

INTERNATIONAL JOURNAL OF APPLIED GEOSPATIAL RESEARCH

April-June 2012, Vol. 3, No. 2

Table of Contents

EDITORIAL PREFACE

- i Free and Open Source Software: Development and Utilization**
Donald P. Albert, Sam Houston State University, USA

RESEARCH ARTICLES

- 1 To Emphasize Openness**
Ken Hartness, Sam Houston State University, USA
- 6 Open Source Based Deployment of Environmental Data into Geospatial Information Infrastructures**
José Gil, Institute of New Imaging Technologies, Universitat Jaume I, Spain
Laura Díaz, Institute of New Imaging Technologies, Universitat Jaume I, Spain
Carlos Granell, Institute for Environment and Sustainability, Italy
Joaquín Huerta, Institute of New Imaging Technologies, Universitat Jaume I, Spain
- 24 A Review of Geospatial Information Technology for Natural Disaster Management in Developing Countries**
Sam Herold, University of Ottawa, Canada
Michael C. Sawada, University of Ottawa, Canada

REPORTS

- 63 Embracing Geographic Analysis Beyond Geography: Harvard's Center for Geographic Analysis Enters 5th Year**
Weihe (Wendy) Guan, Harvard University, USA
Peter K. Bol, Harvard University, USA
- 72 A Reflection on the Ph.D. Program in Spatially Integrated Social Science at the University of Toledo**
Bhuiyan Monwar Alam, The University of Toledo, USA
Jeanette Eckert, The University of Toledo, USA
Peter S. Lindquist, The University of Toledo, USA

Open Source Based Deployment of Environmental Data into Geospatial Information Infrastructures

José Gil, Institute of New Imaging Technologies, Universitat Jaume I, Spain

Laura Díaz, Institute of New Imaging Technologies, Universitat Jaume I, Spain

Carlos Granell, Institute for Environment and Sustainability, Italy

Joaquín Huerta, Institute of New Imaging Technologies, Universitat Jaume I, Spain

ABSTRACT

Today, scientists use local and closed geospatial solutions to run their models and store their results. This may limit their ability to share their models, and results with other interested colleagues. This scenario is changing with the advent of new factors such as the rapid growth and rise of open source projects, or new paradigms promoted by government organizations to manage environmental data, such as Infrastructure for Spatial Information in the European Community (INSPIRE) directive, or the massive use of Web 2.0 techniques where users are looking for applications with a high degree of collaboration, interactivity, and multimedia effects. Many authors address the versatility of Spatial Data Infrastructures where resources are shared and accessed via standard service according to complex specifications. In this context, the authors point out the need to merge the traditional building and maintenance of these infrastructures, driven by official providers, with these more participative methodologies where users can participate in creating and integrating information. It seems necessary to develop new geospatial tools which integrate these new trends. This paper proposes a unified solution offering to the scientific field an open development framework, based on standards and philosophies focused on new technologies and scientific needs.

Keywords: Capacity Building, Data Integration, Data Sharing, Geospatial Services Deployment, INSPIRE, Interoperability, Open Source (OS), Rich Internet Applications (RIA), Spatial Data Infrastructure (SDI)

INTRODUCTION

One of the challenges we are facing today is to better understand the processes that occur over

or near-surface of our planet (Goodchild, 2008). The Earth is a multi-dimensional system made of complex interactions highly interconnected and continuously evolving at many spatial and temporal scales (GEO Secretariat, 2007). This means that to understand such interactions,

DOI: 10.4018/jagr.2012040102

scientists and environmental experts need to collect and integrate different data sets referred to physical and biological aspects of the Earth (Giuliani et al., 2011). In addition, these data sets are often georeferenced, that is, they describe physical phenomena tied to a concrete geographical location in terms of points, lines and sets of polygons, and we refer to them as geospatial data.

