Pressure/Stress	mmH2O
88	centimeter of mercury	Pressure/Stress	cmHg
89	centimeter of water	Pressure/Stress	cmH2O
90	millibar	Pressure/Stress	mbar
91	pound force per square inch	Pressure/Stress	psi
92	torr	Pressure/Stress	torr
93	barie	Pressure/Stress	barie
94	meters per pixel	Resolution	
95	meters per meter	Scale	
96	degree celcius	Temperature	degC
97	degree fahrenheit	Temperature	degF
98	degree rankine	Temperature	degR
99	degree kelvin	Temperature	degK
100	second	Time	sec
101	millisecond	Time	millisec
102	minute	Time	min
103	hour	Time	hr

104	day	Time	d
105	week	Time	week
106	month	Time	month
107	common year (365 days)	Time	yr
108	leap year (366 days)	Time	leap yr
109	Julian year (365.25 days)	Time	jul yr
110	Gregorian year (365.2425 days)	Time	greg yr
111	centimeters per hour	Velocity	cm/hr
112	centimeters per second	Velocity	cm/s
113	feet per second	Velocity	ft/s
114	gallons per day per square foot	Velocity	gpd/ft2
115	inches per hour	Velocity	in/hr
116	kilometers per hour	Velocity	km/h
117	meters per day	Velocity	m/d
118	meters per hour	Velocity	m/hr
119	meters per second	Velocity	m/s
120	miles per hour	Velocity	mph
121	millimeters per hour	Velocity	mm/hr
122	nautical mile per hour	Velocity	knot
123	acre foot	Volume	ac ft
124	cubic centimeter	Volume	cc
125	cubic foot	Volume	ft3
126	cubic meter	Volume	m3
127	hectare meter	Volume	hec m
128	liter	Volume	L

129	US gallon	Volume	gal
130	barrel	Volume	bbl
131	pint	Volume	pt
132	bushel	Volume	bu
133	teaspoon	Volume	tsp
134	tablespoon	Volume	tbsp
135	quart	Volume	qrt
136	ounce	Volume	oz
137	dimensionless	Dimensionless	-

A10 Value Type CV: <ValueTypeCVEnumeration>

Term	Definition
Field Observation	Observation of a variable using a field instrument
Sample	Observation that is the result of analyzing a sample in a laboratory
Model Simulation Result	Values generated by a simulation model
Derived Value	Value that is directly derived from an observation or set of observations

A11 Variable Name CV: <VariableNameCVEnumeration>

Term	Term
Nitrogen, nitrate (NO3) nitrogen as NO3	Biochemical oxygen demand, ultimate carbonaceous
Nitrogen, nitrite (NO2) nitrogen as N	Chemical oxygen demand
Nitrogen, nitrite (NO2) nitrogen as NO2	Oxygen, dissolved

Nitrogen, nitrite (NO2) + nitrate (NO3) nitrogen as N	Light attenuation coefficient
Nitrogen, albuminoid	Secchi depth
Nitrogen, gas	Turbidity
Phosphorus, total as P	Color
Phosphorus, total as PO4	Coliform, total
Phosphorus, organic as P	Coliform, fecal
Phosphorus, inorganic as P	Streptococci, fecal
Phosphorus, phosphate (PO4) as P	Escherichia coli
Discharge, daily average	Iron sulphide
Temperature	Iron, ferrous
Gage height	Iron, ferric
Discharge	Molybdenum
Precipitation	Boron
Evaporation	Chloride
Transpiration	Manganese
Evapotranspiration	Zinc
H2O Flux	Copper
CO2 Flux	Calcium as Ca
CO2 Storage Flux	Calcium as CaCO3
Latent Heat Flux	Phosphorus, phosphate (PO4) as PO4
Sensible Heat Flux	Phosphorus, ortophosphate as P
Radiation, total photosynthetically-active	Phosphorus, ortophosphate as PO4
Radiation, incoming photosynthetically-active	Phosphorus, polyphosphate as PO4
Radiation, outgoing photosynthetically-active	Carlson's Trophic State Index
Radiation, net photosynthetically-active	Oxygen, dissolved percent of saturation
Radiation, total shortwave	Alkalinity, carbonate as CaCO3

