

WPS ENABLED SDI: AN OPEN SOURCE APPROACH TO PROVIDE GEOPROCESSING IN WEB ENVIRONMENT

A. K. Tripathi^{1,*}, S. Agrawal², R. D. Gupta³

¹ GIS Cell, Motilal Nehru National Institute of Technology Allahabad, Prayagraj -211004, Uttar Pradesh, India - rgi1551@mnnit.ac.in

² GIS Cell, Motilal Nehru National Institute of Technology Allahabad, Prayagraj- 211004, Uttar Pradesh, India - sonam@mnnit.ac.in

³ Civil Engineering Department, Motilal Nehru National Institute of Technology Allahabad, Prayagraj -211004, Uttar Pradesh, India - rdg@mnnit.ac.in

Commission V, WG V/7 & Commission IV, WG IV/6

KEY WORDS: Spatial Data Infrastructure (SDI), Web Processing Service (WPS), GeoServer, Open Geospatial Consortium (OGC), Geoprocessing

ABSTRACT:

Sharing and management of geospatial data among different communities and users is a challenge which is suitably addressed by Spatial Data Infrastructure (SDI). SDI helps people in the discovery, editing, processing and visualization of spatial data. The user can download the data from SDI and process it using the local resources. However, large volume and heterogeneity of data make this processing difficult at the client end. This problem can be resolved by orchestrating the Web Processing Service (WPS) with SDI. WPS is a service interface through which geoprocessing can be done over the internet. In this paper, a WPS enabled SDI framework with OGC compliant services is conceptualized and developed. It is based on the three tier client server architecture. OGC services are provided through GeoServer. WPS extension of GeoServer is used to perform geospatial data processing and analysis. The developed framework is utilized to create a public health SDI prototype using Open Source Software (OSS). The integration of WPS with SDI demonstrates how the various data analysis operations of WPS can be performed over the web on distributed data sources provided by SDI.

1. INTRODUCTION

In today's geospatial world, the generation of geospatial data is continuously increasing, which creates pressure on the geospatial communities to maintain and manage it (Tripathi et al., 2020, 2018). Spatial Data Infrastructure (SDI) is designed and developed as a solution to this, and can be defined as the system which encompasses the policies, technologies, standards and human resources necessary for the effective collection, management, access, delivery and utilization of geospatial data in a global community (Coleman and McLaughlin, 1998). It is a geospatial framework that facilitates and coordinates the exchange and sharing of geospatial data among different organizations (Rajabifard et al., 2002). Vancauwenberghe and van Loenen (2018) suggest the concept of open SDI that includes more stakeholders from the businesses, citizens and other non-governmental actors. SDI offers geospatial services through a standardized interface which can be enabled by Service Oriented Architecture (SOA) concept (Agrawal and Gupta, 2017; Li et al., 2018). SOA provides the standard based distributed computing to enable the reusability and interoperability of different component of the system.

The interoperable cartographical web services are the core of any SDI. Most of them are Open Geospatial Consortium (OGC) compliant services that work according to the SOA's publish, find and bind paradigm (Yue et al., 2016). OGC is the consortium which is responsible for defining and developing the standards for geospatial web services. It promotes interoperability and reusability. In SDI, various OGC services

are used. OGC Web Map Service (WMS) is deployed for visualization (de la Beaujardiere, 2006), OGC Web Feature Service (WFS) is used for vector data download and editing (Vretanos, 2005), and OGC Web Coverage Service (WCS) is provided for coverage or raster data (Baumann, 2012). Different Open Source Software (OSS) like GeoServer, MapServer and GeoNetwork are available to adopt such OGC services.

Foundation of SDI lies in data discovery and accessibility (Giuliani et al., 2017). This is provided through Catalogue Service for the Web (CSW) and Web Processing Service (WPS). In SDI, CSW is used for the publication and discovery of data. CSW provides a standard procedure through which data providers can publish their data and users can discover the data (Nebert et al., 2007). It gives the metadata catalog or registry through which data about geospatial data and services are published and searched. In SDI, WPS provides a standard interface through which geospatial process can be accessed. WPS gives the rules that standardize the input of execution parameters for geoprocessing web services or analysis (Schut, 2007). The common difference between the WPS on server and client is that the first one is accessed remotely and can be easily integrated with other computational platforms while the second one is accessed in a standalone manner. Cloud computing and grid computing are the technologies that can merge with different WPS implementation tools to deal with large and variant geospatial data (Singh et al., 2019).

Some of the standard OSS WPS specification implementation tools are 52⁰ North project, Zoo project, Deegree project,

* Corresponding author

PyWPS Project and Geoserver WPS. Several studies on these WPS platforms are performed. An implementation of OGC WPS by using 52⁰ North WPS has been demonstrated into geospatial mass-market application on use case scenario of forest fire risk assessment (Foerster et al., 2009). The integration of GeoPipes in the WPS interface is introduced to process the real time geoprocessing applications such as risk management or alerting services (Herle and Blankenbach, 2018). An implementation of hydrological model to predict watershed runoff, hosted on WPS server, is performed using PyWPS (Castronova et al., 2013). Béjar et al. (2012) developed SDI based business process for territorial analysis. While there are researches implementing OGC WPS services in different geospatial application but these are more focused on their data modeling and implementation of system.