Scientists and environmental experts need to manage vast amounts of these geospatial data to turn them into meaningful information. The applicability of complex environmental processes and models has been the focus of intense research in this domain (Poch, Comas, Rodríguez-Roda, Sánchez-Marrè, & Cortés, 2004; Minsker et al., 2006; Reichert & Schuwirth, 2010). For instance, models for reduction of energy consumption, computation of atmospheric emissions, and simulation of forest fire require multidisciplinary teams of geospatial technologists and environmental experts. Such models, however, have been traditionally performed using multiple, isolated desktop geospatial tools: a specialist in Digital Elevation Models might perform slope computations while other colleague might apply remote sensing techniques over satellite imagery to produce a snow cover product. In this simple example, various GIS software packages could be involved in the activities of a single environmental model. However, this paradigm of closed solutions, which operates with local data sets, provides a way of hiding shared results and collaborative research, and more often, hindering further analysis tasks.

Better decisions and concrete actions at local and global scale to underpin sustainable development can be taken only based on precise knowledge of our environment (Lannotta, 2007). However, geospatial datasets, and other resources are often isolated, hidden to potential interesting stakeholders that could benefit from them. Therefore, the ease of access and retrieval of geospatial and environmental data is a key aspect for information infrastructures (Yang, Raskin, Goodchild, & Gahegan, 2010). There

are consensus-based initiatives which aim is to promote interoperability, standards specifications, and seamless integration of services and information to improve this accessibility to geospatial resources (see next sections). However, the participation of users in these infrastructures has been delayed due to the complexity of current methodologies to participate and integrate new resources; the necessity of skills about how to incorporate real time data with historical, the use of standard services specification to deploy resources and promote interoperable resources, the integration of data from different domains, or simply looking for these data sources have become tedious tasks in current developments (Yang & Raskin, 2009). This makes scientists continue using traditional methods like local processing.

The aim of this article is to present an open source-based framework that enables both: standard access and processing of data, and also the deployment of new content into geospatial information infrastructures. The processing and access of environmental data sets and the deployment and sharing of processing results will be addressed in this paper. All this, in the simplest way possible, through an immersive and user-friendly interface, based entirely on geospatial open source technologies.

The rest of this article is structured as follows. The following three sections outline the basic concepts underlying the topics tackled in this article: architectural styles for information infrastructures, concrete realizations of geospatial information infrastructures, and the concept of open source applied to geospatial technologies. Given the pieces of the puzzle, we present a toolbox that tries to connect all pieces to facilitate the sharing, deployment and publication of new geospatial and environmental data sets into existing geospatial information infrastructures. We discuss the applicability of our framework through a use case in the forest fire domain. The paper concludes by looking to futures challenges in automatic deployment of user-generated content and identifies a number of issues that are in need of further research.

DESIGNING INFORMATION INFRASTRUCTURES

Geospatial and environmental research is a field that involves multidisciplinary teams of scientists and technologists working together. The application of environmental models may span from local to global scope, what undeniably requires suitable information infrastructures to share and do research on the web (Hey & Trefethen, 2005). This complex scenario demands the use of new architectural styles which oppose centralized, isolated solutions, and instead support distributed processing capabilities and remote communications, necessary ingredients to successful collaborative and multidisciplinary research. Service-Oriented Architectures (SOA) and Resource-Oriented Architectures (ROA) are currently the architectural styles adopted in the development of collaborative, distributed Web systems and applications.

SOA is an architectural style to design applications based on a collection of best practices, principles, interfaces, and patterns related to the central concept of service (Bell, 2008). In SOA, services play a key role and become the basic computing unit to support development and composition of larger, more complex services, which in turn can be used to create flexible, *ad hoc* and dynamic applications. The main design principle behind SOA is that a service is a standards-based, loosely-coupled unit composed of a service interface and a service implementation. The service interface describes the functional capabilities of a service. The service implementation codifies what a service should execute. This principle provides a clean separation of concerns especially between service interfaces (what services offer to the public community) and internal implementations (how services work). Essentially SOA introduces a new philosophy for building distributed applications, where services can be discovered, aggregated, published, reused, and invoked at the interface level, independently of the specific technology used internally to implement each service.