Radiation, incoming shortwave	Alkalinity, hydroxode as CaCo3
Radiation, outgoing shortwave	Alkalinity, bicarbonate as CaCO3
Radiation, net shortwave	Carbon, suspended inorganic as C
Radiation, incoming longwave	Carbon, suspended organic as C
Radiation, outgoing longwave	Carbon, dissolved inorganic as C
Radiation, net longwave	Carbon, dissolved organic as C
Radiation, incoming UV-A	Carbon, suspened total as C
Radiation, incoming UV-B	Carbon, total as C
Radiation, net	Langelier Index
Wind speed	Silicon as SiO2
Friction velocity	Silicon as Si
Wind direction	Silicate as SiO2
Momentum flux	Silicate as Si
Dew point temperature	Sulfur
Relative humidity	Sulfur dioxide
Water vapor density	Sulfur, pyretic
Vapor pressure deficit	Sulfur, rganic
Barometric pressure	Sulfate as SO4
Snow depth	Sulfate as S
Visibility	Potassium
Sunshine duration	Magnesium
Hardness, total	Carbon, total inorganic as C
Hardness, carbonate	Carbon, total organic as C
Hardness, non-carbonate	Methylmercury
Bicarbonate	Mercury
Carbonate	Lead

Alkalinity, total	Chromium, total
pH	Chromium, hexavalent
Specific conductance	Chromium, trivalent
Salinity	Cadmium
Solids, total	Chlorophyll a
Solids, total Volatile	Chlorophyll b
Solids, total Fixed	Chlorophyll c
Solids, total Dissolved	Chlorophyll (a+b+c)
Solids, volatile Dissolved	Pheophytin
Solids, fixed Dissolved	Nitrogen, ammonia (NH3) as NH3
Solids, total Suspended	Nitrogen, ammonia (NH3) as N
Solids, volatile Suspended	Nitrogen, ammonium (NH4) as NH4
Solids, fixed Suspended	Nitrogen, ammonium (NH4) as N
Biochemical oxygen demand, 5-day	Nitrogen, ammonia (NH3) + ammonium (NH4) as N
Biochemical oxygen demand, 5-day carbonaceous	Nitrogen, ammonia (NH3) + ammonium (NH4) as NH4
Biochemical oxygen demand, 5-day nitrogenous	Nitrogen, organic as N
Biochemical oxygen demand, 20-day	Nitrogen, inorganic as N
Biochemical oxygen demand, 20-day nitrogenous	Nitrogen, total as N
Biochemical oxygen demand, ultimate	Nitrogen, kjeldahl as N
Biochemical oxygen demand, ultimate nitrogenous	Nitrogen, nitrate (NO3) as N

A12 Vertical Datum CV: <VerticalDatumCVEnumeration>

Term	Definition

NAVD88	North American Vertical Datum of 1988
NGVD29	National Geodetic Vertical Datum of 1929
MSL	Mean Sea Level

A13 Spatial Reference Systems

Spatial reference systems specification follows the definitions and the numbering system adopted by EPSG.

Annex B (informative)

The Context of WaterML: CUAHSI HIS Services Oriented Architecture, Web Services, and Related Challenges

B1 Introduction

The CUAHSI HIS system architecture is envisioned as a component of a large scale environmental observatory effort, which emerges as a network of seamlessly integrated data collection, information management, analysis, modeling and engineering endeavors implemented across disciplinary boundaries.

The hydrologic community has already developed a plethora of databases, data analysis and visualization models and tools, including various watershed and flow models and mapping and time series visualization systems. Important data resources are provided by federal agencies and include large observation data repositories such as the USGS's NWIS and NAWQA, the EPA's STORET, etc. The goal of CUAHSI HIS architecture development is to alleviate fragmentation and duplication in these efforts, and create an environment where these different geographically distributed components work in concert to support advanced data intensive hydrology research. This includes providing easy analytical access to the distributed data resources, ability to publish and manage local observational and model data, interface the data with a variety of community models and analysis and visualization codes, and easily "plug" new research codes and tools into analytical workflows.

