OpenGIS Sensor Web Enablement

Tom McCarty
Science Applications International Corporation (SAIC)
Spatial Information Systems Division
August 13, 2003

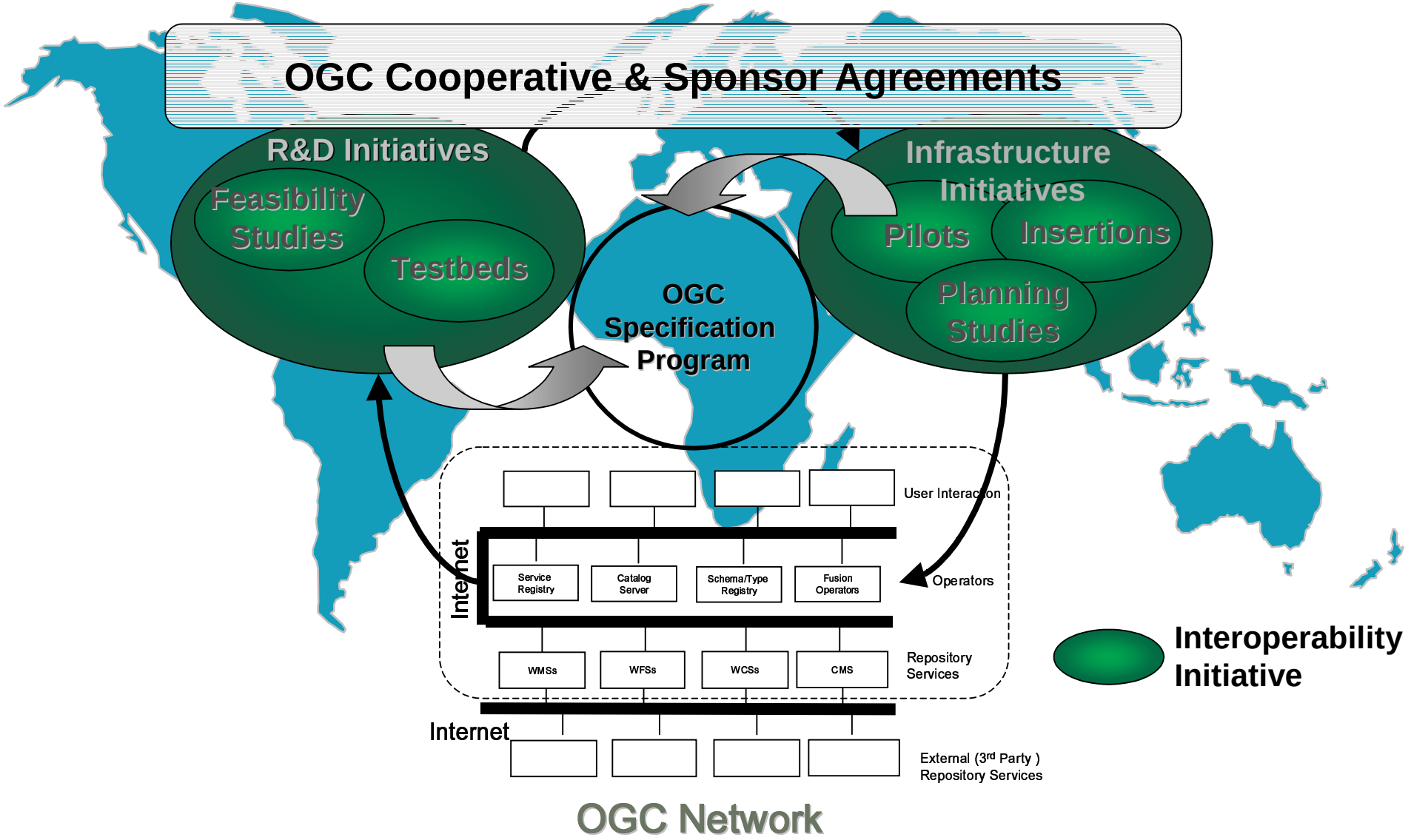
What is OpenGIS?

- Founded in 1994, the OpenGIS Consortium (OGC) is an industry consortium aimed at growing interoperability for technologies involving spatial information.
- OGC members include over 250 public and private companies, universities, government agencies and other organizations from all over the world who are interested in building spatial interoperability.
- OGC originally focused on the use of Distributed Computing Platforms, or DCP's (specifically CORBA, OLE/COM, and SQL).
- The emergence of HTTP and the WWW and the formation of the Web Mapping Testbed redirected OGC into distributed, web service geoprocessing model

