

Open Geospatial Consortium

Publication Date: 2014-07-15

Approved Date: 2014-06-14

Posted Date: 2014-05-16

Reference number of this document: OGC 14-016

Reference URL for this document: <http://www.opengeospatial.net/doc/PER/testbed10/cci-vgi>

Category: Public Engineering Report

Editors: Arne Bröring; Simon Jirka; Matthes Rieke, Benjamin Pross

OGC[®] Testbed-10 CCI VGI Engineering Report

Copyright © 2014 Open Geospatial Consortium

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is not an OGC Standard. This document is an OGC Public Engineering Report created as a deliverable in an OGC Interoperability Initiative and is 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, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements.

Document type: OGC[®] Engineering Report
Document subtype: NA
Document stage: Approved for public release
Document language: English

Abstract

This Engineering Report was created as a deliverable for the OGC Testbed 10 (Testbed-10) initiative of the OGC Interoperability Program. This report describes an approach for integrating Volunteered Geographic Information (VGI) into a spatial data infrastructure and reports on findings about the advancements using VGI resources. It includes optimization ideas, service change recommendations, and lessons learned.

This is not a normative document.

Keywords

ogcdoc, ogc documents, Testbed-10, vgi, cci, sos, wfs, wps, swe,om

Preface

A significant part of the OGC standards development process is the Interoperability Program (IP), which conducts international interoperability initiatives such as Testbeds, Pilot Projects, Interoperability Experiments, and Interoperability Expert Services. These activities are designed to encourage rapid development, testing, validation, demonstration and adoption of open, consensus based standards and best practices.

The OGC Testbed 10 (Testbed-10) is a Testbed within the Interoperability Program. Within Testbed-10, The Cross-Community Interoperability (CCI) thread seeks to build on interoperability within communities sharing geospatial data and advance semantic mediation approaches for data discovery, access and use of heterogeneous data models and heterogeneous metadata models. This thread explored the creation of domain ontologies and tools to create, assemble, and disseminate geographic data provided voluntarily by individuals.

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

| Contents | | Page |
|-----------------|---|-------------|
| 1 | Introduction..... | 1 |
| 1.1 | Scope | 1 |
| 1.2 | Document contributor contact points | 1 |
| 1.3 | Revision history..... | 1 |
| 1.4 | Future work | 2 |
| 1.5 | Foreword | 2 |
| 2 | References..... | 3 |
| 3 | Terms and definitions | 4 |
| 4 | Conventions | 5 |
| 4.1 | Abbreviated terms | 5 |
| 4.2 | UML notation..... | 5 |
| 5 | Testbed-10 CCI VGI Engineering Report | 6 |
| 5.1 | Integrating VGI with Spatial Data Infrastructures | 6 |
| 5.2 | SOS for VGI..... | 7 |
| 5.3 | WFS for VGI..... | 8 |
| 5.3.1 | Example Requests..... | 8 |
| 5.3.2 | Filtering..... | 8 |
| 5.3.3 | Example Responses | 9 |
| 5.4 | Pattern for Integrating VGI Sources..... | 11 |
| 5.4.1 | Invocation of Data Loading..... | 11 |
| 5.4.2 | Data Gathering..... | 11 |
| 5.4.3 | Client Integration | 12 |
| 5.5 | Linking WPS for VGI data..... | 13 |
| 5.5.1 | Implementation | 14 |
| 5.5.2 | Process inputs..... | 14 |
| 5.5.3 | Process Outputs..... | 15 |
| 5.5.4 | Example Execute Request..... | 15 |
| 5.5.5 | Process Execution | 16 |
| 5.5.6 | Example Response..... | 17 |
| 6 | Accomplishments..... | 20 |
| 7 | Lessons Learned..... | 21 |

| Figures | Page |
|--|-------------|
| Figure 1 - Example Flickr observation..... | 7 |
| Figure 2 - Example of integrating Flickr with OGC services..... | 11 |
| Figure 3 Client showing Flickr image (source: PYXIS innovation) | 13 |
| Figure 4 VGI Linking architecture..... | 14 |
| Figure 5 Matching of name parts with tags | 17 |

OGC® Testbed-10 CCI VGI Engineering Report

1 Introduction

1.1 Scope

This OGC® Engineering Report describes the approach on integrating Volunteered Geographic Information (VGI) into the Testbed-10 Cross-Community Interoperability service architecture. It provides an overview of the data sources for VGI as well as related technology. The concept of using a Web Feature Service in combination with a Sensor Observation Service for handling *human as sensors* is illustrated from an interoperability perspective. The developed data model is presented and illustrated with examples from a VGI platform.

1.2 Document contributor contact points

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

| Name | Organization |
|----------------------|--------------------------------|
| Arne Bröring (AB) | 52°North, broering@52north.org |
| Matthes Rieke (MRI) | 52°North, m.rieke@52north.org |
| Benjamin Pross (BPR) | 52°North, b.pross@52north.org |
| Simon Jirka (SJI) | 52°North, jirka@52north.org |

1.3 Revision history

| Date | Release | Editor | Primary clauses modified | Description |
|------------|---------|--------|--------------------------|--|
| 13.12.2013 | 0.0.1 | AB | All | Initial Document |
| 20.02.2014 | 0.0.2 | MRI | Scope, 5.1, 5.3, 5.4 | First Draft |
| 27.03.2014 | 0.03 | BPR | 5.4, 6,7 | Added section about Linking WPS for VGI, worked on Accomplishments and Lessons learned sections. |
| 30.04.2014 | 0.04 | SJI | All | Final Draft |
| 16.05.2014 | 1.0.0 | MRI | | Editorial changes for final version |

1.4 Future work

Improvements in this document are desirable to the following topics:

- Additional VGI sources could be explored to further prove the developed concept of modeling VGI data as human observations
- The use of SWE standards for modeling and serving VGI is very promising. Further work should be invested in this topic to derive best practice guidelines and/or a profile recommending how SWE can be used for VGI.
- The linking concept could be extended in conjunction with the conflation approach (see OGC-14-001) to create rules to decide whether data will be conflated or linked

1.5 Foreword

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

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

2 References

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

OGC 11-063, OWS-8 CCI Semantic Mediation ER

OGC 14-001 Testbed-10 CCI Provenance ER

OGC 06-121r3, OGC[®] Web Services Common Standard

OGC 05-007r7, OpenGIS[®] Web Processing Service

OGC Revision Notes for OpenGIS[®] Implementation Specification: Geographic information - Geography Markup Language Version 3.2.1

NOTE This OWS Common Standard contains a list of normative references that are also applicable to this Implementation Standard.