At the time of implementation SOA-based services must make use of concrete languages and protocols. Here is where web service technology gains importance because it increasingly is becoming the choice to implement SOA-based applications. Web services (Papazoglou, 2008) are, by definition, loosely coupled independent units and are well described (interface description contains functional properties), thereby promoting one of the goals of SOA: enabling interoperability or the ability of services to interact with minimal knowledge of the underlying structure of other services (Goodchild, Egenhofer, Fegeas, & Kottman, 1999). Interoperability is achieved (or optimized) by using standard interfaces. Web service technology includes various standards such as Web Service Description Language (WSDL) for the description of service interfaces, Universal Description, Discovery and Integration registry (UDDI) for their advertisement and discovery, and Simple Object Application Protocol (SOAP) that enables communication among services (Curbera, Duftler, Khalaf, Nagy, Mukhi, & Weerawarana, 2002).

As opposed to SOA, focused on distributed capabilities and services, ROA is an architectural style devoted to manage distributed, heterogeneous domain resources. In SOA client applications interact with a distributed Web applications through delegation, that is, by specifying the desired capability to a service component instead of directly acting on the resources themselves (Mazzetti, Nativi, & Caron, 2009). In ROA, through, client applications interact directly with the exposed resources.

The main constraints behind ROA-based applications are the set of architectural principles known as Representational State Transfer (REST) (Fielding, 2000), namely:

- Resources should be identified properly using global identifiers (URI). However, REST proposes to use URIs not only for identification but also for physical access to the actual resource representation. One important feature is that each resource must

be addressable via a HTTP (Hypertext Transfer Protocol) URI.

- Uniform interface through the use of HTTP as the unique application-level protocol. HTTP has a small, fixed set of operational methods with specific purpose and meaning. For instance, the GET method is for retrieving representations of target resources, the POST method for creating new resources, PUT for updating resources, and finally, the DELETE method to eliminate a given resource.
- Resources are manipulated through representations since clients and servers exchange self-descriptive messages each another.
- Interaction stateless since servers only record and manage the state of their resources they exposes. Client sessions are not maintained in the server.
- Hypermedia as the engine of application state, that is the application state is built following hyperlinks according to the navigation paradigm so that clients can discover related resources from a given one.

As in the case of SOA-based services, ROA-based applications can be implemented in many different ways, such as RESTful web services (Richardson & Ruby, 2007). Designing RESTful web services following the REST constraints described above has gained popularity within the Earth science, mobile and location based services community (Mazzetti, Nativi, & Caron, 2009).

The selection of one or another architectural style depends often on the context of the target application. Both approaches should not be disjoint but complementary. In this paper, we take the complementary strategy for the proposed framework implementation that makes use of both architectural patterns to communicate with remote geospatial services. Accessing to common geospatial services, as we describe in the following, are built upon a SOA approach, while recent open source implementations already expose RESTful interfaces to manipulate and deploy remote geospatial resources.

GEOSPATIAL INFORMATION INFRASTRUCTURES

Several administrations are recognizing the role of geospatial data in e-commerce, sustainable development and government activities, and the ability to be efficiently coordinated and managed for the interest of public in general (Rajabifard, Binns, Masser, & Williamson, 2006). In consequence, the notion of Spatial Data Infrastructures (SDI) refers to the specialization of information infrastructures for the geospatial sciences, allowing discovering, access, publishing, sharing, maintaining, and integrating geospatial data (Nebert, 2004).

Many initiatives at different administrative levels have been recently flourishing to promote the creation of SDIs (Masser, 2005). These initiatives coordinate actions and policies that encourage awareness of institutional agreements, common standards and effective mechanisms for data access, harmonization, standardization, and the development of interoperable geospatial data and technologies to support decision making for multiple purposes. One of these initiatives to eliminate jurisdictional, cultural or domain boundaries is the Infrastructure for Spatial Information in the European Community (INSPIRE) Directive of the European Commission (European Parliament and Council, 2007). INSPIRE Directive establishes a legal framework for the creation and implementation of the European Community SDI, forcing organizations to publish their geospatial resources (focusing on the environmental domain) under standard data services and by promoting the adaptation of the SOA paradigm and geospatial web service standards.