B2 Design principles: What makes the hydrologic cyberinfrastructure different

Integrating common data handling components being developed in neighbor disciplines, specifically those that support secure access to grid resources, single sign-on authentication/authorization, distributed data management, data publication and search, information integration and knowledge management, makes cross-disciplinary data sharing easier, and lets HIS design team focus on the core services specifically needed by hydrologists. Our experience developing the HIS system architecture brought the following conclusions about the specifics of hydrological cyberinfrastructure, and therefore limits of applicability of techniques adopted in other projects:

1) Hydrology community relies to a large extent on federally-organized data collection network, including measurement stations organized in USGS's NWIS and NAWQA, EPA's STORET, Ameriflux tower network, MODIS and DAYMET datasets, and similar networks. These data are in public domain, and repositories are freely accessible via respective web portals. This has two consequences for the cyberinfrastructure: (1) making access to such repositories simpler, more uniform, and model-driven would directly support research efforts for a large group of hydrologists, as was revealed by a CUAHSI user survey, and

(2) the emphasis on data ownership is relatively weaker in hydrologic analysis as compared with other communities such as especially neuroscience (BIRN) and geologists (GEON). This justifies the focus on common web service interface and a hydrologic data access portal easing access to federal observation network archives, without necessary service authentication as is customary in other portal environments.

2) Hydrologic community appears to be organized, to a larger extent than other geoscience communities, by “natural” boundaries that are regional in extent, specifically by river watershed boundaries. This suggests a “natural” network of relatively autonomous hydrologic data nodes that provide access to locally-collected and curated data resources and applications. Therefore, development, deployment and technology support of such nodes is an important component of creating a networked environment for hydrologic data sharing.
3) From the data perspective there are sub-groups in the community focused on analyzing point time series (and incidentally relying largely on Windows platform) and focused on analyzing remote sensing data and time series (and using Linux/Unix platforms to a larger extent than the first group). Supporting different groups of researchers requires that HIS relies on cross-platform data management services and portals that can be deployed in both environments.
4) Given the focus on water resources, CUAHSI communicates mostly with public sector entities (such as local water authorities and related small engineering firms). This creates a lot of opportunities for partnerships at the local level, and underscores the need for a data access infrastructure supporting such partnerships.
5) As revealed by CUAHSI users survey, the community relies on several common COTS (commercial off-the-shelf) software packages, most importantly Excel, ArcGIS and Matlab. Enabling access to time series repositories from these clients, as well as from such popular coding environments as Fortran and VisualBasic, is an important consideration for HIS architecture.
6) Hydrology is an integrative science, with hydrologic models relying on data inputs from several neighbor disciplines (climate and ocean observations, soils, geomorphology and geology, social and demographic datasets, etc.). Consequently, the HIS infrastructure shall support interoperation with data and processing services being developed in other earth science disciplines, and ideally develop similar formats for handling spatio-temporal information.
7) Different hydrologic analyses may require different representations of space and time. For example, hydrologic time series services may need to expose observations in both local and UTC time: the former is common at large scale watershed-level studies while the latter may be needed for compatibility with climate data services. The same applies to handling spatial locations of hydrologic observations, where multiple types of offsets from hydrologic landmarks are commonly recorded.
8) Large numbers of observation variables, on water quality in particular, available in federal repositories (there are nearly 10,000 variables measured within USGS NWIS only) and often inconsistent semantics of variable and measurement unit descriptions across observation networks make the development of observation data catalogs and knowledge bases indispensable.

One of the main benefits of a cyberinfrastructure is the ability to re-use and integrate data and research resources. However, leveraging existing infrastructure components, we must understand the specific research needs and workflows adopted in the discipline. While necessarily generalized and simplified, the features listed above let us conceptualize HIS architecture components and development strategies as presented in the following sections.

B3 The services model for hydrologic observatory

The CUAHSI Hydrology Information System design follows the open services-oriented architecture model that has been explored and developed in several large-scale federally funded cyberinfrastructure projects. Services-oriented architecture (SOA) relies on a collection of loosely coupled self-contained services that communicate with each other and can be called from multiple clients in a standard fashion. Common benefits associated with SOA include: scalability, security, easier monitoring and auditing; standards-reliance; interoperability across a range of resources; plug-and-play interfaces. Internal service complexity is hidden from service clients, and backend processing is decoupled from client applications. In other words, different types of clients, including Web browser and such desktop applications as Matlab, ArcGIS and Excel, exposed as the primary desktop client environments by the CUAHSI user needs assessment, would be able to access the same service functionality, leading to a more transparent and easier managed system.

B4 Main components of CUAHSI HIS architecture

The core of the HIS services-oriented architecture is a collection of WaterOneFlow SOAP web services, that provide uniform access to multiple repositories of observation data, both remote and locally-stored in ODM.