3 Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard [OGC 06-121r3] shall apply. In addition, the following terms and definitions apply.

3.1 Conflation

Conflation is understood as the process of unifying two or more separate datasets, which share certain characteristics, into one integrated all-encompassing result.

3.2 Volunteered Geographic Information

Volunteered Geographic Information in the context of this document is considered any kind of human observation which is georeferenced (e.g. photographs, textual observations, etc.).

4 Conventions

4.1 Abbreviated terms

| | |
|------------|---|
| OGC | Open Geospatial Consortium |
| O&M | Observations & Measurements |
| Testbed-10 | OGC Testbed 10 (Formerly known as OGC Web Services, Phase 10 or OWS-10) |
| SOS | Sensor Observation Service |
| VGI | Volunteered Geographic Information |
| WFS | Web Feature Service |
| WMS | Web Map Service |
| WPS | Web Processing Service |

4.2 UML notation

Some diagrams that appear in this standard are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 06-121r3]. Additionally, UML sequence diagrams are used to illustrate workflow.

5 Testbed-10 CCI VGI Engineering Report

This document summarizes the work carried out within Testbed-10 in the field of VGI as part of the CCI thread. It provides details on the general service architecture, service interfaces as well as on internal workflows developed within this test bed.

5.1 Integrating VGI with Spatial Data Infrastructures

VGI has become an important source of information to geographers, authorities, and the general public facilitating the understanding of our environment. Of particular importance is VGI for supporting the management of emergency and crisis situations. Thereby, from a computational perspective, the *human* can be considered as a *sensor* producing heterogeneous data and contributing to a vast information space of different VGI sources (e.g. Twitter, Flickr, and OpenStreetMap). This view of *humans as sensors* has been suggested and investigated by various researchers (e.g. [1], [2]). Our previous work (e.g. [3], [4]) supports this view by utilizing OGC's Sensor Web Enablement (SWE) framework to integrate VGI with spatial data infrastructures.

By building up on the findings for making use of VGI in the OWS-9 test bed (see [OGC 12-103r3]), we propose to follow the above described view on VGI and to incorporate the advantages of the SWE specifications into the overall architecture. As previous work has shown, the Sensor Web components are particularly useful for this challenging integration task. Using the Observations & Measurements (O&M) data model allows transforming the heterogeneous VGI data into a standardized model and format. O&M is flexible enough to support various VGI sources. At the same time, O&M is a narrow GML application schema and thus ensures a high degree of interoperability.

Once the data is encoded as O&M, we have paved the road for an interoperable usage in OGC service components. VGI data can be collected and provided through the standardized Sensor Observation Service (SOS) as well as the Web Feature Service (WFS). Using the Sensor Event Service (SES) we can also incorporate real time complex event processing on incoming VGI data streams encoded as O&M.

Once these fundamental services allow an interoperable access to VGI, other required services for data linking or rule-based conflation can be plugged into the architecture. Encapsulated behind well-defined interfaces (i.e., WPS profiles), similar services as used in OWS-9 for ontology look-up (SPARQL), semantic mediation, geoparsing VGI, or attribute/feature/dataset conflation can be easily incorporated.

- [1] Goodchild, Michael F. (2007): Citizens as sensors: the world of volunteered geography. *GeoJournal* 69(4), pp. 211-221.
 [2] Schade, S., Diaz, L., Ostermann, F., Spinsanti, L., Luraschi, G., Cox, S. & De Longueville, B. (2013): Citizen-based sensing of crisis events: sensor web enablement for volunteered geographic information. *Applied Geomatics*, 5(1), pp. 3-18.
 [3] Lasnia, D., Bröring, A., Jirka, S. & Remke, A. (2010): Crowdsourcing Sensor Tasks to a Socio-Geographic Network. *AGILE 2010: The 13th AGILE International Conference on Geographic Information Science*. 11.-14. May 2010. Guimaraes, Portugal.
 [4] Jürrens, E.H., A. Bröring & Jirka, S. (2009): A Human Sensor Web for Water Availability Monitoring. *Proceedings of: OneSpace 2009 - 2nd International Workshop on Blending Physical and Digital Spaces on the Internet*. September 1. 2009. Berlin, Germany.

5.2 SOS for VGI

As described in section 5.1 VGI data can be modeled as observation data collected by humans. In order to align the information available in a VGI data source with the O&M model, a conceptual mapping has been developed. Figure 1 illustrates this approach. It features an additional abstraction layer (in blue), the Human Observer. This way an existing VGI data source (e.g. Flickr photos) can be easily mapped against the O&M model.