OGC services like WMS, WFS, WCS and CSW can only distribute the vector and raster data to the client. In SDI, users may require to transform the data for which spatial analysis can be performed using Commercial Off-The Shelf Software (COTS) or OSS. Software used by the client can be platform dependent as well as GIS skills are required to perform the geoprocessing on it. These problems highlight the need for WPS enabled SDI that empowers the client to access the data from anywhere and from any machine that has internet connection along with geoprocessing capabilities. This would help the client to process the data without the need of a GIS software on client side. Therefore, WPS enables SDI is an important research issue. This SDI will not only give the maps and information but also provides the functionalities which are wrapped into WPS. The integration of WPS with SDI will also provide support in decision making. However, very less research has been found that focused on taking OGC WPS services into SDI. The present work is focused on the development of a WPS enabled SDI framework.

There is a large amount of health related data but due to the unavailability of common platform, it is difficult to discover it. This results in the generation of same data again leading to data duplication. As compared to other fields, the deployment of WPS for the health sector is not common. Therefore there is a need of WPS services that support the health sector. This work is an attempt to enhance the limited research on health SDI by implementing WPS and other OGC compatible services along with the Graphical User Interface (GUI) for the common public by developing a WPS enabled health SDI prototype.

The main objective of this paper is to design and develop a OGC compliant WPS enabled SDI framework so that the user can get the capability to perform geoprocessing over the web. This research demonstrates the orchestration of WPS into SDI using OSS. Further, OGC compliant prototype model of public health SDI is created based on developed WPS enabled SDI framework. This prototype will be useful for different potential users like patients, health agencies, and decision makers to access the health related geospatial data as per their requirements.

2. WPS

WPS is one of the interoperability standards promulgated by OGC. In year 2005, WPS initial version 0.4 was launched. Currently, WPS 2.0 is utilized by geospatial communities with improvements (Matthias Mueller, 2015). The OGC WPS is a standardized specification to perform spatial geoprocessing algorithms in web environment. WPS interface works on the concept of SOA (Tan et al., 2016). It facilitates publishing of

geospatial processes at the server which then is discovered and bound by the WPS clients. The core element of WPS is process which is defined as the job that takes the input and transforms it into the output. WPS processes are flexible and remotely accessible; therefore these can be easily reused for different applications. The process takes data in defined formats as input and produces outputs at the client-side.

2.1 Operations

The working of WPS on web environment includes three core operations namely *GetCapabilities*, *DescribeProcess* and *Execute* which are discussed below. These operations communicate among each other using HTTP and eXtended Markup Language (XML) encoding schema. The interface of WPS operation is shown in Figure 1. These operations are deployed on the local server of MNNIT Allahabad and accessed via calling localhost.

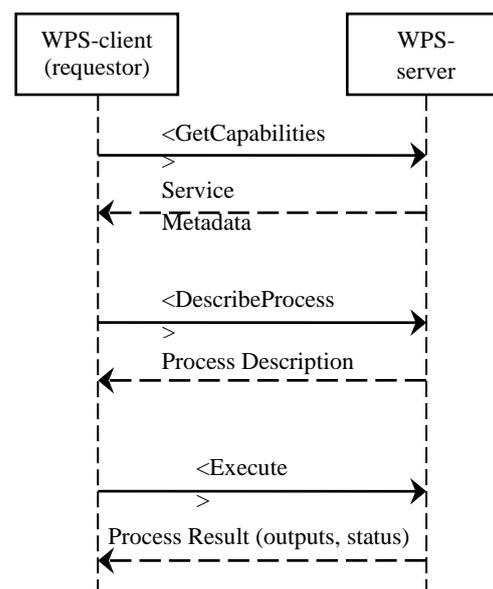


Figure 1. Communication pattern of WPS process

2.1.1 *GetCapabilities*: The *GetCapabilities* operation is a generalized operation for any OGC web services. It is an XML document which provides details about service metadata and list of processes offered by WPS server. Following Uniform Resource Locator (URL) calls this operation: <http://localhost:8080/geoserver/wps?service=WPS&request=getcapabilities>.