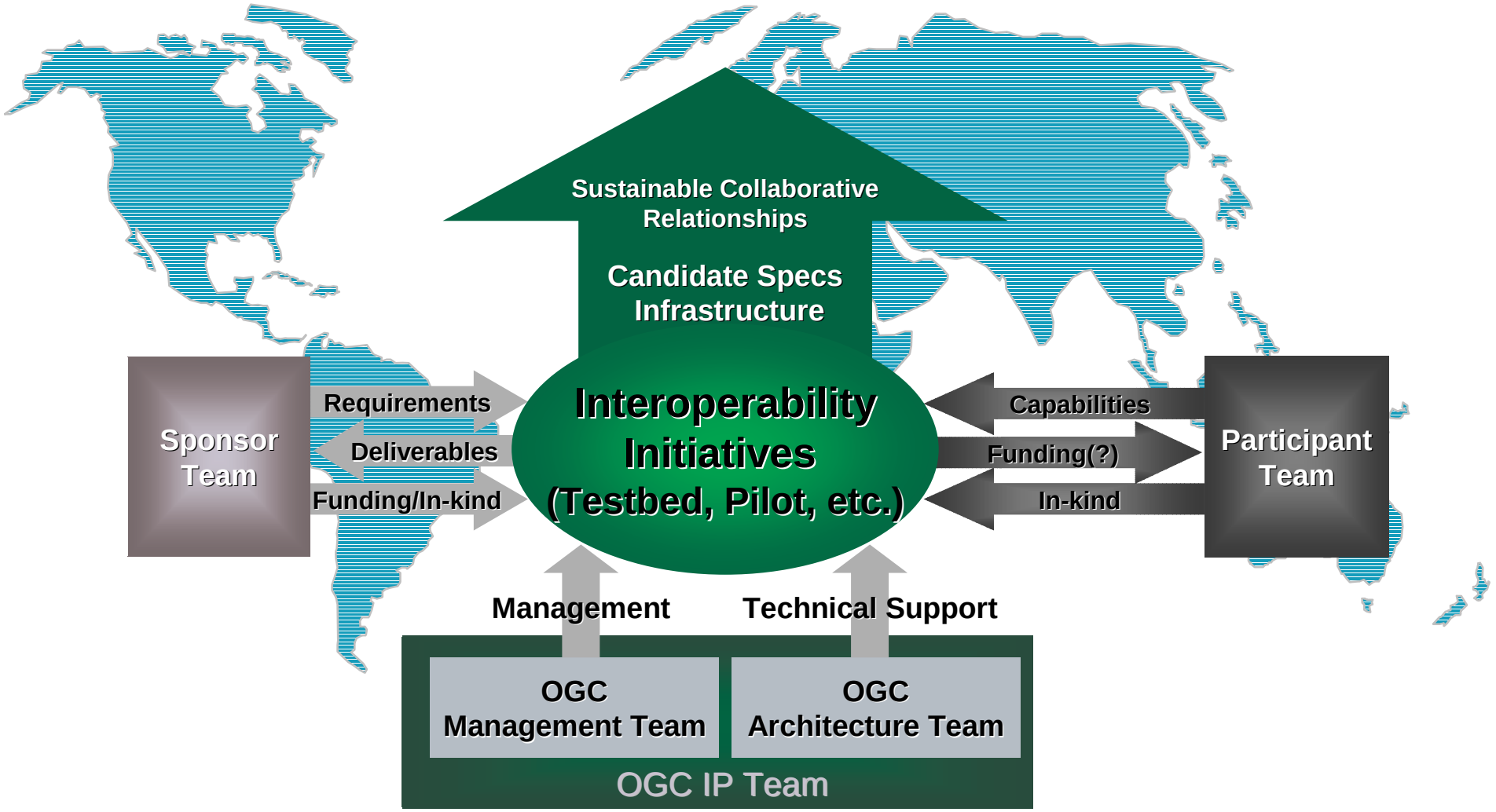
Example Members

- Integrators
  - Raytheon, Lockheed Martin, Mitsubishi, SAIC, Harris, MITRE, BAE, General Dynamics, NGIT
- Major Hardware and Software Companies
  - Sun Microsystems, Oracle, Microsoft, Adobe, HP, SAS
- Developers of GeoSpatial Technology products
  - Intergraph, ESRI, SICAD, Autodesk, LaserScan, MapInfo, Smallworld, PCI
- Government agencies that depend on geoprocessing
  - NIMA, NASA, USGS, USA/TEC, USDA, NOAA, FEMA, Census, NSF,FGDC, DMSO, UK/OS, Au/SA,
- Location Services/Telecoms
  - Hutchison 3G, Webraska, SignalSoft, Vodaphone
- Others
  - Content Providers, Power, Universities, Consultants, Startups