In essence, geospatial web services allow users to access, manage, and process geospatial data in a service-oriented manner (Granell, Gould, & Esbri, 2008). The demand for interoperability has boosted the development of standards and tools to facilitate data transformation and integration, mostly in terms of standard service interfaces specified by Open Geospatial Consortium (OGC¹). The Web Map Service (WMS) (OGC, 2006), the Web Feature

Service (WFS) (OGC, 2005), the Web Coverage Service (WCS) (OGC, 2008), the Sensor Observation Services (SOS) (OGC, 2007a) and the Web Processing Services (WPS) (OGC, 2007b) are some prominent examples of OGC interfaces for geospatial services with a clear applicability to environmental applications (Granell, Diaz, & Gould, 2010). For instance, a WMS could be used to provide a map of the protected areas of a given region, a SOS may serve detailed meteorological observations from near weather stations, and a WCS may offer satellite imagery of the area of interests and its surroundings. The central building-blocks for data, as well as service discovery, are provided by the Catalogue Services for the Web (CSW) (OGC, 2007c). The CSW provides one access point to users that publish and search environmental data.

In the proposed framework implementation, we will use a broad list of OGC services (WMS, WFS, WCS, WPS, and CSW) to let experts to perform not only retrieval but also deployment of geospatial data sets.

GEOSPATIAL OPEN SOURCE TECHNOLOGIES

The use of geospatial applications for environmental studies has experienced a significant boom in the last years (Minsker et al., 2006; Pezzoli, Marciano, & Robertus, 2006; Morisette et al., 2009). Its development has led to progress in making decisions in real time (Ramapriyan, 2008), it has created new relationships between different areas of knowledge (Ball et al., 2008) and has developed new processes to understand the behavior of natural phenomena (Díaz, Granell, Gould, & Olaya, 2008).

For all these efforts to continue to improve, they need to be associated with technological advances that gradually eliminate barriers that scientists found in their research. This requires advanced tools (processing, analysis and visualization) and sophisticated information infrastructures that provide access to large volumes of data. Given the wide array of open source technologies for geospatial data (Steiniger &

Bocher, 2009), these can be a serious solution to address the lack of mechanisms for democratizing the capacity building of geospatial information infrastructures.

A key reference in geospatial technologies' state of the art is the world conference for Free and Open Source Software for Geospatial (FOS-S4G²). One can find the latest innovative tools and developments in the geospatial landscape, such as OpenGeo Suite or GeoNode, which are examples of open source technology related to those proposed in this paper. OpenGeo Suite (Pumphrey, 2009) is an open source project that brings together different technologies (PostGIS, GeoServer, GeoWebCache, GeoExt, OpenLayers) to develop an architecture based on OGC standards. The main objective is that users can easily build a platform that provides maps and data via web applications, mobile and desktop clients. GeoNode provides a new SDI solution by adding a new application discovery layer so that scientists can easily discover the data services. The result is a platform that facilitates the creation, sharing and collaborative use of geospatial data (Benthall & Gill, 2010).

Merging our work with other domains not strictly in touch with geospatial domain, we must review the take off of many web applications that following the Web 2.0 philosophy have created an environment that brings users to participate very actively and add new content to be accessed and shared by many users. This way of content sharing brings new forms of study. Not only with the incorporation of the social factor but with a new relationship with the user. Applications are required to have higher degree of interaction and attractive interfaces that facilitate complex tasks. Solutions such as RIA (Rich Internet Applications) can be a key element in future developments. In this paper we describe how we propose to merge these distinct disciplines. Having an underlying geospatial information infrastructure based on standard services and components we add on top a framework composed by mechanisms and Web 2.0 interfaces to assist users in not only accessing but also deploying and sharing new geospatial resources in this SDI.

FRAMEWORK FOR DEPLOYMENT OF ENVIRONMENTAL DATA

In the previous section we have seen numerous technological trends that can potentially become the basis of a set of solutions in multiple scenarios. On one hand, common policies and agreements in terms of the notion of SDI, understood as geospatial data stores collected by government agencies and private institutions, where data are published via open, standards-based services, can be easily incorporated into scientific applications (Nerbert, 2004). The INSPIRE directive, which creates a collaborative SDI, accessible, designed to facilitate data sharing and standardization of specifications (European Parliament and Council, 2007), reinforced by standardization organizations, such as the Open Geospatial Consortium (OGC) and International Organization for Standardization (ISO). On the other hand, the advent of recent Web 2.0 technologies to create interoperable geospatial web portals (Yang, Li, Xiao, Raskin, & Rambacus, 2007; Wilkins-Diehr, Gannon, Klimeck, Oster, & Pamidighantam, 2008) and that allows users to publish and access data in an easy way. The rise of open-source service-oriented geospatial environments and tools, provide a variety of solutions and knowledge that can be easily extrapolated or incorporated directly into new scientific applications (Pierce, Fox, Choi, Guo, Gao, & Ma, 2009).