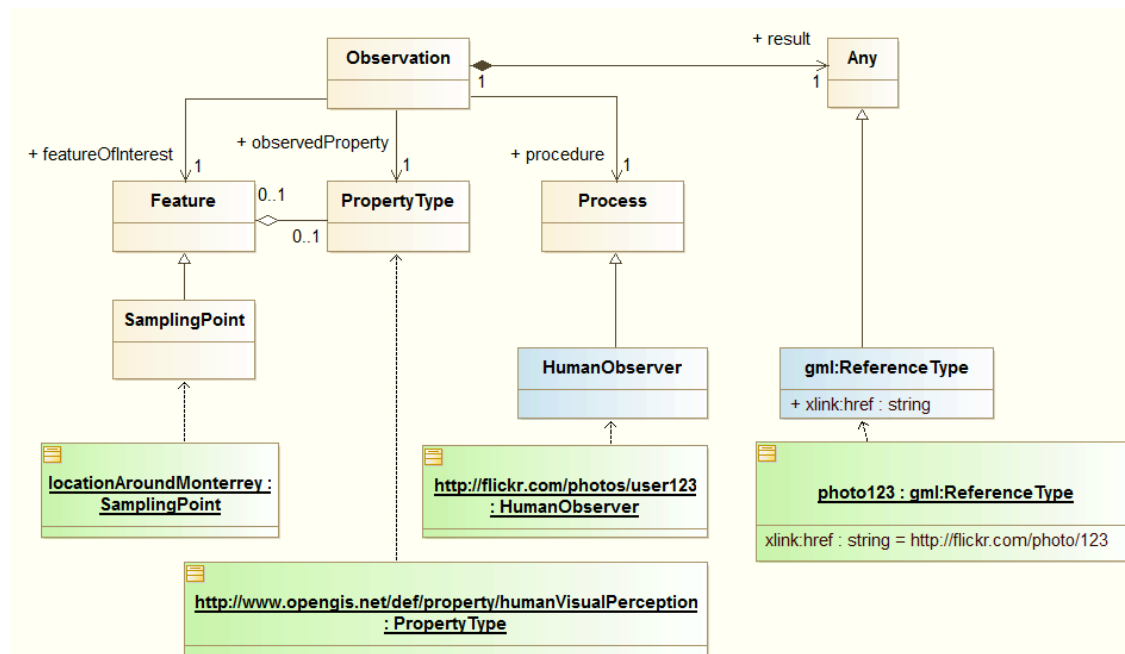


Figure 1 - Example Flickr observation

Since 52°North has developed a reference implementation of the SOS 2.0 standard, this model has been implemented with the latest 52°North SOS server implementation (52°North SOS 4.0). Besides mapping this O&M model to the default data model of the 52°North SOS a minor enhancement of the GetObservation was necessary: a filter for requesting observations (e.g. images) based on their descriptions (e.g. “fire”) was introduced.

5.3 WFS for VGI

However, as the SOS interface was not supported by all clients in the scenario, also the 52°North SOS was enhanced to also provide a WFS 2.0 service interface. Following this approach the WFS for VGI leverages the underlying robust implementation of the SOS standard by mapping incoming WFS 2.0 requests to the appropriate SOS 2.0 requests. As O&M 2.0 is a GML 3.2.1 Application Schema, the WFS for VGI can simply provide the data in the same format as via the SOS 2.0 interface methods. In addition the service has been designed to be able to reuse existing functionality such as filtering via keywords or spatial extents. The following sections provide an overview on the realized functionality.

5.3.1 Example Requests

The WFS for VGI currently supports KVP requests.

A minimal getFeature request for getting all observations is shown below:

```
http://ows.dev.52north.org:8080/52n-wfs-
webapp/sos/kvp?service=WFS&version=2.0.0&request=GetFeature&namespaces=xmlns(om,http://ww
w.opengis.net/om/2.0)&typenames=om:Observation
```

5.3.2 Filtering

Filtering is also supported, for example the can features be filtered by a temporal extent:

```
http://ows.dev.52north.org:8080/52n-wfs-
webapp/sos/kvp?service=WFS&version=2.0.0&request=GetFeature&namespaces=xmlns(om,http://ww
w.opengis.net/om/2.0)&typenames=om:Observation&filter=<Filter
xmlns="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/fes/2.0
http://schemas.opengis.net/filter/2.0.0/filter.xsd http://www.opengis.net/gml/3.2
http://www.opengis.net/gml/3.2.1/base/gml.xsd"><During><ValueReference>om:phenomenonTime<
/ValueReference><gml:TimePeriod gml:id="tp_1"><gml:beginPosition>2012-11-
19T14:00:00.000+01:00</gml:beginPosition><gml:endPosition>2012-11-
19T14:05:00.000+01:00</gml:endPosition></gml:TimePeriod></During></Filter>
```

The specific match against a given TimeInstant is also possible:

```
http://ows.dev.52north.org:8080/52n-wfs-
webapp/sos/kvp?service=WFS&version=2.0.0&request=GetFeature&namespaces=xmlns(om,http://ww
w.opengis.net/om/2.0)&typenames=om:Observation&filter=<Filter
xmlns="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/fes/2.0
http://schemas.opengis.net/filter/2.0.0/filter.xsd http://www.opengis.net/gml/3.2
http://www.opengis.net/gml/3.2.1/base/gml.xsd"><TEquals><ValueReference>om:phenomenonTime
</ValueReference><gml:TimeInstant gml:id="ti_1"><gml:timePosition>2012-11-
19T14:00:00.000+01:00</gml:timePosition></gml:TimeInstant></TEquals></Filter>
```

As well as matching of property-values

```
http://ows.dev.52north.org:8080/52n-wfs-
webapp/sos/kvp?service=WFS&version=2.0.0&request=GetFeature&namespaces=xmlns(om,http://ww
w.opengis.net/om/2.0),xmlns(gml,http://www.opengis.net/gml/3.2)&typenames=om:Observation&
filter=<Filter xmlns="http://www.opengis.net/fes/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/fes/2.0
http://schemas.opengis.net/filter/2.0.0/filter.xsd http://www.opengis.net/gml/3.2
http://www.opengis.net/gml/3.2.1/base/gml.xsd"><PropertyIsLike wildCard="" singleChar=""
escapeChar=""><ValueReference>om:observation/gml:description</ValueReference><Literal>fir
e</Literal></PropertyIsLike></Filter>
```

This is used by the VGI Linking process described in section 5.5.