At the physical level, the infrastructure represents a collection of HIS Servers, and data nodes, that support databases, web services, and several web service clients, both desktop (ArcGIS, Excel, Matlab, etc.) and online (ArcGIS Server-based). A high-level view of this organization is shown in Figure XXX.

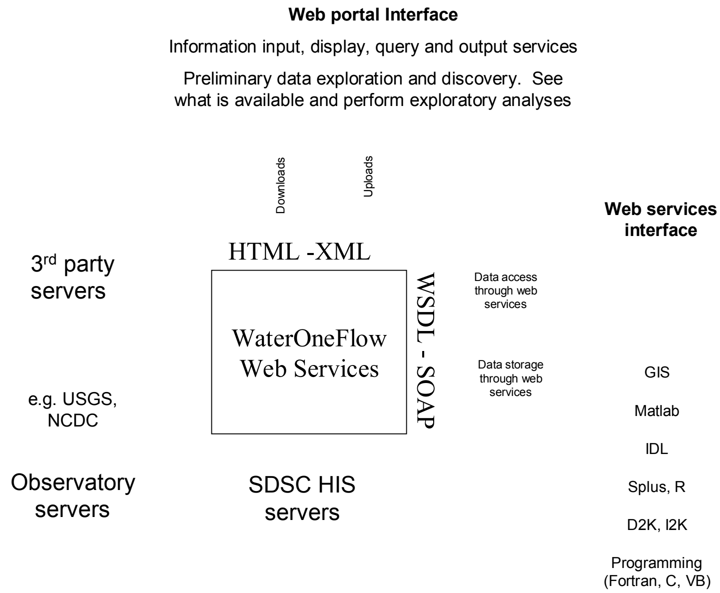

Figure XXX. High-level view of HIS organization

B5 Current status of web service development.

	Delivery		Information (Metadata)						Publication			Discovery		
Data Source	Getvalues	GetValuesObject	GetSiteInfo	GetSiteInfoObject	GetSites	GetSitesXml	GetVariableInfo	GetVariableInfoObject	PutValues	PutVariableInfo	PutSiteInfo	GetSiteList	GetVariables	GetRecordsetWithSQL
USGS NWIS (4 services)	P	P	P	P	P	P	P	P						
DAYMET	P	P					P	P						
MODIS	P	P					P	P						
NAM	P	P					P	P						
EPA Storet	P	P	P	P	P		P	P						
NCDC	D	D	D	D	D	D	D	D				D	D	
CUAHSI ODM	D	D	D	D	D	D	D	D					D	D

Development and testing status is indicated above by the letters following.

P. Provisional. Tested by HIS team and available for evaluation by outsiders on http://water.sdsc.edu/wateroneflow/

D. Development. Undergoing development and testing by the HIS team.

R. Release. Has passed review and released for general use (no services are at the release level)

The shaded boxes above are web service and data set combinations that are not compatible so will not be implemented. Specifically we will not have publication services and record level query capability for third party datasets and do not provide site information for spatial fields not associated with specific sites.

B6 Future work: Outline of web service related tasks

B6.1 Data Publication services for observation data

- Design method signatures, develop and test *PutValues*, *PutSiteInfo, PutVariableInfo* web service methods, for populating ODM and observation data catalogs from various sources, including catalog updates from federal agencies, data from instruments and sensors, researcher-supplied observation data, etc.

- Develop authorization methods for WaterOneFlow services, to enable query access to restricted databases and to support data publication

B6.2 Data Discovery services for observation data

- Attribute-based discovery services, will utilize the semantic mediation work at Drexel and return variables associated with user-entered search terms, and stations where these variables are measured

- Location-based discovery services, returning lists of stations within a particular user-defined region (state, county, hydrologic unit, distance buffer of a linear or point feature, user-defined polygon)

B6.3 Web services for other types of hydrologic and related data

- Develop or adopt web services for publishing and accessing collections of hydrologic vector layers, and incorporate them in HIS Server

- Develop or adopt web services for publishing and accessing climate fields, and incorporate them in HIS Server

- Develop or adopt web services for publishing and accessing remote-sensing data

B6.4 Transformation services

- Develop web services for transformation of hydrologic vocabularies and units

- Enhance existing services with projection, units, time and vocabulary conversion capabilities

Bibliography

[BUTEK] Russell Butek, 2005, *Use polymorphism as an alternative to xsd:choice* retrieved from http://www-128.ibm.com/developerworks/xml/library/ws-tip-xsdchoice.html