Computers & Geosciences 87 (2016) 76–83

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Case study

A BPMN solution for chaining OGC services to quality assure location-based crowdsourced data

Sam Meek*, Mike Jackson, Didier G Leibovici

Nottingham Geospatial Institute, Nottingham Geospatial Building, University of Nottingham, Triumph Road, Nottingham NG7 2TU, United Kingdom

ARTICLE INFO

Article history:
Received 5 January 2015
Received in revised form
30 October 2015
Accepted 2 December 2015
Available online 4 December 2015

Keywords:
Interoperability
Web processing service
Quality assurance
Crowdsourcing
Citizen science

ABSTRACT

The Open Geospatial Consortium (OGC) Web Processing Service (WPS) standard enables access to a centralized repository of processes and services from compliant clients. A crucial part of the standard includes the provision to chain disparate processes and services to form a reusable workflow. To date this has been realized by methods such as embedding XML requests, using Business Process Execution Language (BPEL) engines and other external orchestration engines. Although these allow the user to define tasks and data artifacts as web services, they are often considered inflexible and complicated, often due to vendor specific solutions and inaccessible documentation. This paper introduces a new method of flexible service chaining using the standard Business Process Markup Notation (BPMN). A prototype system has been developed upon an existing open source BPMN suite to illustrate the advantages of the approach. The motivation for the software design is qualification of crowdsourced data for use in policy-making. The software is tested as part of a project that seeks to qualify, assure, and add value to crowdsourced data in a biological monitoring use case.

1. Introduction

Geographic Information Systems (GIS) have been employed for decades as tools for providing spatial intelligence to address issues across many academia, industry and government. With the growth of these systems, spatial problems have become more complex and often employ multiple steps, involve different processes, and disparate datasets. Combining datasets and algorithms to build up *workflows* has been approached in common GIS software, for example the ESRI suite has Model Builder[1] that allows a user to build up a workflow and run it unsupervised, or to use an API to interact with the scripts in ArcToolbox.

The ability to run complex processes in a chain is common among GIS software, other examples include the QGIS API[2] and MapInfo Mapbasic[3]. These approaches have drawbacks in that they are often proprietary, are sometimes confined to a desktop installation, and in the case of the APIs, require programming skills. A specifically built workflow engine without a typical GIS setup is Alteryx.[4] This enables chaining of many processes and offers the user the facility to create their own process workflows in the form of macros. Although very powerful, Alteryx is a proprietary desktop offering and therefore does not include standards based interfaces or centralizing process repositories beyond the confines of the organization. Another issue to consider with the aforementioned software offerings is cost, which can make them prohibitive for many organizations. Workflow engines are not unique to the geospatial community; domains such as medicine (Sutherland and van den Heuvel, 2006) and chemistry (Fernandez et al., 2011) have used workflow creation software to semi-automate tasks. Additionally there have been efforts to create high-performance workflow engines designed specifically for complicated, scientific problems such as Pegasus[5] and Kepler[6].

The Web Processing Service (WPS) 1.0.0 standard was ratified and adopted in 2007 and is a generic interface that wraps geospatial processes. The processes can vary from simplistic calculations that work on dataset attributes to complex tasks such as environmental modeling (Castronova et al., 2013). Recently, the WPS 2.0 standard was released, however it has not been used in this research due to the lack of freely available implementations at time of writing. The advantage of utilizing web-based standards such as WPS is that they can be accessed from anywhere using a variety of standard methods. A useful way of calling WPS

* Corresponding author.
E-mail addresses: sam.meek@nottingham.ac.uk (S. Meek), mike.jackson@nottingham.ac.uk (M. Jackson), didier.leibovici@nottingham.ac.uk (D. Leibovici).

[1] http://video.esri.com/watch/663/getting-started-with-modelbuilder
[2] http://qgis.org/api/
[3] http://www.mapinfo.com/product/mapinfo-mapbasic/
[4] http://www.alteryx.com/
[5] http://pegasus.isi.edu/
[6] https://kepler-project.org/

http://dx.doi.org/10.1016/j.cageo.2015.12.003
0098-3004/

processes is through a desktop GIS client in open source solutions like QGIS and proprietary software such as ArcGIS. There are several advantages to adopting a *standards* based approach to systems that include; little replication of work – a process can be created, stored and maintained centrally, and be utilized by multiple clients. Additionally executions are performed locally on each WPS server, reducing the client's software and library dependency requirements. Extension of interoperable approaches is described in the *Model Web* (Bastin et al., 2013) where there is a call for models to be exposed as web services for improving integration, increasing research transparency, enabling flexibility, and facilitating discovery for reuse of model components and code. Similarly, GEO-WOW focuses on interoperability of weather, ocean and water data within GEOSS (Zsoter, 2014).

Several governmental organizations have opted for standards compliant software, discovery services and data provision services. Salient examples of this include the work done for data services under the INSPIRE directive[7], a European Union wide effort to harmonize geographic data and make it available to the public, and the Global Earth Observations System of Systems (GEOSS)[8] that seeks to centralize discovery of worldwide data and services.

This paper describes a generic solution for chaining geographic processes built on interoperability principles. The motivating use case for undertaking this work is to provide a toolbox for qualifying data collected through crowdsourcing activities. Our approach to qualifying crowdsourced data is applied a biological monitoring use case as part of a European project and described in the second half of this paper.

2. Current approaches

The WPS standard has been adopted and exemplified in a variety of domains including, stream flow predication (Castronova et al., 2013), 3D processing (Lanig and Zipf, 2010), digital elevation model analysis (Lanig and Zipf, 2009), grid computing (Baranski, 2008; Giuliani et al., 2012), and radiation safety (Sagl et al., 2011). Within the WPS 1.0 standard there is a mandatory requirement for WPS process chaining, that is, using the outputs of one process in the next process, however there was no clear indication of how to make WPS processes suited to orchestration, this is evident by the lack of detail in the specification concerning WSDL or SOAP. Chaining processes together is a useful feature as in principle; it allows users to create complex workflows consisting of multiple processes that can be reused without the setup overhead. This is particularly beneficial in examples where the chained process workflow contains many processes with multiple inputs that are time consuming to configure, or in cases where the processes are complex and take time to complete. In the latter case, the user could start a service chain by setting up inputs, execute the process chain, and then return when the chain has completed. Additionally the user has the option of breaking down a complex task into simpler, atomic components that can be used individually, and assists in the workflow testing and editing process.