2.1.2 *DescribeProcess*: This operation provides an XML document that contains a detailed description of the selected process, e.g., input and output parameters. This operation is requested by following using: http://localhost:8080/geoserver/wps?service=WPS&version=1.0.0&identifier=process_name&request=describeprocess

2.1.3 *Execute*: After obtaining the detailed description of the process through *DescribeProcess* request, *Execute* operation is performed by giving all necessary inputs. Through this operation, WPS instance generates output and return it to the WPS client or stores it as a Web resource. This operation can be requested by following using URL:

http://localhost:8080/geoserver/wps?service=WPS&version=1.0.0&identifier=process_name&request=execute&datainputs=data

2.2 Implementation Tool

In this section, different OSS are reviewed for WPS implementation. Some of them are utilized for implementing WPS on web service environment, referred to as server software, while others are integrated with desktop GIS software, referred to as client software. The software products used for WPS implementation on the server side are shown in Table 1. Their detailed description is given below:

S. No.	Software/Project	Programming Languages	Platforms
1.	GeoServer WPS	Java	Windows, Linux, Mac OS, Unix
2.	52 ⁰ North	Java	Windows, Linux, Mac OS
3.	ZOO-Project	Java, Python, C, C++, PHP, JavaScript	Windows, Mac OS, Unix
4.	Deegree WPS	Java	Windows, Linux, Mac OS
5.	PyWPS	Python	Platform Independent

Table 1. OSS for WPS implementation on Server

2.2.1 Geoserver WPS: GeoServer is a Java based OSS which is used to publish, view and edit different types of geospatial data (GeoServer, 2019). GeoServer is the most flexible tool to implement the WPS service. WPS is not an in-built component of the GeoServer like other OGC services such as WMS, WFS and WCS. WPS is an extension which is combined with GeoServer as a service.

2.2.2 52⁰ North : It is a Java based OSS which was launched in 2004 (52⁰North, 2019). It enables clients to perform geoprocessing on the web (Baranski, 2008). It supports various geospatial data formats such as shapefiles, geotiff, GML, KML and WKT. Open source clients like uDig and OpenLayers can be directly integrated with plugins of 520 North WPS so that thick client users can achieve geoprocessing functionality.

2.2.3 ZOO Project : ZOO is a developer friendly open source WPS implementation framework released in 2009 under the norms of the MIT/X-11 license (ZOO_Project, 2019). ZOO kernel, ZOO services and ZOO Application Programming Interface (API) are three main components of the ZOO project (Rautenbach et al., 2012). ZOO kernel is a core component of the ZOO project that can create, manage and chain the web services. This server side C kernel supports multi-programming languages and can be easily integrated with spatial libraries and databases. The ZOO API is a server-side JavaScript API through which calling and chaining of ZOO services have become easier (Fenoy et al., 2013).

2.2.4 Deegree Project : The Deegree is a Java based open source framework that supports the SDIs to implement the major geospatial standards (Deegree, 2018). It was founded in 2002 under the GNU Lesser General Public License (LGPL). Currently, Degree implements WPS standards 1.0.0 and provides flexible environment in terms of its configuration and output formats.

2.2.5 PyWPS Project: PyWPS (Python Web Processing Service) is used for server-side OGC WPS implementation framework (PyWPS, 2018). It was started in 2006 by Jachym Cepicky who started a project that aimed to connect the GRASS GIS based processing over the web interface (PyWPS, 2013). The PyWPS Project is fully written in python language. The basic working PyWPS framework is initiated with receiving of incoming request which is compliant with OGC standards, transmitting it to the GRASS or any other python developed tools and return the result to the client (Cepicky and Becchi, 2007).

3. DESIGN AND ARCHITECTURE

The architecture of the proposed WPS framework is shown in Figure 2. The system works on the concept of multi-tier client-server architecture. It has three layers which are discussed below:

3.1 Client Layer

Web based geoprocessing is initiated from this layer. It includes thin and mobile users who can easily communicate with the web server environment. The communication between the client layer and application layer is done using the HTTP request and response operation. User make a request for geoprocessing from this layer and get response accordingly. The received response can be in XML, GML, raster or vector format. Geospatial functionalities like discovery and visualization of geospatial data can also be performed from this layer.

3.2 Application Layer

This layer includes the development of web based geospatial application which contains business logic. Two different servers, i.e., web server and GIS server are used in this layer. Apache Tomcat is used as the web server and GeoServer is used as the GIS server. Web server handles the HTTP communication and acts as the container for the GIS server. GIS server handles the geospatial data and offers OGC services.

3.3 Data Layer

This layer contains the data based on the developed application. It mainly includes vector and raster data. These geospatial data are created by using QGIS which is an OSS. PostgreSQL/PostGIS database is used to store the vector data. It is an open source database which is used to store the spatial and non-spatial data.