Why then the diverse geospatial technologies, open source tools, and geospatial information infrastructures have not received the same level interests in environmental applications as in others domains? After all, SDI and open source geospatial technologies presents many benefits for environmental scientists and experts, since the intrinsic multidisciplinary aspect in environmental research can be viewed as a source of expertise for collaborative applications between different domains (Westlund, 2010).

From the user's perspective of environmental and geospatial sciences, the increasing complexity of current methodologies, tools and

technologies may lead to underuse such information infrastructures for efficient analysis and decision-making tasks. This means scientists who produce interesting results through models and processes often use information for access, not to publish their own data sets. This implies in low rates of user participation in the process of injecting environmental datasets into information infrastructures for collaborative research.

For this reason, we present an open source framework based on INSPIRE architecture that has been extended with some modules for not only accessing but also deployment of environmental data into geospatial information infrastructures (like SDI). The main characteristics are the following:

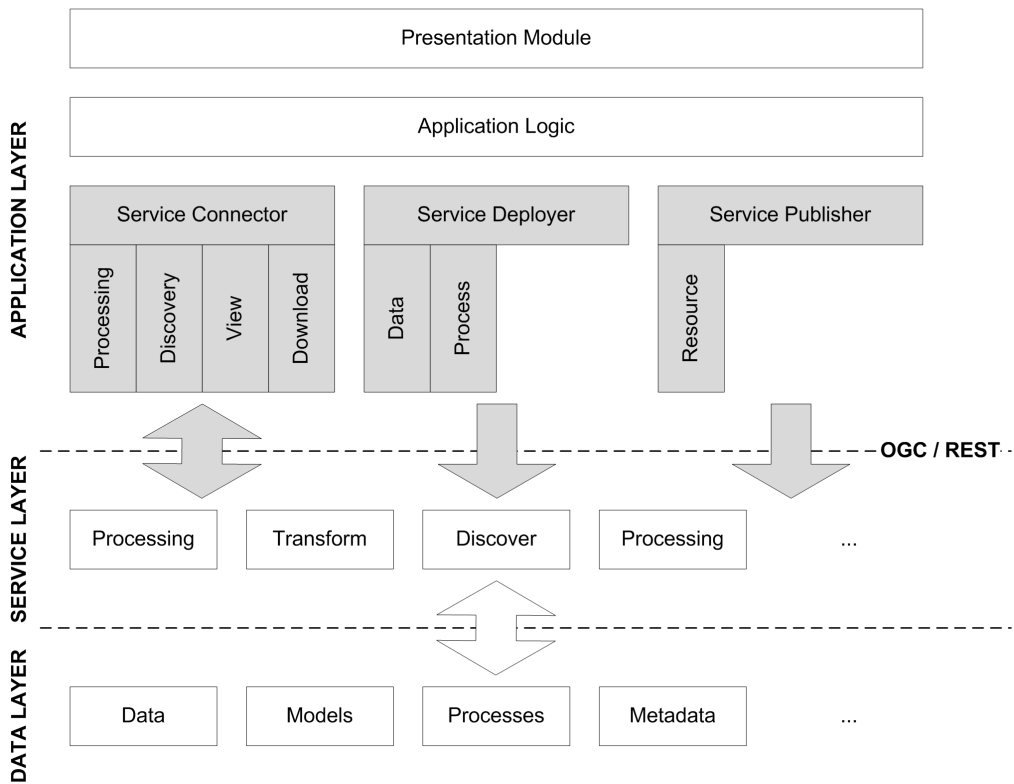
- It allows the access and retrieval of environmental data by means of interoperable OGC-based services),
- It allows the integration and processing of new information by means of deployment mechanisms,
- It allows the generation and publication of metadata descriptions in OGC-based catalogs for further search.