Process chaining is described within the WPS standard 1.0 document (Open Geospatial Consortium, 2007) that recommends that it should be accomplished in one of three ways:

1. Creating a process within a WPS that calls other WPS processes in a sequence.
2. Cascading service calls within an execute request.
3. A BPEL (Business Process Execution Language) engine.

[7] http://inspire.ec.europa.eu/
[8] http://www.epa.gov/geoss/

Calling processes within a WPS fulfills the remit of service chaining within a WPS and has the advantage of having no external dependencies, however it requires a new WPS process to be created and deployed each time a change is made. The simplest method of chaining processes is to cascade service calls, but again requires changes to the execute document each time a change has to be made. It is perhaps more flexible than generating processes that incorporate other processes but can also only create workflows that run sequentially.

2.1. BPEL

Business Process Execution Language (BPEL) is a language that is based upon XML and allows for execution, service chaining and task sharing over distributed computing networks (Fu et al., 2004). Utilizing BPEL for service chaining offers the advantage of a customizable workflow engine to manipulate and monitor services so that they are suited to different tasks. There are several examples of the use of BPEL workflow engines to chain OGC services including wrapping of GRASS GIS into a WPS and then using BPEL to organize the service chain (Brauner et al. 2009). Although successful and now available as a WPS extension[9], the example provided illustrates only a simple linear service chain with little support for reuse and customization, and error management. Additionally, the configurations of the service chains are restricted to a specific order, and only use the processes of a single WPS. Other examples include, post processing of LiDAR (Kang et al., 2010), post disaster image processing (Bielski et al., 2011), and production of thematic mapping (Rautenbach et al., 2013).

There have been multiple examples of the use of BPEL in this area; many of them employ the workflow engine to chain together disparate OGC services such as Web Feature Services (WFS), Web Mapping Services (WMS), Catalog Services for the Web (CSW) and Sensor Observation Services (SOS). However there are technical issues when utilizing BPEL in this way. One of the main issues is that BPEL relies on Web Services Description Language (WSDL) documents to execute workflows and these have to be created for every service in order. Additionally, BPEL scripts use Simple Object Access Protocol (SOAP), which has largely fallen out of favor for REST services and does not offer the ability to utilize notifications relying on polling (Weiser and Zipf, 2007).

Kiehle et al. (2007) found issues when implementing BPEL as a method of chaining OGC services. Some of the problems noted include a missing raw binary data type in the XML schema, WCS responses cannot be encoded directly in SOAP which requires workarounds to be implemented, some are outlined by Sonnet (2004), and Scholten et al. (2006). Brauner et al. (2009) report that there has been a counter movement to orchestrate service chaining beyond BPEL, and that is to use the WPS standard itself to utilize chaining. As of the recent WPS 2.0 candidate standard (Open Geospatial Consortium, 2014), there is no extra provision for service chaining beyond nested post requests, or chaining via process design (wrapping processes in processes) as described in WPS 1.0 and mention of BPEL has been removed. The change required to make OGC web services suitable for chaining using BPEL was the introduction of native SOAP message handling support, however implementation of SOAP messaging in WPS is optional[10], and only supported in WFS 2.0.0[11], and is marked as an extension, not a requirement in WCS[12]. Recent advances and trends in this area have indicated that SOAP is being dropped in favor of REST, this could be due to the relative lightweight requirements and

[9] https://wiki.52north.org/bin/view/Geoprocessing/TutorialBackendGRASSWin
[10] http://geoprocessing.info/wpsdoc/1x0Execute
[11] http://docs.geoserver.org/latest/en/user/services/wfs/basics.html
[12] http://www.opengeospatial.org/standards/wcs

flexibility of REST compared to SOAP (Wagh and Thool, 2012).

Friis-Christensen et al. (2009) outline seminal work on OGC process chaining, they describe four approaches:

- Centralized control flow pattern – the workflow description is hard coded
- Client controlled service chain – the client makes the workflow description calls and the variables are parameterized for user interaction
- Workflow engine controlled service chain – the workflow and visualization clients are independent and the data produced by the workflow can be reused in further service chains
- Cascaded control flow pattern – the client initiates the WPS calls in one execution that are carried through the workflow and instantiated where required

The authors make a good case for using a workflow engine based approach for multiple reasons including but not limited to: code mobility, reusability of workflows, data flow visibility, reproducibility and workflow sharing[13]. We agree broadly with their assessment of methods with a few additions discussed in the following section.

In addition to BPEL there has been other work done in service chaining by using the Taverna workflow orchestration engine (de Jesus et al., 2012), or via RichWPS[14]. This approach relies on WDSL and SOAP protocols although offers a much improved method of orchestrating WPS than BPEL.

3. A BPMN solution

Business Process Modeling Notation (BPMN)[15] is a standard that has many parallels with BPEL, notably that they are both designed to model business processes. However, there are several differences between the standards, an important divergence of note is BPEL is focused on the low-level execution environment where BPMN excels is in modeling processes visually, allowing non-domain experts to communicate and mutually understand their models (Leymann, 2011). This suggests that a BPMN solution should enable users to employ complex processes chained together from disparate services though an intuitive graphical interface. This has been seen with some success with Unified Modeling Language (UML) as a tool for interdisciplinary communication of research (Kobryn, 1999) (Larsen et al., 2009). Chaining non-specific web services has been approached by Weber et al. (2013), where their focus was on providing the user with suitable process names and mapping the input and output messages to forms.