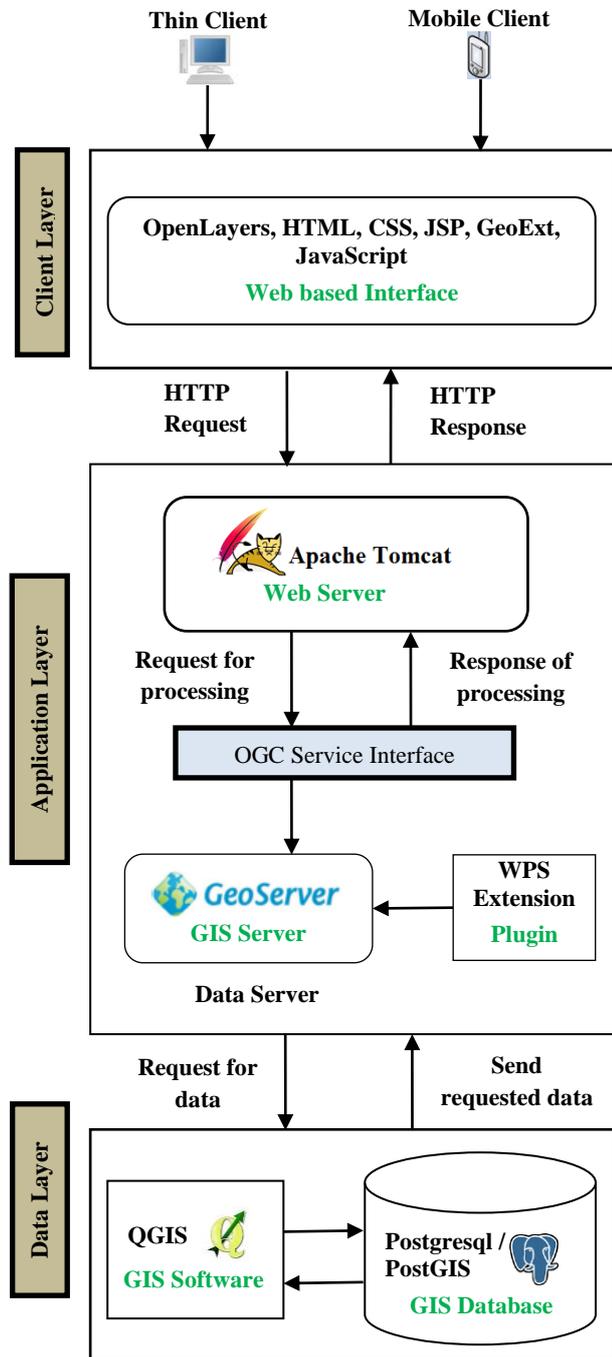


Figure 2. Proposed WPS framework

4. PROTOTYPE DEVELOPMENT

OSS based prototype for the public health SDI is developed which have the capabilities to provide the geoprocessing in the web environment and is based on the framework discussed in the previous section. Figure 3 shows the developed WPS enabled prototype of public health SDI for Prayagraj city. Following phases are involved in the development of public health SDI prototype.

4.1 Data Collection and Geospatial Database Creation

This phase includes the collection of datasets and storing it into a geospatial database. Spatial data collected for the study are listed below:

- i. Prayagraj district and tehsil boundaries are extracted from the Survey of India (SOI) topographical maps using QGIS software.
- ii. Road network of Prayagraj city is digitized from the SOI topographical maps.
- iii. Ward boundary of Prayagraj city is extracted after performing georeferencing of the ward boundary map, and digitization on the ward boundary map.
- iv. Land use and land cover (LU/LC) map is generated through supervised classification of Multispectral Instrument (MSI) sensor image (10m spatial resolution) of Sentinel-2A satellite.
- v. The hospital locations were not available. Hand held Global Positioning System (GPS) based ground surevy was carried out to get the location of 97 government and private hospitals of Prayagraj city. For this purpose, handheld GPS receiver with 2-5 meter accuracy was used. The survey data is later converted into point shapefile.

Above data except LU/LC map are in vector shapefile format. These vectors are uploaded into PostgreSQL/ PostGIS database through pgAdmin interface.

4.2 OGC Service Implementation

This phase includes two main tasks which are: (a) the publication of geospatial data at the server, and (b) analysis of datasets through the processes invoked by the client. To do the first task, GeoServer is used that published the public health data. For this, a vector datastore of PostGIS is created in GeoServer through which stored health data are published over GIS server. Users can navigate the health data via WMS and can access it via WFS. The uDig software is used to create the SLD files which are the XML files that give the rules through which data is rendered on the client screen.

The WPS task is done by adding GeoServer WPS plugin into the GeoServer. This task represents the computational backend for geospatial data. It includes three key components which are data input/ output, geoprocessing and visualization. The deployment of GeoServer enabled processes over the web interface is done by using standardized WPS operations. List of processes offered by GeoServer WPS extension is extracted by using the GetCapabilities operation while the detail of each process is obtained by the DescribeProcess operation.

4.3 GUI Development

This phase includes the development of web based public health SDI application which is created using HTML, CSS, JavaScript, JSP and OpenLayers. OpenLayers is a JavaScript library that helps to create dynamic web maps.