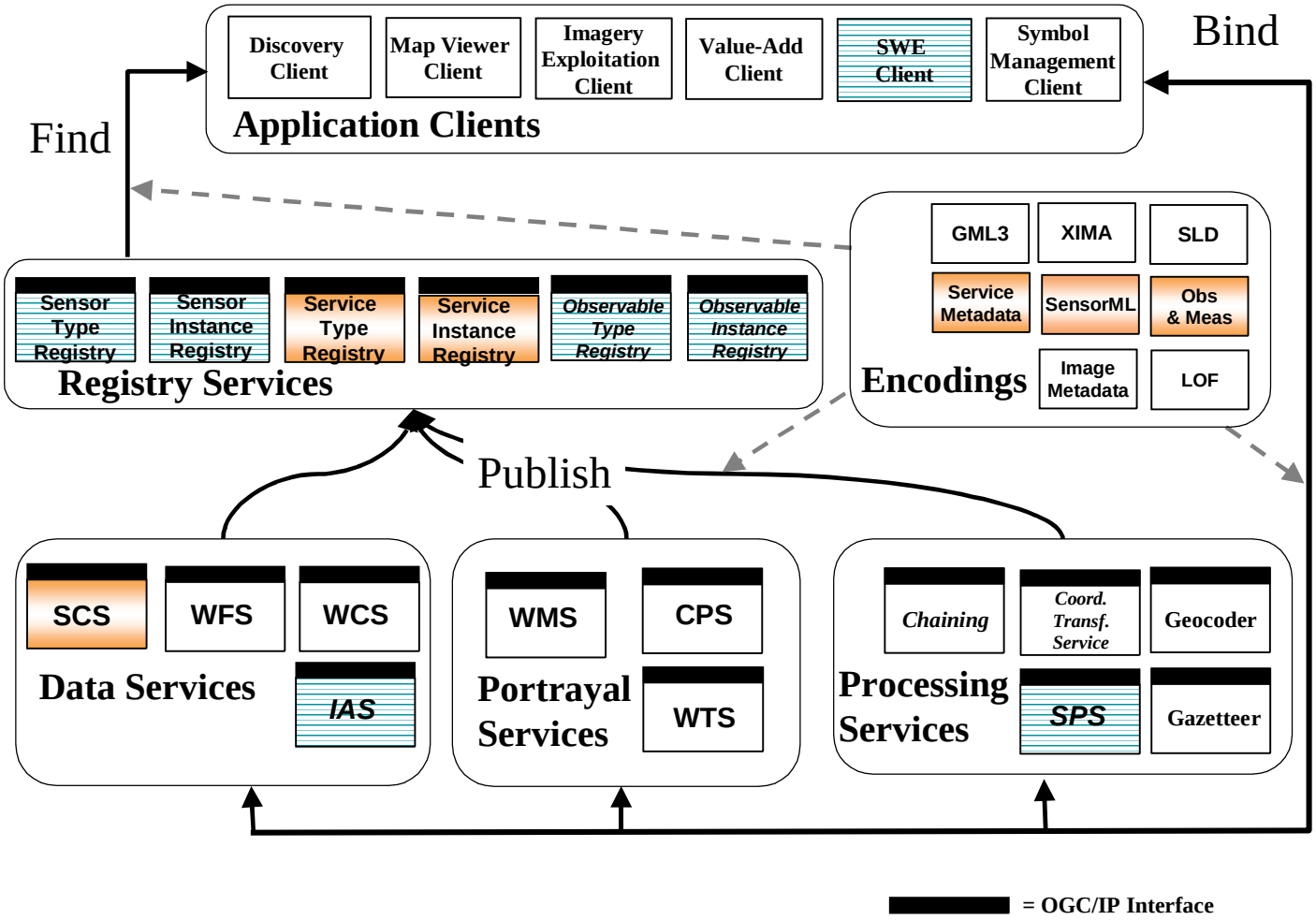
OGC Specification Process



Interoperability Initiatives - Collaboration



OpenGIS Open Web Services 1.2 Common Source Processing

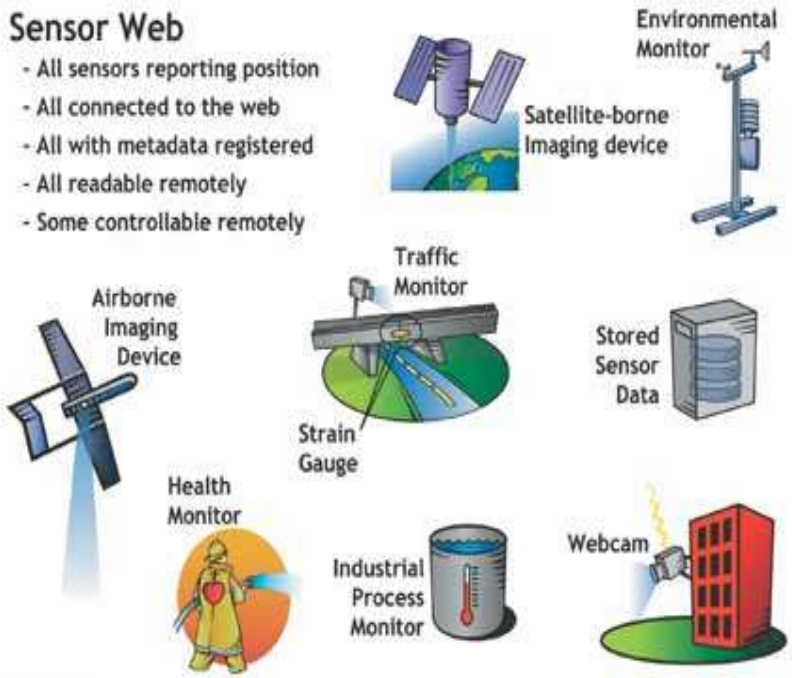


OGC Sensor Web Enablement (SWE) History

- OGC Open Web Services 1.1 (Sept 2001 – March 2002) focused significant effort on SWE design and implementation for in-situ sensors
- OGC Open Web Services 1.2 (May – Dec 2002) focused significant effort on SWE implementation for dynamic, remote sensors
- OGC issued Open Web Services 2 Request for Technology (May 2003). OWS 2 Project to begin in Fall 2003.
- Primary sponsors: NIMA, NASA, EPA, USGS

OGC Sensor Web Enablement (SWE) is a collection of open-standard schema and web service specifications that:

- **Enable the discovery of sensors and sensor data**
- **Enable the description of sensor systems and sensor data**
- **Enable the request and retrieval of both archived and real-time sensor observations**
- **Enable the tasking of sensors and sensor processing systems**
- **Enable the tasking of simulations and retrieval of simulation results**
- **Enable the notification of tasking results and alerts of phenomena of interest**
- **Enable autonomous sensor webs through inter-communication, tasking, and self-description of sensor and sensor observations**



SWE Definitions

- Observable – **a phenomenon that can be observed and measured**
- Sensor – **a device that observes and/or measures a phenomenon**
- Observed value (Observation/Measurement) – **the value returned by or derived from a sensor observation (e.g. quantity, count, boolean, category, ordered category, position)**
- Sensor Web **– a loosely-connected collection of sensor that are accessible through common web services**

SWE Components

- Sensor Model Language (SensorML) – **an XML schema for providing sensor system descriptions to support sensor discovery, and geolocation and processing of sensor observations**
- Observation & Measurements (O&M) – **a framework and XML schema for measurements and observations**
- Sensor Collection Service (SCS) – **a service that provides real-time or archived observed values**
- Sensor Planning Service (SPS) **– a service that enables sensor tasking, acquisition requests, processing and simulation requests, and registration for alert notification**

SWE Components (cont)

- Web Notification Service (WNS) **– Provides a means for Sensor Planning Services to alert people, software, or other sensor systems of SPS results or alerts regarding phenomena of interest**
- Web Registry Service (WRS) – **provide discovery mechanism for sensors and observed values**
- Observables Dictionary **– provides definitions for observables that can be measured by sensors**

Component Overview:

SensorML

Editor: Mike Botts, University of Alabama - Huntsville

an XML schema for providing sensor system descriptions to support sensor discovery, and geolocation and processing of sensor observations

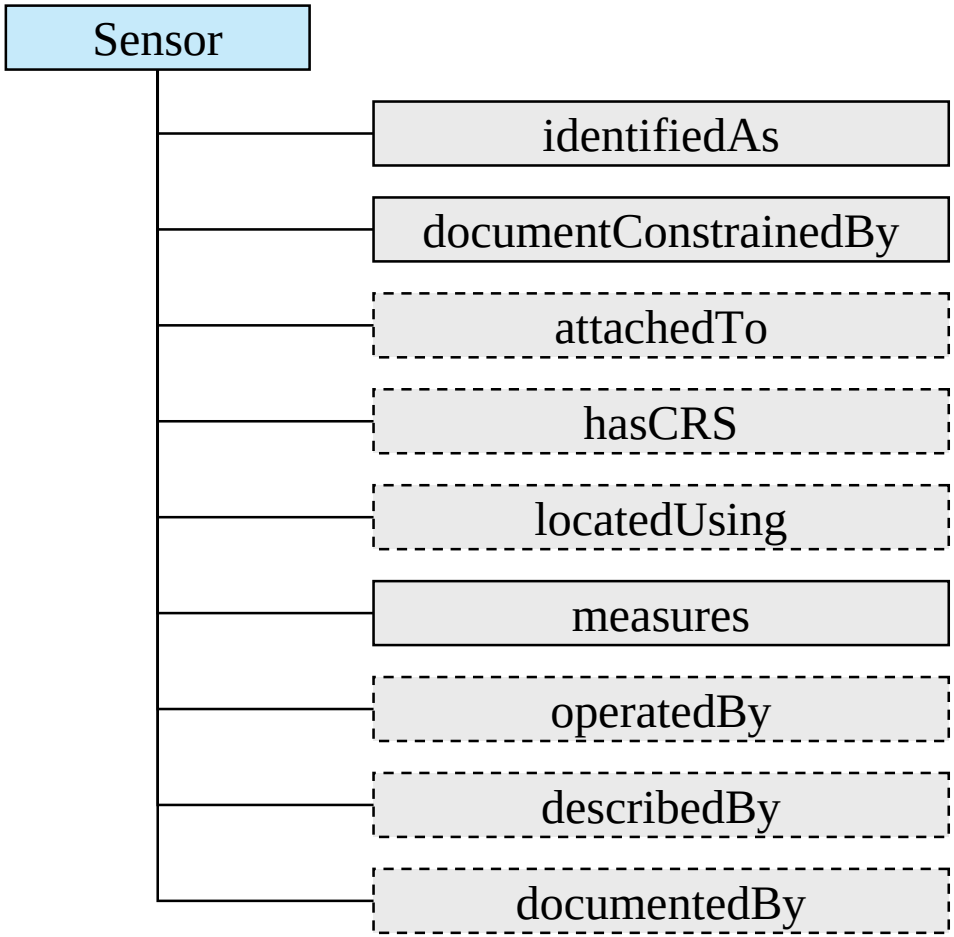
Scope of SensorML Support

- Designed to support a wide range of sensors
  - Including both dynamic and stationary platforms
  - Including both in-situ and remote sensors
- Examples:
  - Stationary, in-situ **– chemical “sniffer”, thermometer, gravity meter**
  - Stationary, remote **– stream velocity profiler, atmospheric profiler, Doppler radar**
  - Dynamic, in-situ **– aircraft mounted ozone “sniffer”, GPS unit, dropsonde**
  - Dynamic, remote **– satellite radiometer, airborne camera, soldier-mounted video**