As described in the introduction, our motivating use case for this chaining software is to provide tools to build workflows for qualifying crowdsourced data. A graphical based approach to creating workflows to qualify incoming data is appropriate for crowdsourcing or citizen science project stakeholders. There is a wide range of use cases that the software is designed for, and some qualification techniques are domain specific. The target user base for the application are project stakeholders, who are likely to have some knowledge of what constitutes a *qualified record* in their domain, but may not necessarily have programming skills, although it is acknowledged that stakeholders are likely to require training in how to create workflows and operate the software.

BPMN offers similar functionality as BPEL and there are methods to convert from one to the other. However due to the standardized graphical notation available in BPMN, workflows expressed as diagrams offer semantic interoperability among users and modelers. The diagrams along with the XML encoding can be shared as metadata linked to lineage (Lanter, 1991) or provenance (Buneman et al., 2000) for the data outputs that have been processed using a service chain. This also allows for discovery of multiple WPS that are not necessarily co-located, opening up the possibility of interoperability with other services such as those provided by GEOSS. From a user perspective, BPEL does not have a standardized graphical notation to represent its processes, which has caused some fractures within the community, as different organizations often create their own. In contrast to this, BPMN is standardized and contains easily recognizable processes to enable the stakeholder to author workflows that can be carried across any BPMN based system (2).

3.1. System design

The generic requirements for a chaining system have been outlined in the previous sections, however the implementation is influenced by the requirements for qualification of crowd observations in the Citizen Observatory WEB (COBWEB)[16], a project that is concerned with the utilization of crowd collected data to update or augment authoritative data for use in policy creation and decision-making.

Our solution to the chaining problem of OGC web services is to implement a BPMN workflow engine that can call processes that are held in one or more WPS acting as a service repositories. Workflows are configured via a web editor and then saved as *deployments* that can be run via the editor console, or called via REST API.

When the workflow is run, the engine creates and executes a request to the WPS using the input data provided by the user and then waits for the results. The results are then passed on to the next process in the chain to be used as an input to that process with outstanding variables defined in the editor and another execute request generated. The design of the system allows the user to chain any process from any web facing WPS, providing the input and output formats are known and supported by the client. The chaining of processes according to a defined order is coordinated in a linear fashion in terms of how they are executed; however the workflows themselves can maintain parallel branches and concurrent result generation. Fig. 1

Our system is built upon existing open source implementations of the WPS and BPMN standards. The WPS and client libraries are by 52 North[17], and the BPMN implementation is JBPM[18], which is maintained by JBOSS[19]. Both of the pieces of software are Java based and run on Java application servers, the WPS runs on Apache Tomcat[20] and JBPM is deployed on JBOSS Wildfly[21]. A UML diagram of the implementation is shown in Fig. 2.

A JBPM solution to chaining is workflow oriented but somewhat different to the system proposed by Fris-Christensen et al. (2009). Their solution has a client application that enables viewing of the results of the workflow and an engine that performs the processing. Our solution differs in that the workflow engine makes use of Geotools[22] FeatureCollection[23] to facilitate the passing of

[13] http://www.myexperiment.org/home
[14] https://richwps.github.io/
[15] www.omg.org/bpmn
[16] https://cobwebproject.eu/
[17] http://52north.org/wps
[18] http://www.jbpm.org/
[19] http://www.jboss.org/
[20] http://tomcat.apache.org/
[21] http://wildfly.org/
[22] http://www.geotools.org/
[23] http://docs.geotools.org/stable/javadocs/org/geotools/feature/FeatureCollection.html

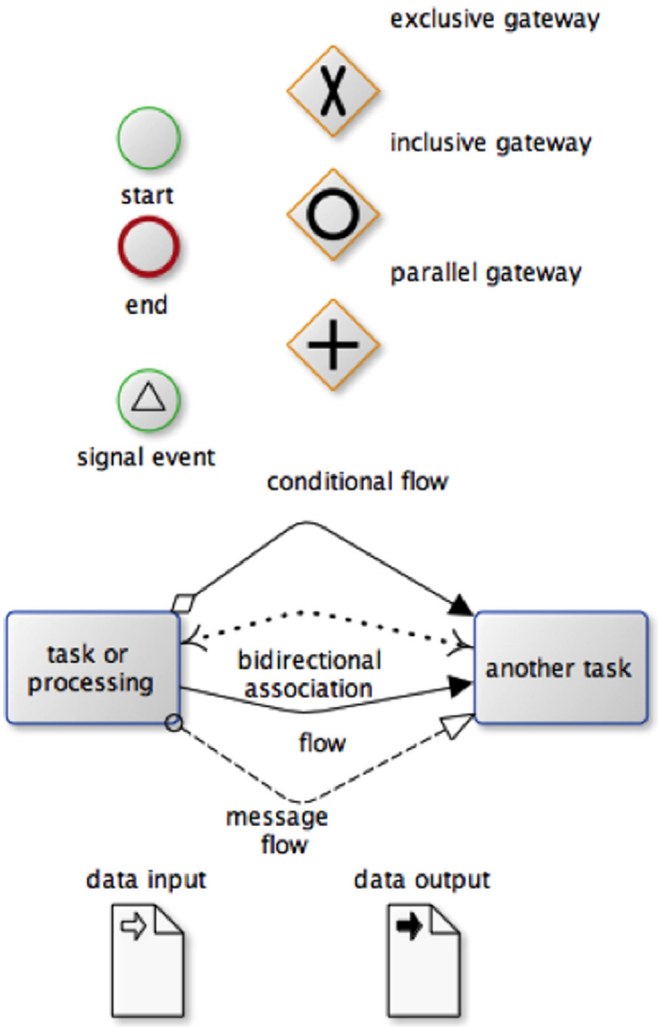

Fig. 1. Basics of a BPMN diagram.

intermediate results to other processes in the workflow, this implies that many of the data inputs are facilitated via WFS and are therefore vector based. The WPS acts as a process repository with processes that can be utilized as standalone services in interoperable software, the workflow engine acts as another client to the WPS.