5.3.3 Example Responses

In the following an example response containing features created from VGI data is shown:

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:om="http://www.opengis.net/om/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xlink="http://www.w3.org/1999/xlink"
timeStamp="2014-04-16T11:23:26.633+02:00" numberMatched="unknown"
numberReturned="22"
xsi:schemaLocation="http://www.opengis.net/wfs/2.0
schemas.opengis.net/wfs/2.0/wfs.xsd http://www.opengis.net/gml/3.2
http://schemas.opengis.net/gml/3.2.1/gml.xsd http://www.opengis.net/samplingSpatial/2.0
http://schemas.opengis.net/samplingSpatial/2.0/spatialSamplingFeature.xsd
http://www.opengis.net/om/2.0 http://schemas.opengis.net/om/2.0/observation.xsd
http://www.opengis.net/sampling/2.0
http://schemas.opengis.net/sampling/2.0/samplingFeature.xsd">
  <wfs:member>
    <om:OM_Observation gml:id="o_83264B50F3D2C25BEF6EB2B2CA76F7E0E015DC1E">
      <gml:description>forest fire near monterrey</gml:description>
      <om:type
        xlink:href="http://www.opengis.net/def/observationType/OGC-
OM/2.0/OM_CategoryObservation" />
      <om:phenomenonTime>
        <gml:TimeInstant gml:id="phenomenonTime_158">
          <gml:timePosition>2014-02-04T12:02:23.000Z</gml:timePosition>
        </gml:TimeInstant>
      </om:phenomenonTime>
      <om:resultTime>
        <gml:TimeInstant gml:id="ti_2336D75401AA7406702EDE8A4D73D237542442A6">
          <gml:timePosition>2014-02-04T12:23:21.000Z</gml:timePosition>
        </gml:TimeInstant>
      </om:resultTime>
      <om:procedure xlink:href="http://www.flickr.com/photos/residae" />
      <om:observedProperty
        xlink:href="http://www.opengis.net/def/property/humanvisualPerception" />
      <om:featureOfInterest>
        <sams:SF_SpatialSamplingFeature
          xmlns:sams="http://www.opengis.net/samplingSpatial/2.0"
          xmlns:sf="http://www.opengis.net/sampling/2.0"
          gml:id="ssf_6EB259299855DFF130189FB65DE2F90E61E35EDC">
          <gml:description>forest fire monterrey</gml:description>
          <gml:identifier codespace="">photo-295155260-
feature</gml:identifier>
          <gml:name>Forest Fire near Monterrey, CA</gml:name>
          <sf:type
            xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-
OM/2.0/SF_SamplingPoint" />
          <sf:sampledFeature
            xlink:href="http://www.52north.org/sos/featureOfInterest/world"
          />
          <sams:shape>
            <gml:Point
              gml:id="point_ssf_6EB259299855DFF130189FB65DE2F90E61E35EDC">
                <gml:pos
                  srsName="http://www.opengis.net/def/crs/EPSSG/0/4326">-121.8847 36.5792</gml:pos>
```

```

        </gml:Point>
      </sams:shape>
    </sams:SF_SpatialSamplingFeature>
  </om:featureOfInterest>
  <om:result xmlns:ns="http://www.opengis.net/gml/3.2"
    xlink:href="http://www.flickr.com/photos/residae/295155260"
    xsi:type="ns:ReferenceType" />
</om:OM_Observation>
</wfs:member>
<wfs:member>
  <om:OM_Observation gml:id="o_61D86F43F5B87B83431926262AE0E6C2E8C99C76">
    <om:type
      xlink:href="http://www.opengis.net/def/observationType/OGC-
OM/2.0/OM_CategoryObservation" />
    <om:phenomenonTime>
      <gml:TimeInstant gml:id="phenomenonTime_147">
        <gml:timePosition>2012-05-06T09:55:00.000Z</gml:timePosition>
      </gml:TimeInstant>
    </om:phenomenonTime>
    <om:resultTime>
      <gml:TimeInstant gml:id="ti_A82B45AC5E505C78F6141AA4AA4EB28D1FF89B95">
        <gml:timePosition>2012-07-02T04:44:00.000Z</gml:timePosition>
      </gml:TimeInstant>
    </om:resultTime>
    <om:procedure xlink:href="http://www.flickr.com/photos/bmw328driver" />
    <om:observedProperty
      xlink:href="http://www.opengis.net/def/property/humanVisualPerception" />
    <om:featureOfInterest>
      <sams:SF_SpatialSamplingFeature
        xmlns:sams="http://www.opengis.net/samplingSpatial/2.0"
        xmlns:sf="http://www.opengis.net/sampling/2.0"
        gml:id="ssf_883FB6DBAA3561460F315899C5545D6F969162AB">
        <gml:description>monterrey, airport, monterreyairport, california,
captainroyvoris,
        biography, history, statues, sculptures, busts, military,
        militaryhistory</gml:description>
        <gml:identifier codeSpace="">photo-7484667352-
feature</gml:identifier>
        <gml:name>Aviation Legends</gml:name>
        <sf:type
          xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-
OM/2.0/SF_SamplingPoint" />
        <sf:sampledFeature
          xlink:href="http://www.52north.org/sos/featureOfInterest/world"
        />
        <sams:shape>
          <gml:Point
            gml:id="point_ssf_883FB6DBAA3561460F315899C5545D6F969162AB">
              <gml:pos
                srsName="http://www.opengis.net/def/crs/EPSSG/0/4326">-121.8505 36.587166</gml:pos>
              </gml:Point>
            </sams:shape>
          </sams:SF_SpatialSamplingFeature>
        </om:featureOfInterest>
      <om:result xmlns:ns="http://www.opengis.net/gml/3.2"
        xlink:href="http://www.flickr.com/photos/bmw328driver/7484667352/"
        xsi:type="ns:ReferenceType" />
    </om:OM_Observation>
  </wfs:member>
</wfs:FeatureCollection>

```


5.4 Pattern for Integrating VGI Sources

This section provides information on how VGI data has been gathered from the Flickr platform and integrated into the WFS for VGI instance.

The figure below shows an example integration of Flickr as a VGI source with spatial data infrastructures. Therefore, a new VGI Component serves as an integrator which is capable of crawling Flickr for certain photographs. The VGI Component can be triggered via its own REST interface, to start the crawling process and to upload metadata about photographs to the SOS/WFS. Once uploaded to this service, the data is available from there through the standardized interface of the SOS/WFS and encoded as O&M data.

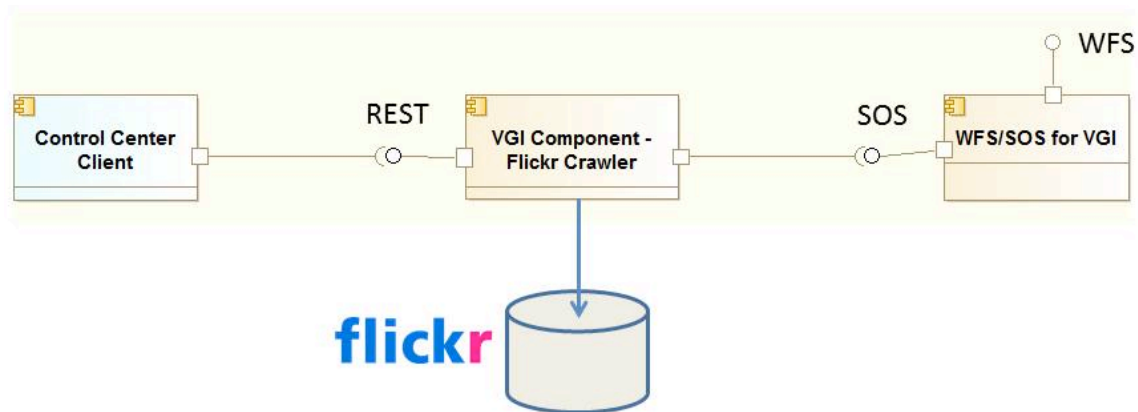


Figure 2 - Example of integrating Flickr with OGC services

In the following we describe in detail the functionality of the different components involved in the integration.

Please note: A similar approach has also been implemented to harvest tweets from Twitter. The implementation approach is identical except that the Twitter API is called instead of the Flickr API. Thus, all of the following descriptions also apply to the Twitter implementation.

5.4.1 Invocation of Data Loading

To initiate data loading / harvesting, a dedicated software component, the Control Center Client, was used. This component is the entry point to the integration process. The user can utilize this tool to trigger the download of specific Flickr/Twitter metadata which will then be made available on the SOS/WFS server. As the VGI Component itself is stateless, the client has to keep track of already integrated data. This was realized via simple time-based query definitions (see section 5.4.2).

5.4.2 Data Gathering

The VGI component is the central component responsible for the interaction with VGI sources such as Flickr and Twitter. It offers a REST API to trigger the search for VGI

data sets (observations). The REST API accepts specific parameters (keywords, bbox, and time) to search for photographs, e.g.:

```
http://<server>/flickrMetaCrawler/search?keywords=Monterey,Airport&bbox=-140,16,-100,56&minTime=2008-12-01&maxTime=2013-12-31
```

Subsequently, this component uses the Flickr API to find photographs conforming to the submitted query. The metadata of those photographs are downloaded and uploaded as observations to the SOS/WFS server (see section 5.2).

This crawling process can be called *on demand* through the REST interface of the VGI component. For example, the FlickrCrawler can find all Flickr photographs for a specified point of interest (POI) and with certain keywords attached (e.g., all Flickr photos in the area of “LAX” airport and tagged with the word “fire”).

The collected VGI data is translated to the O&M model and uploaded to the SOS/WFS for VGI using the transactional operations InsertSensor and InsertObservation as defined by the SOS standard. Flickr metadata can be uploaded here as O&M. Metadata is made available and queryable through the standardized WFS and SOS interfaces. From the SOS/WFS, the VGI data is available for display, or further processing.

5.4.3 Client Integration

The client accesses the VGI data via the WFS 2.0 interface. It displays the features using the shape of the SpatialSamplingFeature enclosed in the O&M Observation. The URL of the Flickr image/the Twitter tweet is stored as attribute of the feature. It can be visualized in a built-in browser as shown in Figure 3.

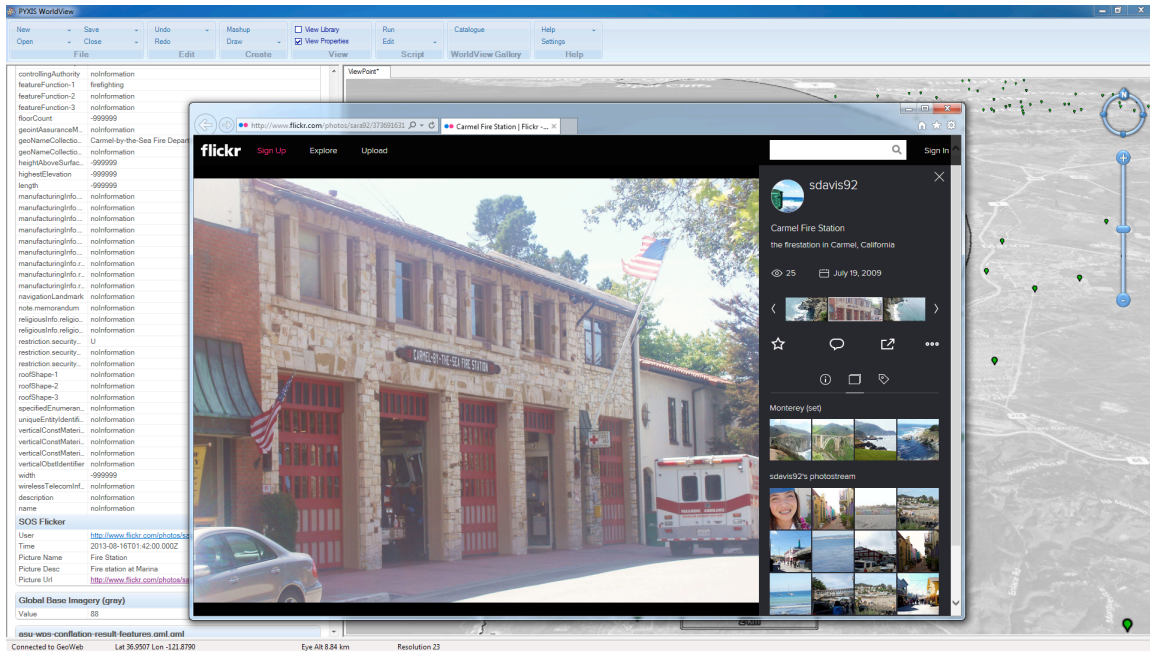


Figure 3 Client showing Flickr image (source: PYXIS innovation)

5.5 Linking WPS for VGI data

Linking of VGI data that is described here means the creation of relationships between VGI data and geospatial datasets by matching the names or textual descriptions of the single features. By the process of VGI data linking the existing datasets can be enriched with additional information such as recent images of a feature or recent discussions about a feature from social platforms.

The linking process works as follows:

1. As the linking process builds upon the conflation process [OGC-14-001] this has to be executed first by the client.
2. The result of the conflation process will be displayed in the client.
3. The second dataset is loaded from the SOS/WFS for VGI as described in section 5.2.
4. The linking process is executed with both datasets and additional parameters.
5. The feature name/descriptions are matched and the features are ranked.
6. A table with the ranked matches is returned to the client.
7. Relationships between features of the two involved datasets can be visualized by the client, e.g. by drawing a line.

5.5.1 Implementation

For the demo scenario the VGI linking process uses a conflated dataset of fire stations in the Monterey area and images from Flickr. The matching of the names is performed using the FuzzyWuzzy¹ software package. The architecture of the involved services is depicted in Figure 4.

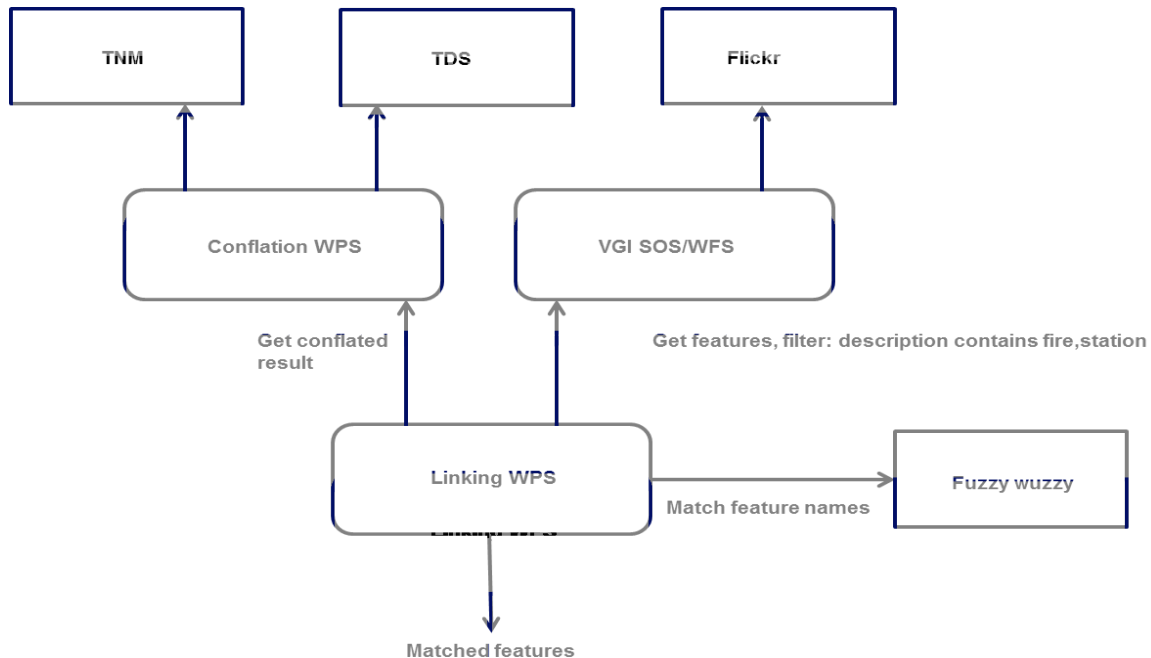