Through web interface of developed WPS enabled prototype of public health SDI, user can view, discover, access and process the spatial data. The selection of WPS process, corresponding inputs and output is shown in the left panel of the interface. The layer uploaded on the server and processed WPS data is

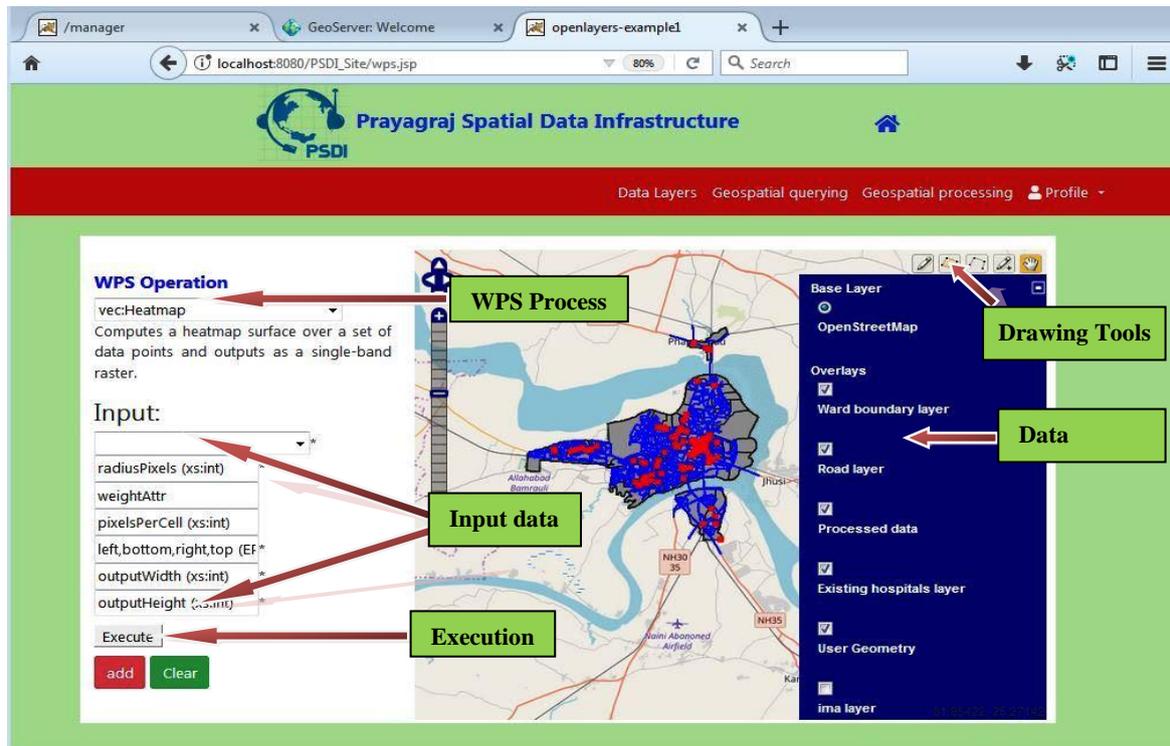


Figure 3. Developed WPS enabled prototype of public health SDI for Prayagraj city

displayed in the right panel of the interface. OpenStreetMap is being used as a base map by default. All three layers ward, i.e., boundary layer, road layer and existing hospitals layer are shown in this interface. Processed data layer displays the processed result. User Geometry layer displays the user generated geometries that can be drawn using drawing tool.

5. FUNCTIONALITIES OF WPS ENABLED PUBLIC HEALTH SDI PROTOTYPE

Geoprocessing functionalities provided by the developed prototype are discussed in this section. The WPS enabled public health SDI prototype provides different vector and raster geoprocessing operations which are described below:

5.1 Vector Operations

The developed prototype offers several vector functions through WPS that can be performed on the vector data like proximity analysis. Suppose, a user needs to identify 1 km area around the existing hospitals. For this, the user will call the buffer operation on the WPS server. This operation will create a buffer around the feature of the given distance. This distance is either given manually by the user or taken from some specific attribute of the feature. The buffer operation performed on the public health SDI prototype using WPS geoprocessing capabilities in the web environment is shown in Figure 4(a). In this, necessary inputs are given in the left panel of the interface and the corresponding result is shown in Figure 4(b) where the red points are the hospital locations and the orange circles are the buffers or region of interests. User can get the final result in the form of vector formats like shapefile, XML or GML. User can visualize the outcome on the map or download it.

Several other vector WPS operations are featured on the public health SDI prototype. These operations are listed below:

- i. Bounds: It computes the bounding box of the feature. It is the smallest rectangle that completely encloses the given feature.
- ii. Centroid: It computes the centroid coordinate of feature, i.e. ($X_{centroid}$, $Y_{centroid}$)
- iii. Clip: It clips the input feature according to the polygon drawn by the user.
- iv. Dissolve: It is provided by CollectGeometries operation that combines or aggregates the adjacent features.
- v. Count: It computes the number of features in an input feature.
- vi. Feature: It converts the user drawn geometry into a feature, i.e., vector data.
- vii. Heatmap: It is a visualization tool that takes the vector data and converts it into the interpolated raster surface.
- viii. Reprojection: This operation reprojects the feature into another coordinate reference system as specified by the user.
- ix. Nearest: It identifies the feature from feature collection that is at the smallest distance from the user drawn geometry.