The graphical interface on the workflow engine enables the user to provide endpoints for the data inputs and values for the variables that each process requires. JBPM includes the option to create *Custom Work Items,* defined by a MVEL (MVFLEX Expression Language) .wid file; these can be customized to run executable pieces of code while providing handlers for the input and outputs. In our system, each *Custom work Item* constitutes one reusable WPS process. Each work item can be called multiple times and in any order with different inputs instances. Fig. 3 shows the DescribeProcess XML document snippet and the corresponding entry in a MVEL custom work item file. Most of the data inputs and types are taken from the process description; the URL parameter is defined as the URL where the processes are located.

This is handled via a generic WPS client that sits between the work items and the WPS. The client takes the inputs from the work item and translates them into HTTP Post requests, it is also at this point where the work item provides the generic WPS client with either a reference to a dataset (usually a WFS endpoint) or if a process is utilizing an output from a previous output, it will parse the data directly, the client also provides a handler for the response of the output(s). The input data and modified data are generally vector datasets; therefore these are handled via the Geotools FeatureCollection class. All numerical inputs are handled as StringDataType and then parsed to the appropriate data type server side. The workflow engine will also handle raster datasets by calling via Web Coverage Service (WCS).

4. Case study: Quality assurance in crowdsourcing

A prevalent issue within crowdsourcing and citizen science is the ability to validate and quality control the data collected. Data captured by citizens often lacks metadata about its quality and may incorporate deliberate bias and disinformation, which results in many scientists disregarding it (Alabri and Hunter, 2010) but contrarily it can frequently complement or update authoritative surveys (Jackson et al., 2010). There are several methods of gaining knowledge about the quality of citizen collected data, they include; using a majority decision or control group (Hirth et al., 2012), using a reputation system (Alabri and Hunter, 2010), (Clow and Makriyannis, 2011), and using mobility patterns and previous contribution quality (Mashhadi and Capra, 2011).

Currently, full utilization of data collected through crowdsourcing activities is not achieved. This is potentially due to a deficiency of trust in the quality and the validity of citizen contributions. Types of data quality can be split into two categories, internal quality, which refers to aspects such as completeness, positional accuracy and consistency, and external quality. A full set of internal quality metrics is defined by ISO 19157 (ISO, 2013). External quality describes whether the data collected are fit for purpose by the user of the data product (Wang et al., 1996). Frank (2008) describe a metadata schema in the GeoViQua[24] project named the “consumer model” that is useful for crowdsourcing projects as it seeks to qualify the usability of the data. Additionally, Meek et al. (2014) describe the “stakeholder model” that aims to qualify the data collector’s judgment, trust and reliability.

The Citizen Observatory WEB (COBWEB) is a EU funded FP7 project involving 13 partners from 5 European countries. The project aims to utilize sources of data collected from the crowd to qualify and verify authoritative data in three use cases:

1. Biological monitoring
2. Earth observation enhancement
3. Flooding

The types of data that COBWEB seeks to qualify ranges from citizen science observations made using an array of mobile data collection applications, harvesting embedded sensor information, and utilizing techniques to compare observations with authoritative data. The study areas for COBWEB are four UNESCO biospheres located in three countries, Wales, Germany and Greece. The first deployment of the COBWEB system is in the Dyfi Biosphere[25] in Central and Western Wales.

One of the main aims of COBWEB is to research and test a system that enables these datasets to be qualified to a degree that they could be utilized in local and national policy making. The three use cases over the four considered biospheres have different requirements for a dataset to be considered *quality assured*, therefore the system that undertakes these processes must meet certain requirements including; flexibility, range, standards-based to interoperate with GEOSS and INSPIRE, and be sufficiently powerful to process large datasets from a variety of sources.