Information Provided by SensorML

- Observation characteristics
  - **Physical properties measured (e.g. radiometry, temperature, concentration, etc.)**
  - **Quality characteristics (e.g. accuracy, precision)**
  - **Response characteristics (e.g. spectral curve, temporal response, etc.)**
- Geometry Characteristics
  - **Size, shape, spatial weight function (e.g. point spread function) of individual samples**
  - **Geometric and temporal characteristics of sample collections (e.g. scans or arrays)**
- Description and Documentation
  - **Overall information about the sensor**
  - **History and reference information supporting the SensorML document**

SensorML Schema: Sensor Identification



Response Example - YSI Wind Speed Sensor

```
<characterizedBy>
    <dynamicRange axis="http://www.opengis.net/observables#windSpeed">
        <low uom="http://www.opengis.net/units#mph">0.0</low>
        <high uom="http://www.opengis.net/units#mph">134.0</high>
    </dynamicRange>
    <threshold uom="http://www.opengis.net/units#mph"
            axis="http://www.opengis.net/observables#windSpeed">2.2</threshold>
    <operationalRange axis="http://www.opengis.net/observables#temperature">
        <low uom="http://www.opengis.net/units#celsius">-40.0</low>
        <high uom="http://www.opengis.net/units#celsius">40.0</high>
    </operationalRange>
    <survivableRange axis="http://www.opengis.net/observables#windSpeed">
        <low uom="http://www.opengis.net/units#mph">0.0</low>
        <high uom="http://www.opengis.net/units#mph">220.0</high>
    </survivableRange>
</characterizedBy>
```

SensorML Status

- Initial core schema complete
- Geolocation sensor models complete for
  - **Scanners and profilers (1D, 2D, and 3D)**
  - **Frame camera**
  - **Rapid Positioning Coordinates (RPC)**
- Specification released for public comment (2/03)
- Will push for approval as Technical Specification in summer 2003

Component Overview:

Observations and Measurements

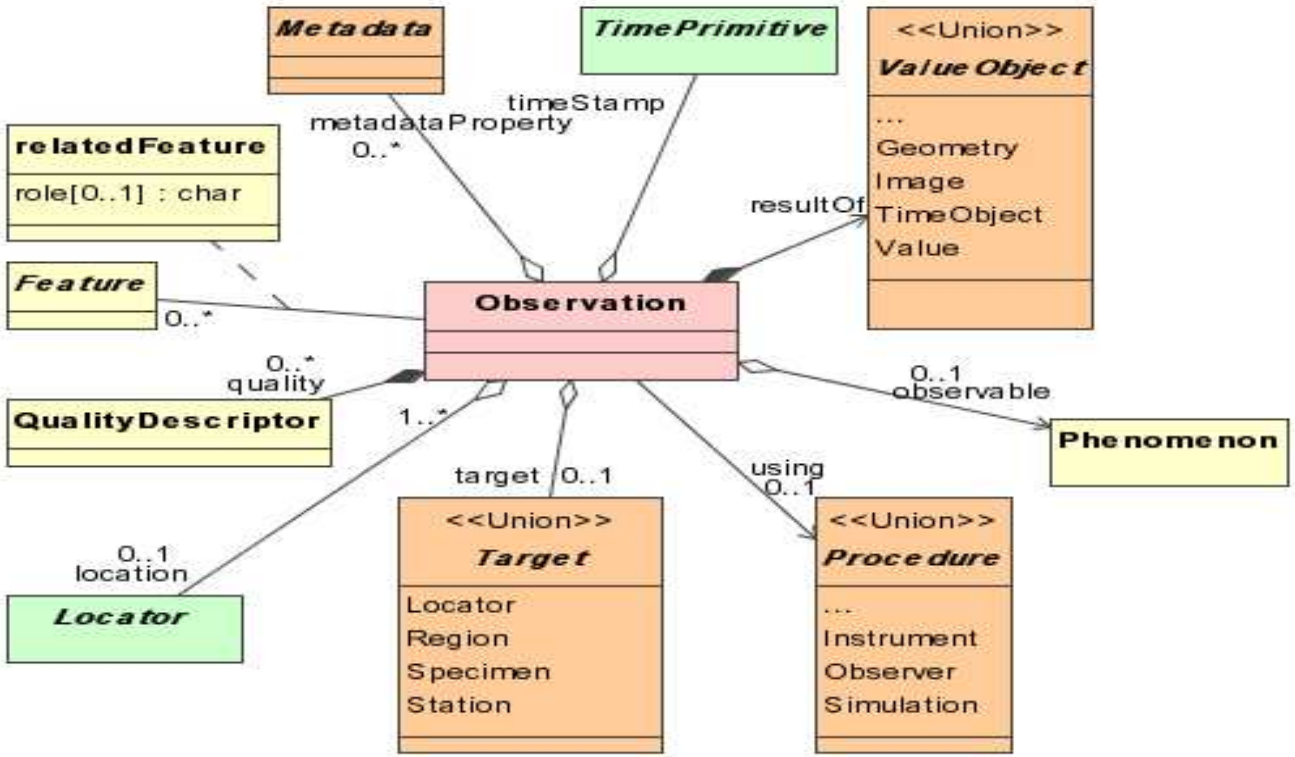
Editor: Simon Cox, CSIRO - Australia

a framework and XML schema for measurements and observations

O&M Concepts

- An Observation is considered an event with a result which is a value describing some phenomenon
  - **modelled as a Feature within the context of the OGC Feature Model.**
  - **binds the result to the (spatiotemporal) location where it was made.**
  - **involves a procedure to determine the value, which may involve a sensor or observer, an analytical procedure, simulation or other numerical process.**
  - **results in an estimate of the Value of a property or phenomenon related to the target of the observation.**

Observation Model



Example of Simple Observation

```
<om:RichObservation gml:id="OD654">
  <gml:timeStamp>
    <gml:TimeInstant><gml:timePosition>2001-12-12</gml:timePosition></gml:TimeInstant>
  </gml:timeStamp>
  <gml:using xlink:href="sensors.xml#TDS"/>
  <gml:target xlink:href="stations.xml#s432"/>
  <gml:resultOf>
    <gml:Quantity uom="units.xml#gpL">72.1</gml:Quantity>
  </gml:resultOf>
  <om:observable xlink:href="phenomena.xml#DissolvedSolids"/>
  <om:quality method="quality.xml#confidence95percent">
    <om:Bounds uom="units.xml#gpL">65.2 75.2</om:Bounds>
  </om:quality>
  <om:relatedFeature xlink:role="previous observation" xlink:href="archive/observations.xml#OD321"/>
</om:RichObservation>
```