5.2 Raster Operations

The developed prototype also offers several raster functions through WPS that can be performed on the raster data like cropping or clipping. Suppose, a user needs to extract some specific part of the raster layer. For this, the user will call the crop operation on the WPS server and select the raster layer that needs to be cropped. The layer will be cropped as per the user drawn polygon. The crop operation performed on the public health SDI prototype is shown in Figure 5(a) where the necessary inputs are given in the left panel of the interface. In this interface, orange polygon is showing the user defined area

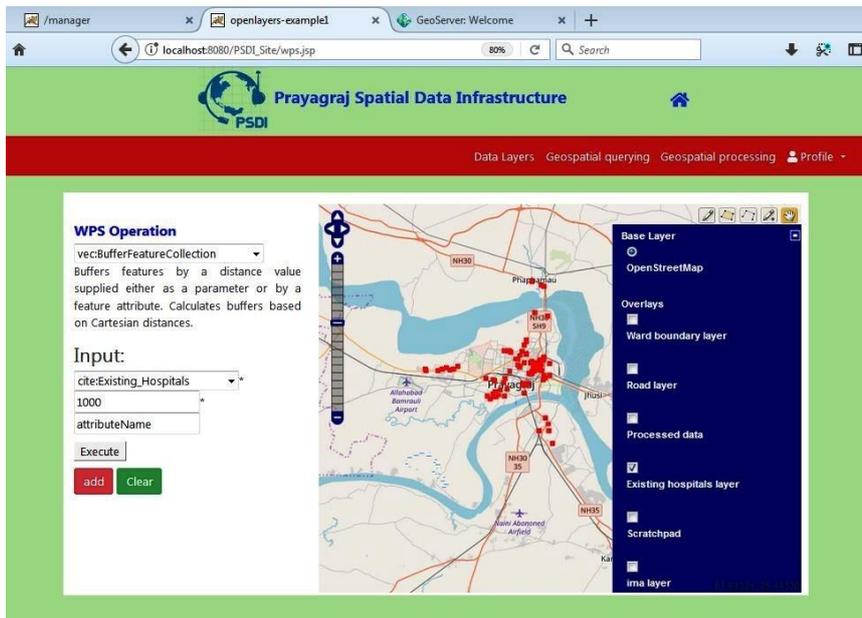


Figure 4(a). WPS based buffer operation performed on public health SDI



Figure 4(b). Result of buffer operation

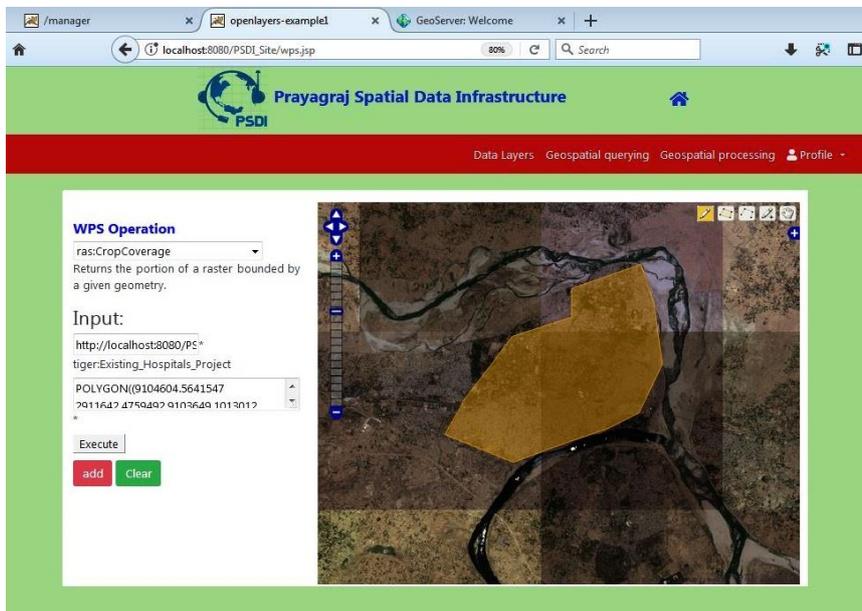


Figure 5(a). Clip operation on public health SDI using WPS

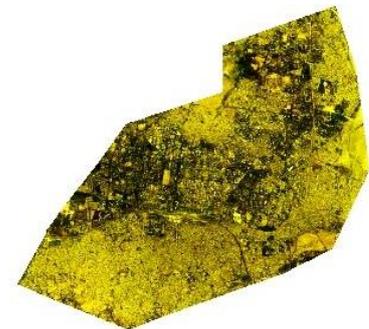


Figure 5(b). Result of clip operation

that need to be extracted by the user. The corresponding result is shown in Figure 5(b), which represent the clipped raster map.