[24] http://www.geoviqua.org/
[25] http://www.biosfferdyfi.org.uk/

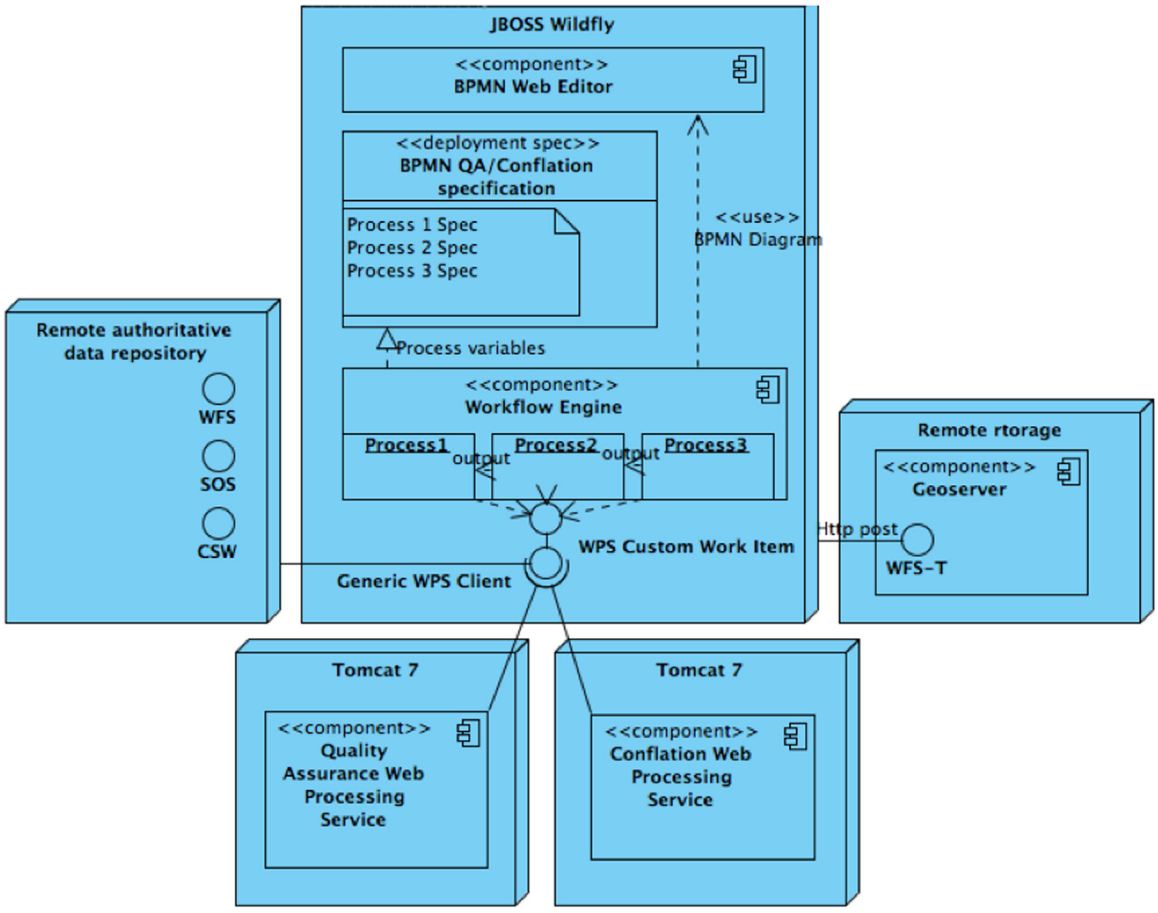

Fig. 2. BPMN workflow system design.

4.1. Qualification process design

Quality assurance (QA) workflows are configurable so that stakeholders are able to design a solution to fit their use case from a set of generic WPS processes. To do this, the processes are designed in a way to enable maximum flexibility within the system.

Fig. 4 describes the design of a generic WPS QA process within the qualification system. The process takes a set of inputs such as authoritative data and other crowdsourced data that are required to function. In the case of a simple buffer algorithm, these inputs constitute the dataset of features to be buffered, and the radius of the buffer. The outputs include the process result, i.e. whatever the process does to the data, in the case of a simple buffer algorithm, the modified data would be the buffered features. The modified input data is the same dataset as the input data with attributes added if applicable. With the buffer algorithm, this could be the size of the buffer applied. The metadata output varies depending on the quality elements that the process modifies or generates, for example a quality element maybe whether the generated buffer overlaps an authoritative dataset indicating an observation's relative location.

The aim of the design is to be flexible so that it can fit a number of use cases, a simple example is that a process may be included in a chain because the user wants to generate metadata for the output and not alter the dataset contents.

The BPMN based toolkit described in this paper provides

```
<ows:Identifier>PointInBuffer</ows:Identifier>
<ows:Title>Point In Buffer</ows:Title>
    <DataInputs>
        <Input maxOccurs="1" minOccurs="1">
            <ows:Identifier>inputObservations</ows:Identifier>
            <ows:Title>inputObservations</ows:Title>
                <ComplexData>
                    <Default>
                        <Format>
                            <MimeType>text/xml; subtype=gml/3.1.1</MimeType>
                            <Schema>
                                http://schemas.opengis.net/gml/3.1.1/base/gml.xsd
                            </Schema>
                            </Format>
                    </Default>
                    <Supported>
                    <Format>
                            <MimeType>text/xml; subtype=gml/3.1.1</MimeType>
                            <Schema>
                                http://schemas.opengis.net/gml/3.1.1/base/gml.xsd
                            </Schema>
                            </Format>
                    </Supported>

                </ComplexData>
```

```
[
    "name" : "PointInBuffer",
    "parameters" : [
          "URL" : new StringDataType(),
          "processDescription" : new StringDataType(),
          "inputObservations" : new FeatureCollection(),
          "inputAuthoritativeData" : new FeatureCollection(),
          "inputBufferDistance" : new StringDataType(),
          "outputFeatures" : new FeatureCollection(),
    ],

    "icon" : "icons/world.png",
   "displayName" : "PointinBuffer"

]
```

Fig. 3. Left. ProcessDescription XML file. Right. Corresponding MVEL file.

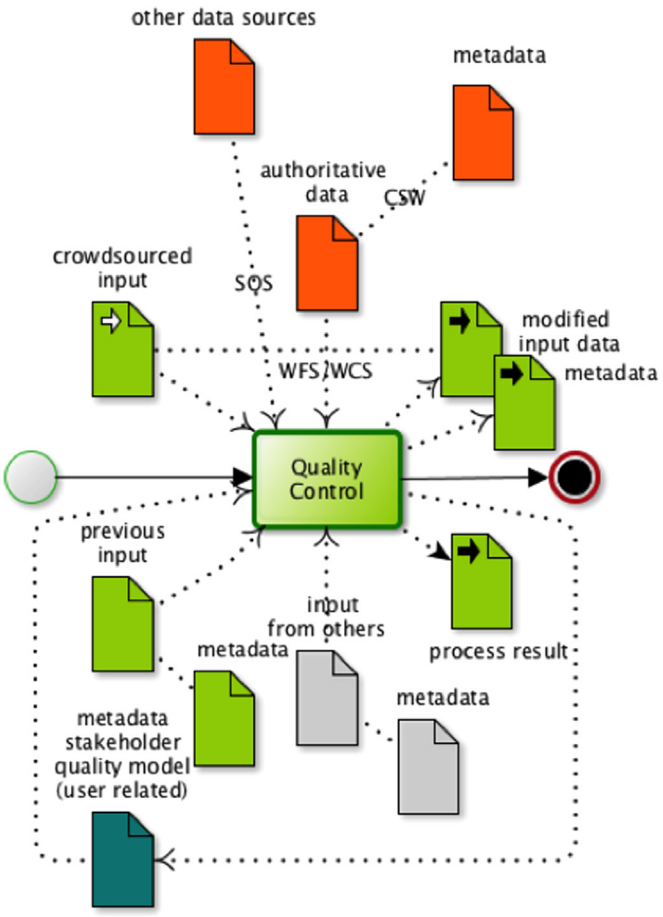

Fig. 4. Generic WPS process design.

Fig. 4; they therefore all return the modified input data, the process result and metadata generated within the process.