Figure 4 VGI Linking architecture

5.5.2 Process inputs

The linking process requires the following inputs:

- Source Features: The names of the features will be matched against the target features
- Target Features: The names of the features will be matched against the source features
- Target Description Filter: Target features are only considered, if their description (Flickr tags) contain the value of this input

¹ <https://github.com/seatgeek/fuzzywuzzy>

- Search Distance: Distance around a source feature in which target features will be considered for matching
- FuzzyWuzzy Threshold: If the FuzzyWuzzy score of two matched names is above this threshold, the respective features are considered a match.

5.5.3 Process Outputs

The following output is returned by the process:

- Matched Features: The client can select, whether the result should be encoded as CSV or RDF.

5.5.4 Example Execute Request

In the following an example Execute request for the VGI linking process is shown:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0" xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsExecute_request.xsd">
  <ows:Identifier>org.n52.wps.server.algorithm.conflation.LinkingProcess</ows:Identifier>
  <wps>DataInputs>
    <wps:Input>
      <ows:Identifier>Source_Features</ows:Identifier>
      <wps:Reference mimeType="text/xml"
        schema="http://schemas.opengis.net/gml/3.2.1/gml.xsd"
        xlink:href="http://ows.dev.52north.org:8080/wps/RetrieveResultServlet?id=12d4e411-9833-
48d8-b635-21f73a917d6bconflated_result.c892514e-44b0-4455-96ce-25877263fa00" />
    </wps:Input>
    <wps:Input>
      <ows:Identifier>Target_Features</ows:Identifier>
      <wps:Reference mimeType="text/xml; subtype=gml/3.2.1"
        schema="http://schemas.opengis.net/gml/3.2.1/gml.xsd"
        xlink:href="http://ows.dev.52north.org:8080/52n-wfs-
webapp/sos/kvp?service=WFS&version=2.0.0&request=GetFeature&namespaces=xmlns(
om%2Chttp%3A%2F%2Fwww.opengis.net%2Fom%2F2.0)&typenames=om%3A0bservation" />
    </wps:Input>
    <wps:Input>
      <ows:Identifier>Target_Description_Filter</ows:Identifier>
      <wps>Data>
        <wps:LiteralData>fire, station</wps:LiteralData>
      </wps>Data>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>Search_Distance</ows:Identifier>
      <wps>Data>
        <wps:LiteralData>1</wps:LiteralData>
      </wps>Data>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>Fuzzywuzzy_Threshold</ows:Identifier>
      <wps>Data>
        <wps:LiteralData>70</wps:LiteralData>
      </wps>Data>
    </wps:Input>
  </wps>DataInputs>
  <wps:ResponseForm>
    <wps:ResponseDocument>
      <wps:Output mimeType="text/csv">
        <ows:Identifier>Matched_Features</ows:Identifier>
        <ows:Title>result</ows:Title>
        <ows:Abstract>result</ows:Abstract>
      </wps:Output>
    </wps:ResponseDocument>
  </wps:ResponseForm>
</wps:Execute>
```

```
</wps:ResponseForm>
</wps:Execute>
```