The following raster WPS operations are also provided on the public health SDI portal:

- i. Overlay by addition (AddCoverages): It generates combined raster map by addition of two raster maps.
- ii. Overlay by multiplication (MultiplyCoverages): It generates combined raster map after multiplying the two raster maps.
- iii. Normalization (NormalizedCoverages): It generates a normalized map of the input map.
- iv. Reclassification (RangeLookup): It reclassifies a raster map into user defined set of ranges.
- v. Point Vectorization (RasterAsPointCollection): It extracts the point features of the raster pixels.

- vi. Polygon Vectorization (Polygon Extraction): It extracts vector polygons from a raster based on user defined ranges.

6. CONCLUSIONS

SDI is developed to process a large amount of spatial data. In the present research, the concept of multi-tier client-server architecture is used for design and development of WPS enabled SDI framework to provide the capabilities of geospatial data processing and analysis on the web. The primary importance of WPS interface is that the user can globally access and process the data. WPS functionality is added in the SDI using OSS like Apache Tomcat, GeoServer and PostgreSQL/PostGIS. This research demonstrates the integration of WPS into SDI through OGC compliant services. Several OGC

services are also deployed in this SDI framework like WMS for visualization, WFS for vector data editing, WCS for raster data, and CSW for data discovery.

A prototype for public health SDI is also created based on the developed WPS enabled SDI framework. For this purpose, a geospatial database is created which consist of hospital locations and their details, road network, among others. The developed health SDI prototype will provide access to health related data to the public through web based user-interactive GUI along with providing the geoprocessing capabilities in the web environment. The client can submit the data processing request over the server and can visualize the result on the map. Further, one vector and one raster WPS enabled process is demonstrated in detail along with the short description of others. The present research will help planners and decision makers in smart health management.

Acknowledgments: The author, Ashutosh Kumar Tripathi, would like to acknowledge the Technical Education Quality Improvement Programme (TEQIP), phase-III for providing him the fellowship for carrying out this research work. The authors are also thankful to Copernicus programme of European Space Agency (ESA) for providing Sentinel-2A data free.

REFERENCES

52°North, 2019. 52° North Web Geoprocessing Service Features [WWW Document]. URL <https://wiki.52north.org> (accessed 7.12.19).

Agrawal, S., Gupta, R.D., 2017. Web GIS and its architecture: a review. *Arabian Journal of Geosciences* 10. <https://doi.org/10.1007/s12517-017-3296-2>

Baranski, B., 2008. Grid computing enabled web processing service, in: *GI-Days*. Münster, Germany.

Baumann, P., 2012. OpenGIS Web coverage Service (WCS) implementation specification. Wayland, MA. [https://doi.org/Open Geospatial Consortium Document No. 06-083r8](https://doi.org/Open%20Geospatial%20Consortium%20Document%20No.%2006-083r8)

Béjar, R., Latre, M.Á., Lopez-Pellicer, F.J., Noguera-Iso, J., Zarazaga-Soria, F.J., Muro-Medrano, P.R., 2012. SDI-based business processes: A territorial analysis web information system in Spain. *Computers & Geosciences* 46, 66–72. <https://doi.org/https://doi.org/10.1016/j.cageo.2012.04.010>

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. *Environmental Modelling and Software* 41, 72–83. <https://doi.org/10.1016/j.envsoft.2012.11.010>

Cepicky, J., Becchi, L., 2007. Geospatial processing via Internet on remote servers-PyWPS. *OSGeo Journal* 1, 39–42.

Coleman, D.J., McLaughlin, J., 1998. Defining global geospatial data infrastructure (GGDI): components, stakeholders and interfaces. *Geomatica* 52, 129–143.

de la Beaujardiere, J., 2006. OpenGIS Web Map Server (WMS) implementation specification, Open Geospatial Consortium. Wayland, MA. [https://doi.org/Open Geospatial Consortium](https://doi.org/Open%20Geospatial%20Consortium)

Document No. 06-042

Deegree, 2018. deegree Web Map Service v2 . 4 [WWW Document].

Fenoy, G., Bozon, N., Raghavan, V., 2013. ZOO-Project: The open WPS platform. *Applied Geomatics* 5, 19–24. <https://doi.org/10.1007/s12518-011-0070-0>

Foerster, T., Schaeffer, B., Brauner, J., Jirka, S., 2009. Integrating OGC web processing services into geospatial Mass-market applications, in: *2009 International Conference on Advanced Geographic Information Systems & Web Services*. IEEE, Cancun, Mexico, pp. 98–103. <https://doi.org/10.1109/GEOWS.2009.19>

GeoServer, 2019. GeoServer Documentation [WWW Document]. URL <http://geoserver.org/> (accessed 8.17.19).