A concern for the system for qualifying crowdsourced data is the generation and storage of metadata. In terms of authoritative data, metadata is usually collected at the *dataset* level, for example in the Ordnance Survey product MasterMap[26]. However, the concept of a *dataset* is vague with crowdsourced data as individual records are likely to vary in quality with few suitable groupings to form a traditional dataset. With data collected by citizens using a mobile device, the devices differ; the citizens have differing levels of expertize, and experience varying environmental conditions. It is for these reasons that metadata is required to be collected and generated at the *feature* level, therefore the metadata records are tightly coupled to the data. In terms of OGC services, metadata is usually held in a CSW, however CSW is usually used to discover data at the dataset level, rather than the feature level. In COBWEB the metadata is attached as extra attributes and stored in a database, this is accomplished via a Transactional Web Feature Service (WFS-T) that extends the schema of the qualified data to accommodate fields from ISO 19157, eventually to be extended to the GeoViQua producer model (Papeschi et al., 2014) and OGC Observations and Measurements[27].

Fig. 5 shows an implementation of a workflow to qualify data post-capture from a mobile application designed to support a biological monitoring use case. The scenario is that there is a citizen science project running in an area that is looking at recording tree species and residents in the local area have an application on their smartphone that allows them to record a point location of different tree specimens. The QA process in this instance is used to qualify and isolate records that refer to Ash trees, as there is fear of Ash Dieback in the area (Halmschlager and Kirisits, 2008).

The workflow calls on two separate WPS end points, one for conflation and one for QA. Most of the processes generate metadata, the exception being the conflation process. The metadata is recorded and held as part of the outputs of the process; in this

Table 1
Implemented WPS processes.

Pillar	Example process	Example description
Pillar 1 - Location Based Services - Positioning	Determine spatial accuracy	Uses the position data recorded on the device to assess spatial accuracy, e.g. • Number of satellites • Phone's estimate of accuracy
	Get line of sight coordinates	Takes the device's position and orientation data and uses a surface model to determine what the device is pointing at (techniques described in Meek et al., 2013).
Pillar 2 - Cleaning	Attribute range check	Assesses a given attribute field for a numerical range
	Filter on attribute	Filters out records based upon a field name and list of acceptable (or unacceptable) values.
	Position on bounding box	Filters out records based upon a bounding box, either in or out.
Pillar 4 - Comparison with authoritative data	Point in polygon	Checks to see whether a record falls within a given set of polygons.
	Point in buffer	Similar to point in polygon but allows the use to enter a tolerance for the position of a record.
	Number of neighborhood named features	Uses a given distance and checks for the number of features with a specific name within that distance.
	Number of same features in the neighborhood	Uses a given distance and checks for the number of features named by a given field.
Pillar 6: Linked data analysis	Count numbers of Tweets	Counts the number of geolocated Tweets with a provided hash tag, location, date since and radius.

knowledgeable stakeholders the ability to create workflows that can qualify data inputs that are suited to their use case. In this implementation of the system, the stakeholder has the choice of several different processes to qualify their data. The processes are registered in a WPS and are classified according to the *Seven Pillars* of quality assurance outlined in Meek et al. (2014). Examples per pillar of those processes are shown in Table 1. Each of the processes can be used as part of a BPMN orchestrated workflow or can be used as single processes for a variety of WPS clients including typical desktop GIS. Each of the processes is designed according to simplified case the metadata records have a binary output, 1 if the observation conforms to the input parameters, and 0 if it does not. Additionally the user has the option of filtering out records that do not conform to the input parameters, this is useful in cases such as removing all records that are outside the area of interest, for example.

[26] http://data.gov.uk/dataset/os-mastermap-topography-layer
[27] http://www.opengeospatial.org/standards/om

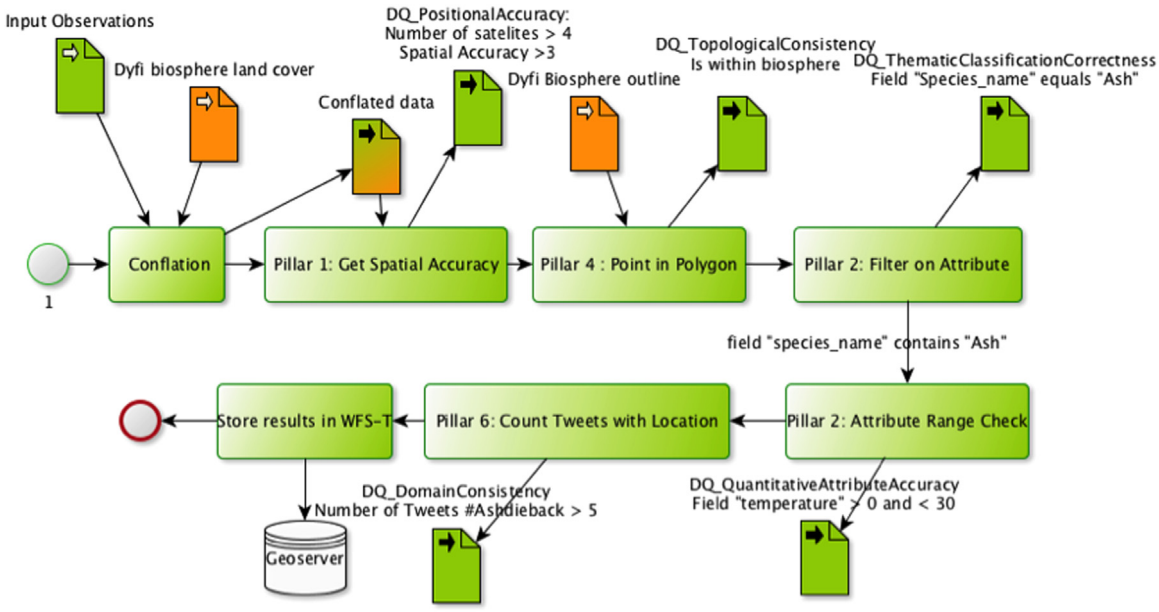

Fig. 5. A typical biological monitoring workflow in COBWEB.

The processes are chained into a suitable order within the editor and are then deployed to the workflow engine as tasks that can be executed at the stakeholder's request. This is currently a manual task, as the user of the authoring tools has to press a button in the editor to instantiate the task. However tasks can be executed through other methods including REST API or the workflows can be re-wrapped into a WPS process and then entire workflows could be chained if appropriate.