Component Overview:

Sensor Collection Services (SCS)

Editor: Tom McCarty , SAIC – McLean, VA

a service that provides real-time or archived observed values

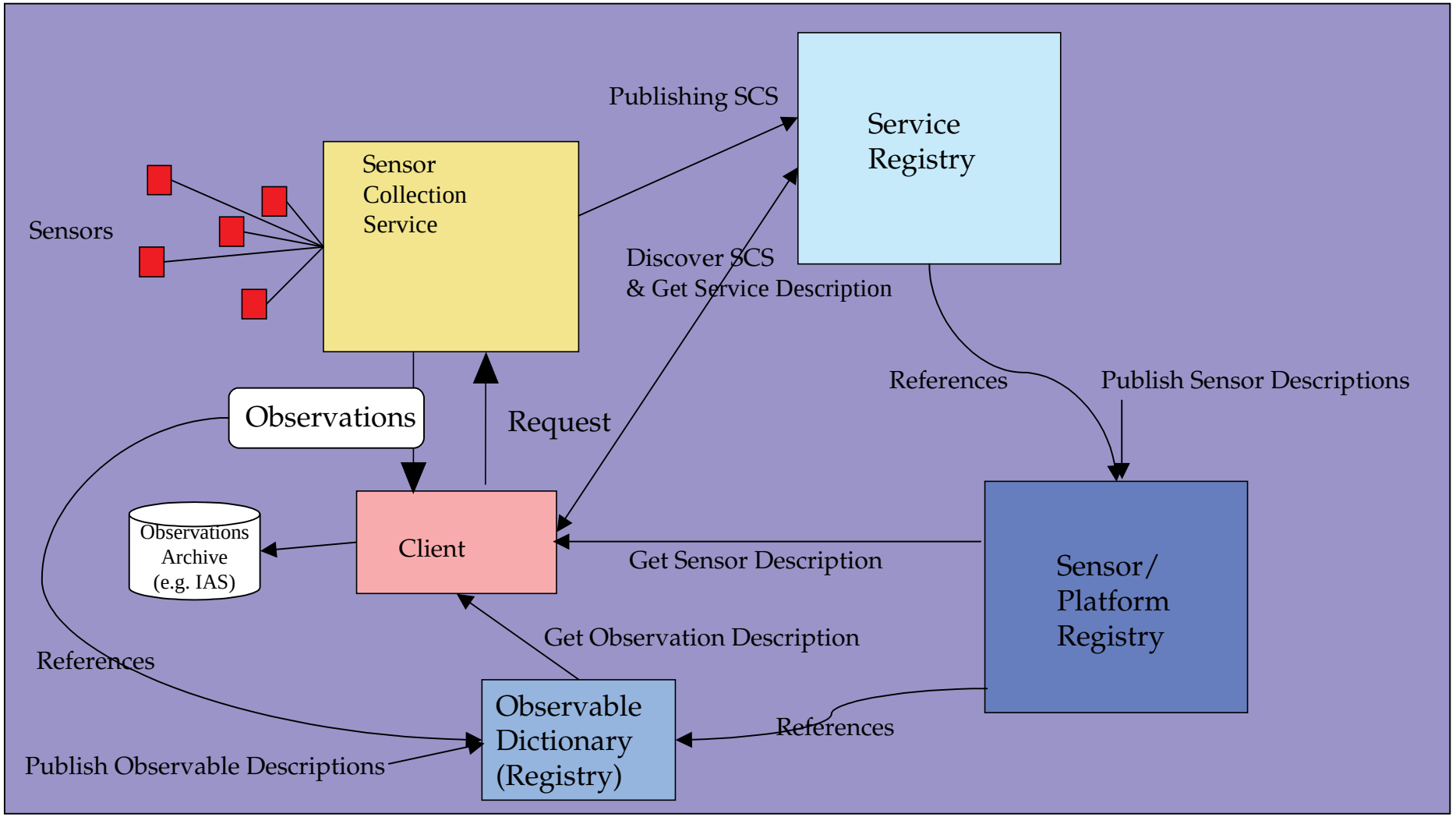
SCS Overview

- Web Service Interface that supports query and retrieval of real-time and archived sensor information.
- Facilitates the registration and discovery of deployed, on-line sensor networks.
- Provides links to sensor descriptions (SensorML)
- Returns sensor data (Observations) and provides links to Observation Dictionaries

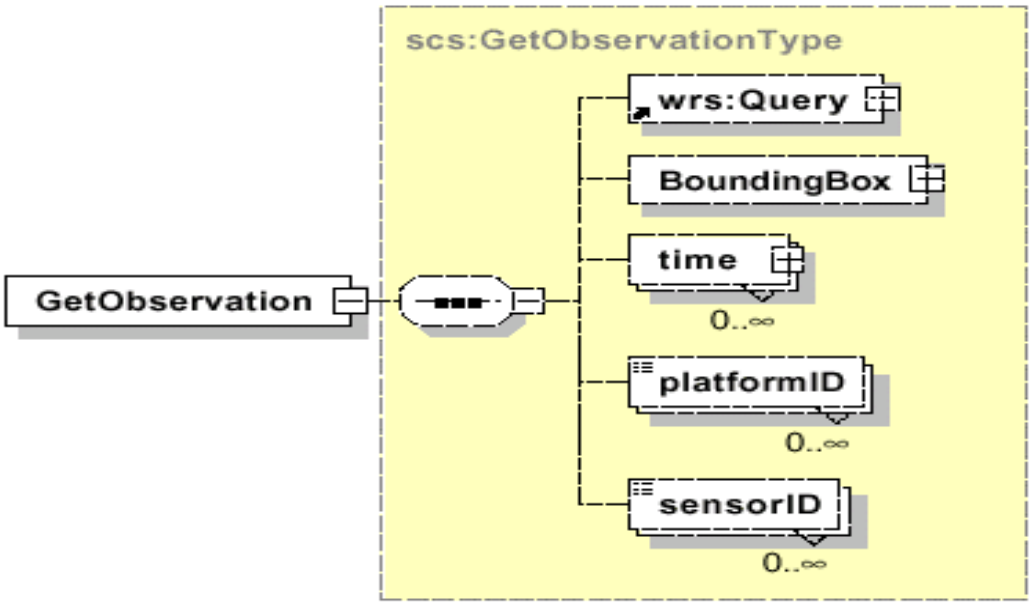
SCS Requests

- getCapabilities - **implements the standard GetCapabilities operation defined by OWSCommon.getCapabilities**
- getObservation **- query sensor collections to retrieve observation data (returns GML Observations or Coverage data)**
- describeSensor **– request sensor description (returns SensorML)**
- describePlatform **– request platform information (returns SensorML)**