The Source Features are retrieved from a WPS instance. These features are the result of a conflation process that has been executed beforehand. The Target Features are loaded from the SOS/WFS for VGI. In this example, no filter is applied so that all available features are fetched from the SOS/WFS for VGI. The Search Distance has been set to 1 mile and the threshold for the FuzzyWuzzy software has been set to 70.

5.5.5 Process Execution

The process iterates over all source features. The target features are filtered against the search distance around the source feature. Several approaches were tested to find the best attributes for performing the matching algorithm.

As the result, from the conflated dataset the `name` attribute was chosen.

The VGI data has several attributes which hold relevant information about the feature. For example the `description` attribute that is used to store the original tags of the Flickr image and the `name` attribute is used to store the original image title. Based on our tests with the VGI data we came to the conclusion that the description/tags of the images are best suited for the matching process. The use of the complete string of tags of the VGI data item (e.g. photograph or tweet) and the complete name of the conflated feature did not lead to satisfying results². Even though the FuzzyWuzzy software has built-in methods³ to match composite strings, we were not able to retrieve satisfying results using these either. In our tests the best method turned out to be splitting the names in parts and matching each of the parts against each other using FuzzyWuzzy (see Figure 5). The output of the matching is a score value.

The scores are gathered and the average score is calculated. If the average score is greater than the threshold specified by the client, the features are considered as a match. If more than one VGI feature is considered as a match, the one with the highest FuzzyWuzzy score and the closest proximity to the source feature is chosen. FuzzyWuzzy score, distance, both feature IDs and names are stored and returned to the client.

² The names in the conflated dataset tend to contain articles and prepositions that are often missing from the descriptions/tags of the VGI data. A satisfying result would return a high value (between 70 and 100) even if there are words missing in one of the matched strings or if there are spelling errors.

³ Cf. <http://chairerd.seatgeek.com/fuzzywuzzy-fuzzy-string-matching-in-python/>

| | | |
|------------|------------|--------|
| Presidio | Monterey | -> 25 |
| of | Presisidio | -> 89 |
| Monterey | Fire | -> 33 |
| Fire | Station | -> 40 |
| Department | | |
| ----- | | |
| Presidio | Monterey | -> 100 |
| of | Presisidio | -> 22 |
| Monterey | Fire | -> 33 |
| Fire | Station | -> 40 |
| Department | | |
| ----- | | |
| Presidio | Monterey | -> 33 |
| of | Presisidio | -> 30 |
| Monterey | Fire | -> 29 |
| Fire | Station | -> 24 |
| Department | | |

Figure 5 Matching of name parts with tags

5.5.6 Example Response

The result of the process is shown in the following:

CSV:

FW Score,Dist(MI),NGA_ID,VGI_ID,NGA_NAME,VGI_NAME

82,0.5537791700787285,CWFID.ST_FIRE_STATION.0.24.4D92295C14D9295E1F20020000,o_79297999E839C301E4708EDFDDA965F29B4AE02F,Presidio of Monterey Fire Department,Monterey Presisidio Fire Station

82,0.697960013797296,CWFID.ST_FIRE_STATION.0.28.4D922C4D7A3553401F20020000,o_234324KDW235246DK243562BUW23075235FEFW34,Marina Fire Department,Fire station Marina

82,0.779479322935927,CWFID.ST_FIRE_STATION.0.16.4D9221B24EF356781F20020000,o_9732KHOH32027353K32432GKWD234072342EFWEF,Pacific Grove Fire Department,Fire station Pacific Grove

81,0.8071084229538052,CWFID.ST_FIRE_STATION.0.22.4D9223A2DF1583761F20020000,o_96324HKG234834IEWFE09782342134FVFE329478,City of Seaside Fire Department,Seaside Fire Station

70,0.1525441338102515,CWFID.ST_FIRE_STATION.0.20.4D9222C84B9CF9041F20020000,o_45834892353HG30583LGF8U97231245124214GHD,Monterey Peninsula Airport Fire Department,Airport Fire Station Monterey

The result values are separated by commas, the first value is the FuzzyWuzzy score, the second is the distance in miles. After this the ID of the conflated feature, the ID of the matching VGI feature, the name of the conflated feature and the name of the VGI feature are shown.