The workflow uses the data captured by participants in the field and marks each record with a metadata entry about the quality of that record. These metadata records include information such as the spatial accuracy of each observation and whether each observation falls within a study area. In this instance, the workflow also includes a conflation process based upon each observation's distance relationship to an authoritative dataset (Wiemann and Bernard, 2014), a record of previous flood warning areas. This conflation process is deployed in a different WPS to the rest of the QA processes and is initiated by the workflow engine passing the WPS a reference to an observation dataset and an authoritative dataset, both WFS, and a distance threshold.

In this workflow, the records are first conflated with a dataset of land cover and a relationship field is recorded in the resultant dataset. The results of this are passed to the second process *Get Spatial Accuracy*, and the metadata attributes are created and populated for DQ_SpatialAccuracy. The *Filter on Attribute* process identifies all records that that refer to ash trees and unlike the other processes in this workflow, filters them out. The *Point in Polygon* process identifies through the metadata all of the records that do not fall within the biosphere and the *Attribute Range Check* process flags all of the records in the metadata that have a recorded temperature on the device of above 30 degrees Celsius, as this is deemed to be improbable for the autumn in the UK and indeed the Dyfi Biosphere. The *Count Tweets with Location* process uses the Twitter API to search for instances of the word #Ashdieback with a radius of 10 miles of our study area and records in the metadata for those records whether there were more than 5 Tweets with this hash-tag in the last 24 hours. The Twitter process is experimental and is shown here, as a demonstration of the variety of processes that can be utilized in the QA workflow, and its utility in qualifying records requires further investigation. Finally the results of the process are uploaded to a WFS-T where they are made publically available for viewing and manipulation. Additionally the records can be used as an input to another workflow should a stakeholder require it.

5. Conclusion

In this paper we have addressed priority research topics (i) a design and implementation of a method of chaining WPS processes and (ii) the need for a standards-based methodology for addressing the validation of crowdsourced spatial data and its quality assurance. The first has been a long-standing issue within the standards community where we believe that our approach goes some distance to addressing the problem of OGC service orchestration and chaining. Quality assuring crowdsourced data is a critical requirement if it is to be reliably used for modeling, decision-making and policy. However we are aware that our introduction of the BPMN standard as a method of orchestrating the chaining is not OGC standards compliant. Additionally there are other workflow paradigms such as Taverna (de Jesus et al., 2012) that have also met with success. Nevertheless, our approach is tested in a challenging use case and has met with success in workflow process outputs and in terms of performance.

We have several challenges ahead to iterate and improve the system involving editor usability, WPS and process discovery and organization, semantic clarification of the processes, and best practices for implementing and using the software. Our future research is concerned with expanding on crowdsourced data quality control and assurance to take into account wider sources of data such as web crawling, image content analysis and further use of social media sources. Another challenge with this approach is managing and understanding error propagation through the processes (Heuvelink, 1998).

We believe that our approach to chaining WPS processes that have the ability to incorporate multiple OGC services is a viable way forward for future research in this area. Our paper has focused on a case study of quality control for crowd collected data within a UNESCO biosphere, but the general approach to chaining services can be applied in other situations that require processes to be atomic and singularly executable but also part of a workflow.

Acknowledgments

This work has been part supported by the project "Citizen Observatory WEB" (COBWEB) funded by the European Union under the FP7 ENV.2012.6.5-1 funding scheme, EU Grant Agreement Number: 308513.

References

Alabri, A., Hunter, J., 2010. Enhancing the quality and trust of citizen science data. In: Proceedings IEEE Sixth International Conference on e-Science Brisbane, Australia. pp. 81–88.

Baranski, B., 2008. Grid computing enabled Web Processing Service. GI-Days Münst. Ger., 12.

Bastin, L., Cornford, D., Jones, R., Heuvelink, G.B.M., Pebesma, E., Stasch, C., Nativi, S., Mazzetti, P., Williams, M., 2013. Managing uncertainty in integrated environmental modeling: tthe UncertWeb Framework. Environ. Model. Softw. 39, 116–134.

Bielski, C., Gentilini, S., Pappalardo, M., 2011. Post-disaster image processing for damage analysis using GENESI-DR, WPS and Grid computing. Remote. Sens. 3 (6), 1234–1250.

Buneman, P., Khanna, S., Tan, W. C., 2000. Data provenance: Some basic issues. In: Proceedings Foundations of Software Technology and Theoretical Computer Science 2000. New Delhi, India. pp. 87–93.

Brauner, J., Foerster, T., Schaeffer, B., Baranski, B., 2009. Towards a research agenda for geoprocessing services. In: Haunert, J., Kieler, B., Milde, J. (Eds.), 12th AGILE International Conference on Geographic Information Science, Hanover, Germany.

Castronova, A.M., Goodall, J.L., Elag, M.M., 2013. Models as web services using the open geospatial consortium (OGC) web processing service (WPS) standard. Environ. Model. Softw. 41, 72–83.

Clow, D., Makriyannis, E., 2011. iSpot Analysed: Participatory learning and reputation. In: Proceedings of the 1st International Conference on Learning Analytics and Knowledge. Banff, Canada. pp. 34–43.

de Jesus, J., Walker, P., Grant, M., Groom, S., 2012. WPS orchestration using the Taverna workbench: the eScience approach. Comput. Geosci. 47, 75–86.

Fernandez, H., Tedeschi, C., Priol, T., 2011. A chemistry-inspired workflow management system for scientific applications in clouds. In: Proceedings 7th International Conference on e-Science. Stockholm, Sweden. pp. 39–46.

Friis-Christensen, A., Lucchi, R., Lutz, M., Ostländer, N., 2009. Service chaining architectures for applications implementing distributed geographic information processing. Int. J. Geogr. Inf. Sci. 23 (5), 561–580.

Fu, X., Bultan, T., Su, J., 2004. Analysis of interacting BPEL web services. In: Proceedings of the 13th international conference on World Wide Web, New York, USA, pp. p621–630.