Typical SCS Configuration



GetObservation request schema



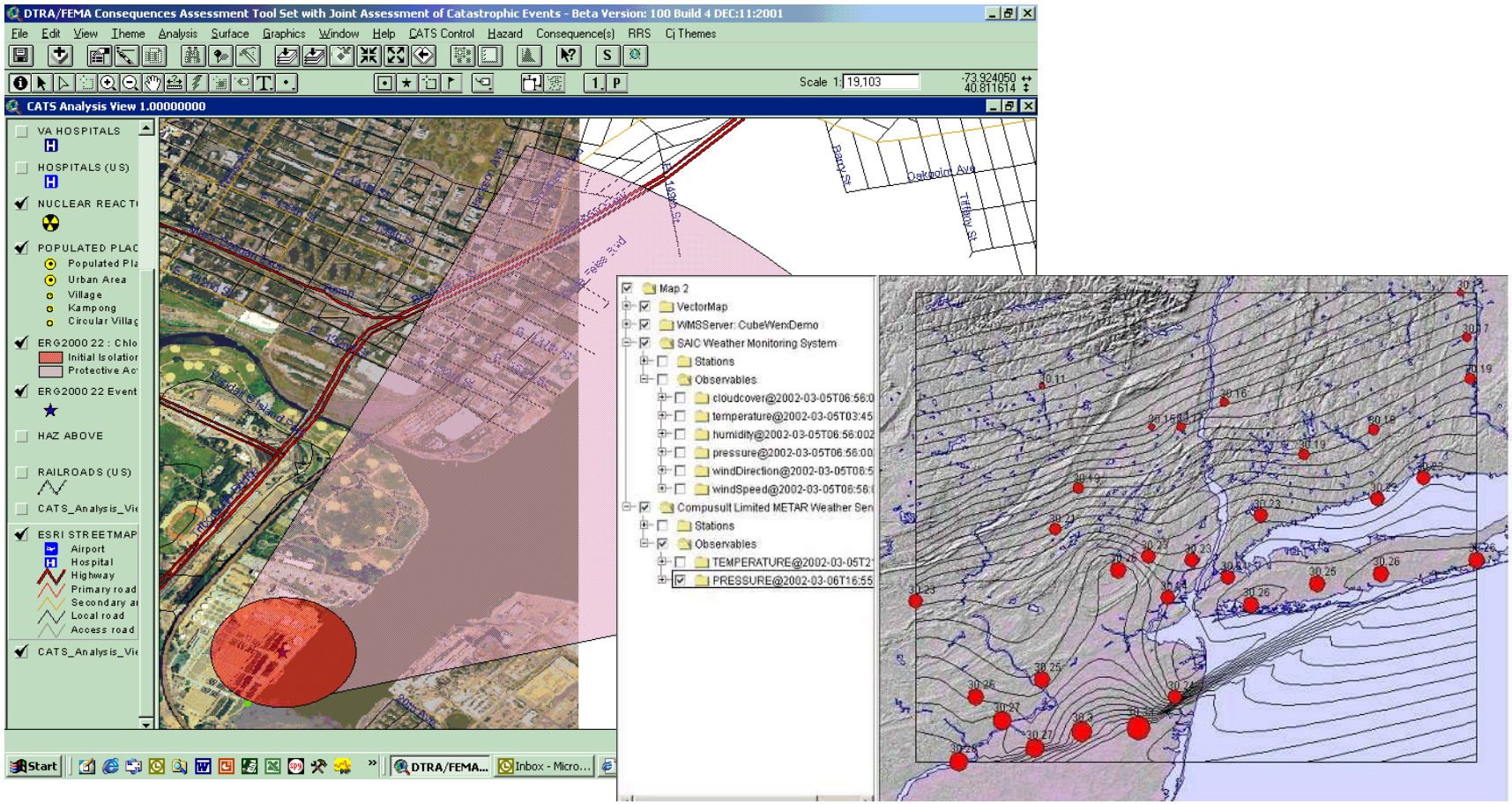
Query/Response Example

```
<GetObservation
      xmlns:gml=\"http://www.opengis.net/gml\">
 <BoundingBox srsName=\"EPSG:4326\">
  <gml:coord>
   <gml:X>-77.25</gml:X>
   <gml:Y>38.333</gml:Y>
  </gml:coord>
  <gml:coord>
   <gml:X>-76.5</gml:X>
   <gml:Y>39.0</gml:Y>
  </gml:coord>
 </BoundingBox>
 <time>
  <gml:TPeriod>
   <gml:begin>
    <gml:TInstant>
     <gml:tPosition>2002-10-
    01T19:00:00Z</gml:tPosition>
    </gml:TInstant>
   </gml:begin>
   <gml:end>
    <gml:TInstant>
     <gml:tPosition>2002-10-
    01T19:00:00Z</gml:tPosition>
    </gml:TInstant>
   </gml:end>
  </gml:TPeriod>
 </time>
</GetObservation>
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<ows:ObservationArray
      xmlns:ows="http://www.opengis.net/ows"
      xmlns:gml="http://www.opengis.net/gml">
 <gml:boundedBy>
  <gml:Envelope>
   <gml:pos srsName="EPSG:4326">38.5098 -
    77.1972</gml:pos>
   <gml:pos srsName="EPSG:4326">38.975 -
    76.7606</gml:pos>
  </gml:Envelope>
 </gml:boundedBy>
 <ows:observationMembers>
  <ows:Observation>
   <gml:location>
    <gml:Point>
     <gml:pos srsName="EPSG:4326">38.8035 -
    76.7606</gml:pos>
    </gml:Point>
   </gml:location>
   <gml:timeStamp>
    <gml:TInstant>
     <gml:tPosition>2002-10-
    01T16:00:00Z</gml:tPosition>
    </gml:TInstant>
   </gml:timeStamp>
   <ows:resultOf>
    <gml:Quantity uom="ppb">10.0</gml:Quantity>
   </ows:resultOf>
   <ows:quality method="qualitative code">
    <gml:Category>G</gml:Category>
```

OpenGIS Sensor Web Enablement Testbed 1.1

OpenGIS SWE Clients



Component Overview:

Sensor Planning Service (SPS)

Editor: Jeff Lansing, Polexis – San Diego, CA

a service that enables sensor tasking, acquisition requests, processing and simulation requests, and registration for alert notification