Giuliani, G., Lacroix, Á.P., Guigoz, Á.Y., Roncella, Á.R., Bigagli, L., Santoro, M., Mazzetti, P., Nativi, S., Á, N.R., Á, A.L., 2017. Bringing GEOSS Services into Practice: A Capacity Building Resource on Spatial Data Infrastructures (SDI). *Transactions in GIS* 21, 811–824. <https://doi.org/10.1111/tgis.12209>

Herle, S., Blankenbach, J., 2018. Enhancing the OGC WPS interface with GeoPipes support for real-time geoprocessing. *International Journal of Digital Earth* 11, 48–63. <https://doi.org/10.1080/17538947.2017.1319976>

Li, D., Chen, Z., Deng, Z., Huang, W., Tang, J., Di, F., Gao, X., Xie, Q., Liu, L., Jiang, X., 2018. A Wide Area Service Oriented Architecture Design for Plug and Play of Power Grid Equipment, in: *Procedia Computer Science*. Elsevier B.V., Qufu,China, pp. 353–357. <https://doi.org/10.1016/j.procs.2018.03.088>

Matthias Mueller, 2015. OGC WPS 2.0 Interface Standard. Wayland, MA.

Nebert, D., Whiteside, A., Vretanos, P., 2007. OpenGIS Catalogue Services (CSW) Specification. Wayland, MA. [https://doi.org/Open Geospatial Consortium Document No. 07-006r1](https://doi.org/Open%20Geospatial%20Consortium%20Document%20No.%2007-006r1)

PyWPS, 2018. PyWPS Project Documentation [WWW Document]. URL <https://buildmedia.readthedocs.org/media/pdf/pywps/master/pywps.pdf> (accessed 8.14.19).

Rajabifard, A., Feeney, M.-E., Williamson, I.P., 2002. Directions for the future of SDI Development. *International Journal of Applied Earth Observation and Geoinformation* 4, 11–22.

Rautenbach, V., Coetzee, S., Strzelecki, M., Iwaniak, A., 2012. Results of an Evaluation of the Orchestration Capabilities of the ZOO Project and the 52° North Framework for an Intelligent Geoportal, in: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. pp. 163–168. <https://doi.org/10.5194/isprsannals-I-4-163-2012>

Schut, P., 2007. Open GIS® Web Processing Service (Version 1.0.0) (No. Open Geospatial Consortium Document No. 05-007r7). Wayland, MA.

Singh, H., Karnatak, H.C., Garg, R.D., 2019. An automated and optimized approach for online spatial biodiversity model: a case study of OGC web processing service. *Geocarto International* 34, 194–214. <https://doi.org/10.1080/10106049.2017.1381178>

Tan, X., Di, L., Deng, M., Huang, F., Ye, X., Sha, Z., 2016. Agent-as-a-service-based geospatial service aggregation in the cloud: A case study of flood response. *Environmental Modelling and Software* 84, 210–225. <https://doi.org/10.1016/j.envsoft.2016.07.001>

Tripathi, A.K., Agrawal, S., Gupta, R.D., 2020. A Conceptual Framework of Public Health SDI, in: Ghosh, J.K., Silva, I. da (Eds.), *Applications of Geomatics in Civil Engineering. Lecture Notes in Civil Engineering*. Springer Singapore, Dehradun, India, pp. 479–487. <https://doi.org/10.1007/978-981-13-7067-0>

Tripathi, A.K., Agrawal, S., Gupta, R.D., 2018. A comparative analysis of conventional hadoop with proposed cloud enabled hadoop framework for spatial big data processing, in: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. ISPRS, Dehradun, India, pp. 425–430. <https://doi.org/10.5194/isprs-annals-IV-5-425-2018>

Vancauwenberghe, G., van Loenen, B., 2018. Exploring the Emergence of Open Spatial Data Infrastructures: Analysis of Recent Developments and Trends in Europe, in: Saqi, S., T., R., Zaigham, M. (Eds.), *User Centric E-Government. Integrated Series in Information Systems*. Springer, Cham, pp. 23–45. https://doi.org/10.1007/978-3-319-59442-2_2

Vretanos, P.A., 2005. *OpenGIS Web Feature Service (WFS) implementation specification*. Wayland, MA. https://doi.org/10.1007/978-3-319-59442-2_2

Yue, P., Guo, X., Zhang, M., Jiang, L., Zhai, X., 2016. Linked Data and SDI: The case on Web geoprocessing workflows. *ISPRS Journal of Photogrammetry and Remote Sensing* 114, 245–257. <https://doi.org/10.1016/j.isprsjprs.2015.11.009>

ZOO_Project, 2019. *ZOO-Project Documentation [WWW Document]*. URL <http://zoo-project.org/docs/ZOO-Project.pdf> (accessed 8.12.19).