Giuliani, G., Nativi, S., Lehmann, A., Ray, N., 2012. WPS mediation: an approach to process geospatial data on different computing backends. Comput. Geosci. 47, 20–33.

Halmschlager, E., Kirisits, T., 2008. First report of the ash dieback pathogen Chalara fraxinea on Fraxinus excelsior in Austria. Plant Pathol. 57 (6), 1177.

Heuvelink, G.B., 1998. Error propagation in environmental modelling with GIS. CRC Press.

Hirth, M., Hoßfeld, T., Tran-Gia, P., 2012. Analyzing costs and accuracy of validation mechanisms for crowdsourcing platforms. Math. Comput. Model. 57 (11), 2918–2932.

ISO, 2013. ISO 19157 – Geographic Information – Data Quality. International Standards Organisation.

Jackson, M.J., Rametulla, H., Morley, J., 2010. The Synergistic Use of Authenticated and Crowd-Sourced Data for Emergency Response. In: Proceedings of International Workshop on Validation of Geo-Information Products for Crisis Management (VALgEO). pp. 91–99.

Kang, L., Wu, Q., Yuan, Y., 2010. A distributed LIDAR processing model based on OWS and BPEL. In: Proceedings International Geoscience and Remote Sensing Symposium. Honolulu, USA. pp. 3628–3631.

Kiehle, C., Greve, K., Heier, C., 2007. Requirements for next generation spatial data infrastructures-standardized web based geoprocessing and web service orchestration. Trans. GIS 11 (6), 819–834.

Kobryn, C., 1999. UML 2001: a standardization odyssey. Commun. ACM 42 (10), 29–37.

Lanig, S., Zipf, A., 2009. Interoperable processing of digital elevation models in grid infrastructures. Earth Sci. Inform. 2 (1–2), 107–116.

Lanig, S., Zipf, A., 2010. Proposal for a web processing services (WPS) application profile for 3D processing analysis. In: Proceedings Second International Conference on Advanced Geographic Information Systems, Applications, and Services, St. Maarten, Netherlands Antilles, pp. 117–122.

Lanter, D.P., 1991. Design of a lineage-based meta-data base for GIS. Cartography and Geographic. Inf. Syst. 18 (4), 255–261.

Larsen, T.J., Niederman, F., Limayem, M., Chan, J., 2009. The role of modeling in achieving information systems success: UML to the rescue? Inf. Syst. J. 19 (1), 83–117.

Leymann, F., 2011. BPEL vs. BPMN 2.0: Should you care? Lect. Notes Bus. Inf. Process. 67, 8–13.

Mashhadi, A.J., Capra, L., 2011. Quality control for real-time ubiquitous crowdsourcing. In: Proceedings of 2nd International Workshop on Ubiquitous Crowdsouring. Bejing, China. pp. 5–8.

Meek, S., Priestnall, G., Sharples, M., Goulding, J., 2013. Mobile capture of remote points of interest using line of sight modeling. Comput. Geosci. 52, 334–344.

Meek, S., Jackson, M.J., Leibovici, D.J. (2014) A Flexible Framework for Assessing the Quality of Crowd-Sourced Data. In: Proceedings of 17th AGILE Conference on Geographic Information Science. Castellion, Spain.

Open Geospatial Consortium, 2007. OpenGIS Web Processing Service. OGC Ref Num OGC 05-007r7 version 1.0. Status: OGC Standard.

Open Geospatial Consortium, 2014. OpenGIS Web Processing Service. OGC Ref Num OGC 14-065 version 2.0. Status: OGC Candidate Standard.

Papeschi, F., Bigagli, L., Masó, J., 2014. Designing and Implementing a Quality Broker: the GeoViQua Experience. In: Proceedings EGU General Assembly, Vienna, Austria.

Rautenbach, V., Coetzee, S., Iwaniak, A., 2013. Orchestrating OGC web services to produce thematic maps in a spatial information infrastructure. Comput. Environ. Urban Syst. 37, 107–120.

Sagl, G., Lippautz, M., Resch, B. and Mittlboeck, M., 2011. Near Real-Time Geo-Analyses for Emergency Support: A Radiation Safety Exercise. In: Proceedings of the 14th AGILE International Conference on Geographic Information Science, Utrecht, The Netherlands.

Scholten, M., Klamma, R., Kiehle, C., 2006. Performance evaluation of spatial data infrastructures for geoprocessing. Internet Comput. 10 (5), 34–41.

Sonnet, J., 2004. OWS 2 Common Architecture: WSDL SOAP UDDI. WWW document, https:// portal.opengeospatial.org/files/index.php?artifact_id=8348.

Sutherland, J., van den Heuvel, W. J., 2006. Towards an intelligent hospital environment: adaptive workflow in the OR of the future. In: Proceedings of the 39th Annual Hawaii International Conference on System Science. Honolulu, USA.

Wagh, K., Thool, R., 2012. A comparative study of soap vs rest web services provisioning techniques for mobile host. J. Inf. Eng. Appl. 2 (5), 12–16.

Wang, R.Y., Strong, D.M., Guarascio, L.M., 1996. Beyond accuracy: what data quality means to data consumers. J. Manag. Inf. Syst. 12 (4), 5–33.

Weber, I., Paik, H.,Y., Benatallah, B., 2013. Form-based web service composition for domain experts. ACM Trans. Web 8 (1), 2.

Weiser, A., Zipf, A., 2007. Web service orchestration of OGC web services for disaster management. Web Wirel. Geogr. Inf. Syst. Lect. Notes Comput. Sci. 4857, 239–251.

Wiemann, S. Bernard, L., 2014 Linking crowdsourced observations with INSPIRE. In: Proceedings of 17th AGILE Conference on Geographic Information Science 2014. Castellion, Spain.

Zsoter, E., 2014. GEOWOW – Benefits of TIGGE Ensemble Forecast Data for the GEOSS Community. EGU General Assembly, Vienna, Austria, p. 2014.