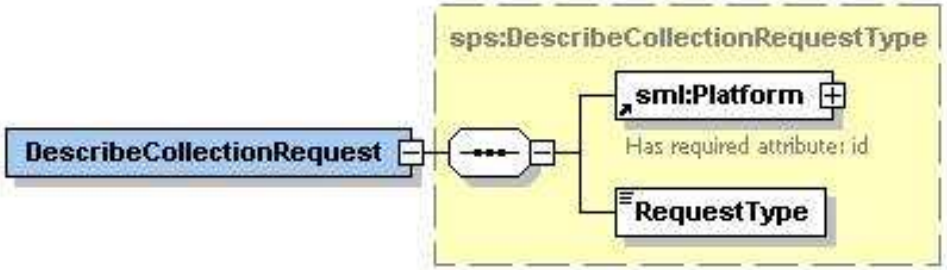
SPS Overview

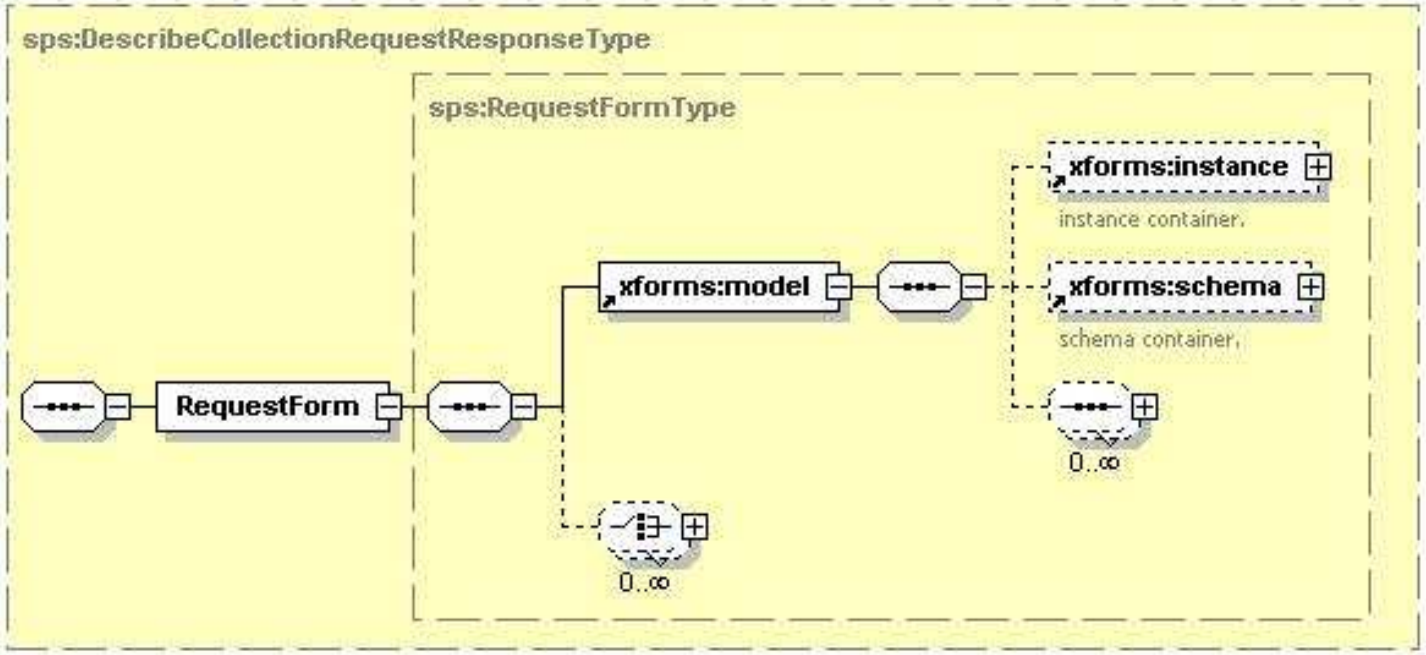
- **Provides capabilities for querying and tasking Sensor Assets**
- **Supports tasking of sensors, sensor observation processing, and simulation**
- **Based on needs for:**
  - **Collection Management**
  - **Requirements Management**
  - **Mission Management**
  - **Asset Management**
- **Supports both short-term and long-term transactions**

SPS Operations

- GetCapabilities – **SPS advertises tasking capabilities it can provide**
- DescribeCollectionRequest – **SPS provides XForm template for specifying required parameters for submitting a request (synchronous)**
- GetFeasibility – **check feasibility of successfully completing requested task (synchronous or notified later through WNS)**
- SubmitRequest – **actually request task**
- UpdateRequest – **change task**
- CancelRequest – **remove task**
- GetStatus – **check on task progress**

DescribeCollection





XForm Example

Please complete or correct fields marked **in red** .

SensorID AMSU-A

StartTime (Enter value in the format yyyy-mm-ddThh:mm:ss.ssssssZ) 2002-01-01T00:00:00

StopTime (Enter value in the format yyyy-mm-ddThh:mm:ss.ssssssZ) 2002-01-02T12:00:00

westLon (Enter value in degrees between -180 to +180) -45.7

eastLon (Enter value in degrees between -180 to +180) -40

southLat (Enter value in degrees between -90 to +90) 36.57

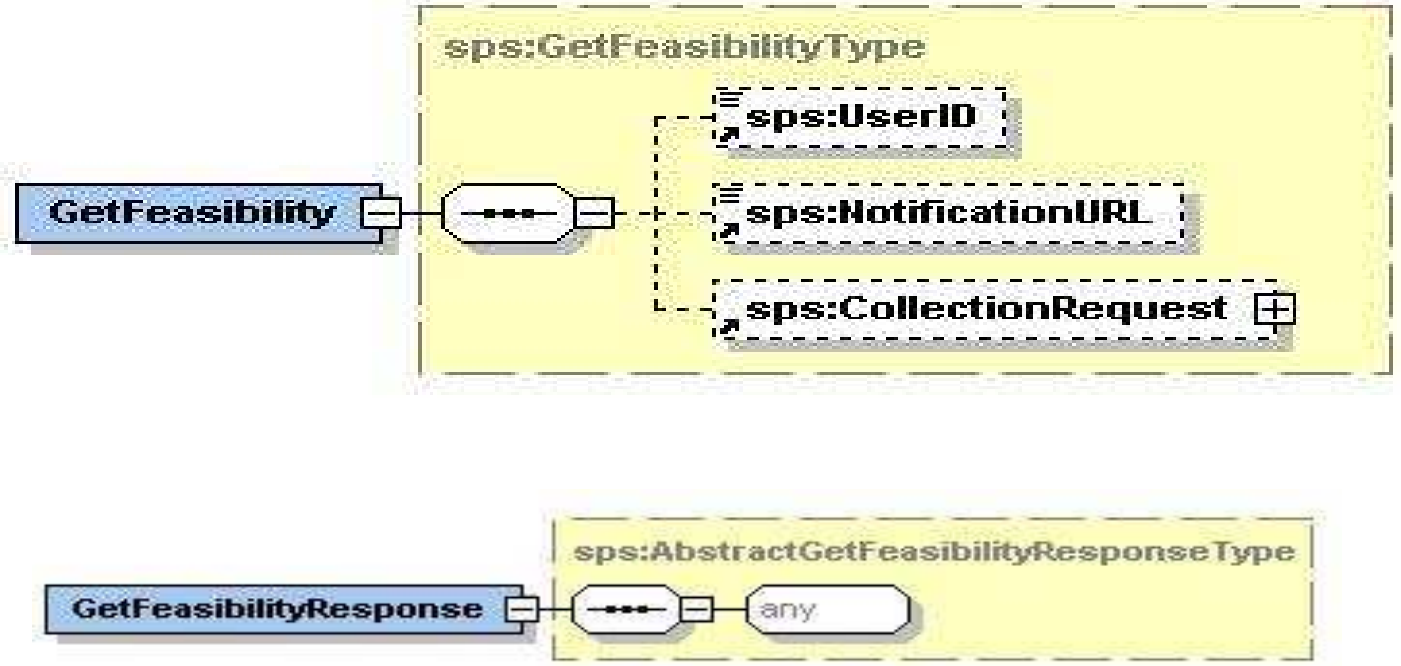
northLat (Enter value in degrees between -90 to +90) 37.12