You can see that the first feature has a high average FuzzyWuzzy score, even though there is a spelling error in the tags of the VGI feature.


```

rdfs:label 'City of Seaside Fire Department';
id:identifier "CWFID.ST_FIRE_STATION.0.22.4D9223A2DF1583761F20020000".

<http://www.52north.org/o_96324HKG234834IEWFE09782342134FVFE329478> a f2n:F2N_Feature;
rdfs:label 'Seaside Fire Station';
id:identifier "o_96324HKG234834IEWFE09782342134FVFE329478".

<http://metadata.dod.mil/mdr/ns/GSIP/3.0/tds/3.0/CWFID.ST_FIRE_STATION.0.16.4D9221B24EF35
6781F20020000> a nga:NGA_Feature;
rdfs:label 'Pacific Grove Fire Department';
id:identifier "CWFID.ST_FIRE_STATION.0.16.4D9221B24EF356781F20020000".

<http://www.52north.org/o_9732KHOH32027353K32432GKWD234072342EFWEF> a f2n:F2N_Feature;
rdfs:label 'Fire station Pacific Grove';
id:identifier "o_9732KHOH32027353K32432GKWD234072342EFWEF".

<http://metadata.dod.mil/mdr/ns/GSIP/3.0/tds/3.0/CWFID.ST_FIRE_STATION.0.28.4D922C4D7A355
3401F20020000> a nga:NGA_Feature;
rdfs:label 'Marina Fire Department';
id:identifier "CWFID.ST_FIRE_STATION.0.28.4D922C4D7A3553401F20020000".

<http://www.52north.org/o_234324KDW235246DK243562BUW23075235FEFW34> a f2n:F2N_Feature;
rdfs:label 'Fire station Marina';
id:identifier "o_234324KDW235246DK243562BUW23075235FEFW34".

<http://metadata.dod.mil/mdr/ns/GSIP/3.0/tds/3.0/CWFID.ST_FIRE_STATION.0.20.4D9222C84B9CF
9041F20020000> a nga:NGA_Feature;
rdfs:label 'Monterey Peninsula Airport Fire Department';
id:identifier "CWFID.ST_FIRE_STATION.0.20.4D9222C84B9CF9041F20020000".

<http://www.52north.org/o_45834892353HG30583LGF8U97231245124214GHD> a f2n:F2N_Feature;
rdfs:label 'Airport Fire Station Monterey';
id:identifier "o_45834892353HG30583LGF8U97231245124214GHD".

<http://metadata.dod.mil/mdr/ns/GSIP/3.0/tds/3.0/CWFID.ST_FIRE_STATION.0.24.4D92295C14D92
95E1F20020000> a nga:NGA_Feature;
rdfs:label 'Presidio of Monterey Fire Department';
id:identifier "CWFID.ST_FIRE_STATION.0.24.4D92295C14D9295E1F20020000".

<http://www.52north.org/o_79297999E839C301E4708EDFDDA965F29B4AE02F> a f2n:F2N_Feature;
rdfs:label 'Monterey Presisidio Fire Station';
id:identifier "o_79297999E839C301E4708EDFDDA965F29B4AE02F".

```

Here the result is shown in the form of RDF triples. The matching features are represented as SimilarityLinks contained in a LinkSet. The FuzzyWuzzy score and the distance values are grouped in a ScoreDetails element. There were considerations to create an aggregated score out of both values for each SimilarityLink, but this was postponed for future work. The wps:score element is a placeholder for the aggregated score and is set to null for all SimilarityLinks for the time being.

6 Accomplishments

Within this work an approach to expose and process VGI data in a standardized way has been developed. It was shown how the OGC Sensor Web Enablement Standards are suited to handle VGI data sets. In detail, the following results were achieved during the activities in the Testbed-10 testbed.

- Approach for modeling VGI data using the O&M 2.0 model: It has been shown, how VGI data can be represented in the O&M 2.0 data model so that it can be easily be used within a broad range of applications.
- Conceptual approach for interoperable provision of VGI data through standardized interfaces: This approach describes how the SOS 2.0 interface can be used to serve VGI data. In addition, it has been shown, how the integration into existing components can be achieved using a WFS layer on top of the SOS 2.0 interface.
- Implementation of SOS/WFS instances for VGI: Based on the 52°North SOS, a combined SOS/WFS implementation for serving VGI data encoded in O&M 2.0 was developed.
- Methods for harvesting VGI data: A simple interface for initiating the harvesting of VGI data has been designed and implemented. Furthermore harvesting mechanisms for VGI data from Flickr and Twitter were implemented.
- Conflation process for VGI data for use within a WPS: A process for conflating VGI data with other data sets was developed for the 52°North WPS implementation. The resulting implementation was successfully tested. This process can be used to link VGI data to existing datasets to enrich them with additional information in the form of VGI sources such as Flickr images.

7 Lessons Learned

Our experiments show that mapping VGI data to O&M 2.0 is an excellent way to achieve a consistent representation of VGI from different sources and of different types. In some cases, it is necessary to define additional fields (e.g. for providing the textual description of an image) but O&M offers the necessary means to include such information. Thus, we give the recommendation that O&M is well suited to handle VGI data. For the future, it would be useful, to advance the results of this reports into a Best Practice Document which gives guidance to organizations, users, and organizations that need to deal with the interoperable exchange of VGI data.

Similarly to the use of O&M 2.0 as a data model/encoding, an appropriate interface for data access is required. We have experimented with the SOS 2.0 interface and came to the conclusion that its well structured set of query parameters offers an excellent basis for interoperability accessing VGI resources. Especially, the dedicated query parameters for the different observation properties are a significant advantage. Consequently, we also recommend including a detailed guidance how to apply the SOS 2.0 interface to VGI into a Best Practice Document about the interoperable exchange of VGI data. In addition we have shown that a mapping to the VGI SOS to the WFS 2.0 interface is possible. This facilitates the integration into existing client software which is not yet able to interact with SOS 2.0 servers.

The number of VGI resources found during the harvesting of Flickr images for the demonstration scenario was not very abundant. Few images were georeferenced and the image titles and tags were often not very meaningful. These issues also appear on other social platforms such as Twitter. We expect that this will further improve in the near future as cameras and mobile devices with built-in positioning functionality are getting more and more common. However, in the context of Testbed-10, manual processing steps had to be executed to prepare the VGI data for its use in the demonstration scenarios.

The lack of meaningful feature names and descriptions reduced the chances to find related VGI data by means of the linking process. The access to the mapped VGI data during the WPS linking process worked well.