Submit Form

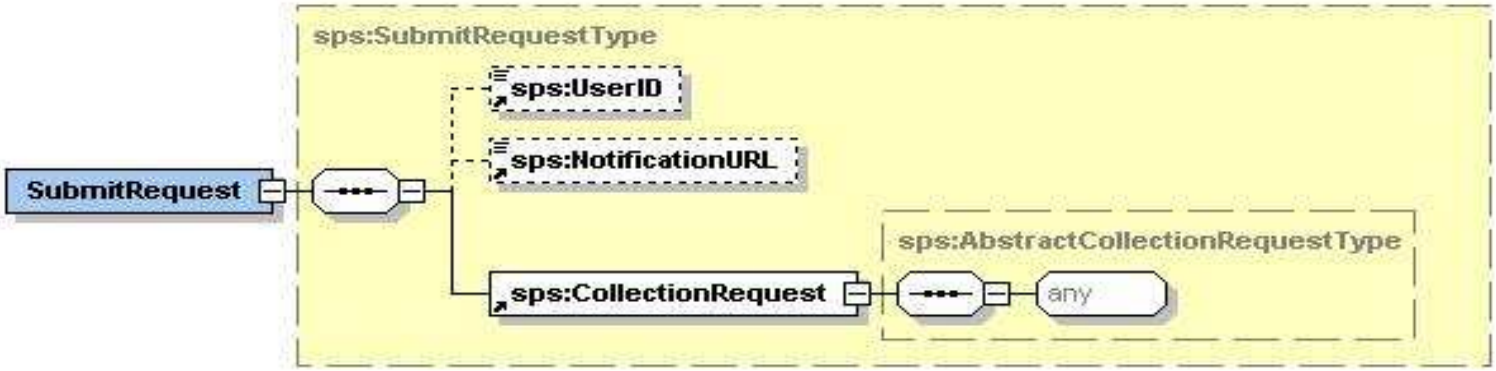
Record - data user has entered as of last submission

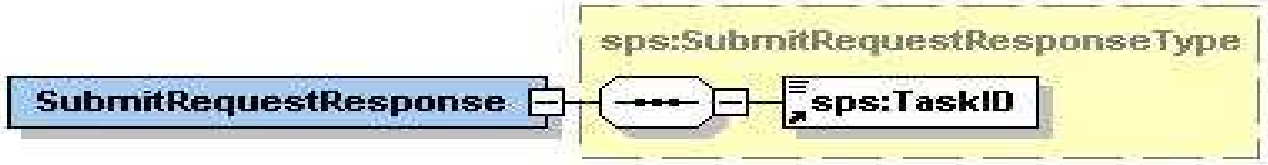
```
<?xml version="1.0"?>

   <form>
      <SensorID>AMSU-A</SensorID>
      <timeRange>
         <StartTime>2002-01-01T00:00:00</StartTime>
         <StopTime>2002-01-02T12:00:00</StopTime>
      </timeRange>
      <roi>
         <westLon>-45.7</westLon>
         <eastLon>-40</eastLon>
         <southLat>36.57</southLat>
         <northLat>37.12</northLat>
      </roi>
      <submit>Submit Form</submit>
   </form>
```

GetFeasibility



SubmitRequest





Component Overview:

Web Notification Service (WNS)

Editors: Ingo Simonis / Andreas Witzisk,
University of Muenster – Muenster, GE

provides a means to alert people, software, or other sensor systems of SPS results or alerts regarding phenomena of interest

WNS Overview

- **An asynchronous and stateful service.**
- **A web interface (e.g. operated by the SPS) that allows sending notifications to a client with well structured content.**
- **Mechanism for notification delivery can include:**
  - **email**
  - **URL / HTTP Post (e.g. CGI or servlet)**
  - **SMS**
  - **Instant Messenger**
  - **Phone / FAX**
  - **Mail**

WNS Operations

- GetCapabilities – **advertises WNS capabilities**
- RegisterUser – **allows user to register and set notification method (returns userID)**
- DoNotification – **called to initiate notification of a user (requires a userID and a message; optional corrID)**
- DoCommunication – **initiate a communication with a user. An asynchronous dialogue structure, means that the WNS will send a notification to the user. This notification indicates to the user that further action has to be taken (which can be done automatically).**
- DoReply – **allows the user to answer a dialogue notification; uses the http-post command**

Example Message for DoNotification

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XML Spy v4.4 U (http://www.xmlspy.com)-->
<DoNotification xmlns="http://www.opengis.net/wns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/wns ..\wns.xsd" Version="0.0.1" Service="WNS">
    <UserID>4294967295</UserID>
    <Message>
        <Type>Operation completed</Type>
        <MessageParameter>
            <CorrID>2147483647</CorrID>
            <Key>Requested data available at</Key>
            <URI>http://a.data.source/data.xyz</URI>
        </MessageParameter>
        <MessageParameter>
            <CorrID>2147483647</CorrID>
            <Key>Costs</Key>
            <String>USD 2000.-</String>
        </MessageParameter>
    </Message>
</DoNotification>
```

Example DoCommunication

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XML Spy v4.4 U (http://www.xmlspy.com)-->
<CommunicationRequestMessage xmlns="http://www.opengis.net/wns"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/wns
d:\Projekte\OGC_HLA\SWE\WNS\XML\wnsMessage.xsd">
    <Action>Information needed</Action>
    <MessageParameter>
        <CorrID>2147483647</CorrID>
        <Key>Lookangle has to be defined</Key>
        <Unit>degree</Unit>
        <Options>-10 -5 0 5 10</Options>
    </MessageParameter>
</CommunicationRequestMessage>
```

Final Points

- OpenGIS standards are voluntary, but leverage common commercial platforms and user bases
- Most SensorNet applications will be geospatially enabled
- SensorNet sponsorship or participation in Open Web Services 2

Contact Information

- OpenGIS Consortium
  - http://www.opengis.org
- Tom McCarty
  - mccartyt@saic.com