Figure 1 illustrates the conceptual architecture of the proposed framework. It basically follows the INSPIRE technical architecture, that is, three-layer SOA: the Application layer (top) contains client side services, application logic and presentation modules; the Service layer (middle) with server side services that use a standard interface for communication; and the Data layer (bottom). Grey boxes represents the components contained within the Service Framework. In the following we outline the main functionality of each layer. Next section will describe in detail each component in grey in Figure 1

Application Layer

Our contribution on this layer is to add the Service Deployer and Publisher that allows user not only access resources but also deploy new resources in the SDI and a user interface

Figure 1. Multi-layered architecture for the proposed framework



that following web 2.0 principles offers a entry point to the underlying SDI that allows users to participate in the building and maintenance of the SDI.

Following the Model-View-Controller (MVC) pattern (Freeman, Sierra, & Bates, 2004) to build the client side, we find the Application layer is divided into three levels according to their proximity to the user. At the level closest to the user, we have the visual part of the application, multimedia effects and interaction with the user. At the intermediate level, we find the application logic associated with the controller of the actions produced by the user. At the lower level we have client side's services that allow communication with the geospatial services deployed at the Service layer, through OGC standards and RESTful technology.

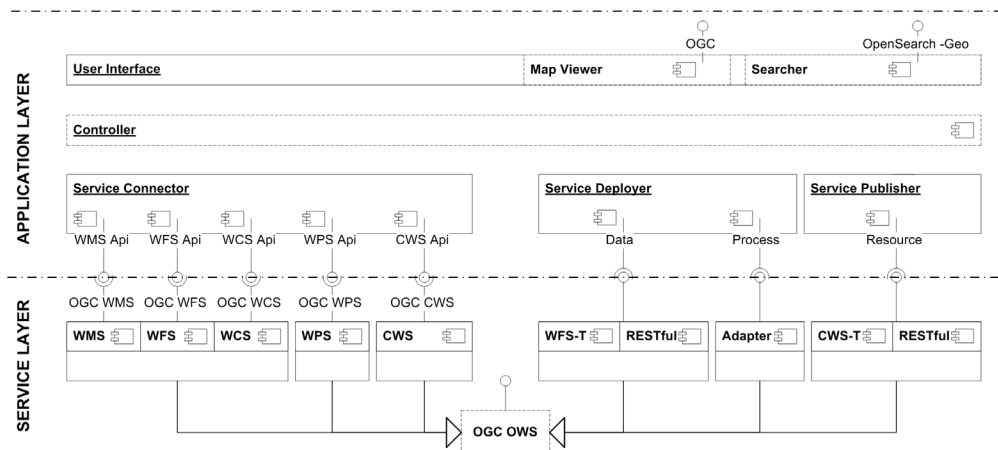
Service Layer

The Service layer is responsible for mediating between the Application and the Data layer. Its development focuses on the paradigm INSPIRE/SDI/SOA to minimize problems of accessibility and interoperability, so that wherever possible requests for data on this layer are performed using OGC standards. However, on certain occasions such as the insertion of new data, we need to use a RESTful approach. In this layer reside the instances of the OGC services in which we deploy the new content that users will provide using the framework interface. Service Framework is preconfigured to work with existing instances of geospatial web services.

Data Layer

In this part we find the final elements for data storage, such as databases or simply memory

Figure 2. Interfaces used to expose the components functionality



disk regions, where the Service layer can store their data. It is out of the scope of this paper to describe further this layer since it will be something managed by the OGC standard service implementations that we use to deploy new data content and that reside in the service layer.

OPEN SOURCE BASED COMPONENTS

As we have seen throughout the paper, there are many open source projects in the geospatial domain and in particular many projects that implement open source components according to OGC standard specifications. We describe next the service framework components regarding the technology used to implement them. For the Service Framework implementation we have reuse some of these existing open source components and also we have self-develop some components to be offered through an open source license.

The rest of this section describes our solution from two points of view: interfaces and technologies used. Figure 2 shows the interfaces implemented by each component in the Service Framework. Figure 3 shows the technology that has been used to implement each of these components. Some of these technologies are existing open source projects and self imple-

mented components to be delivered as open source projects as well.

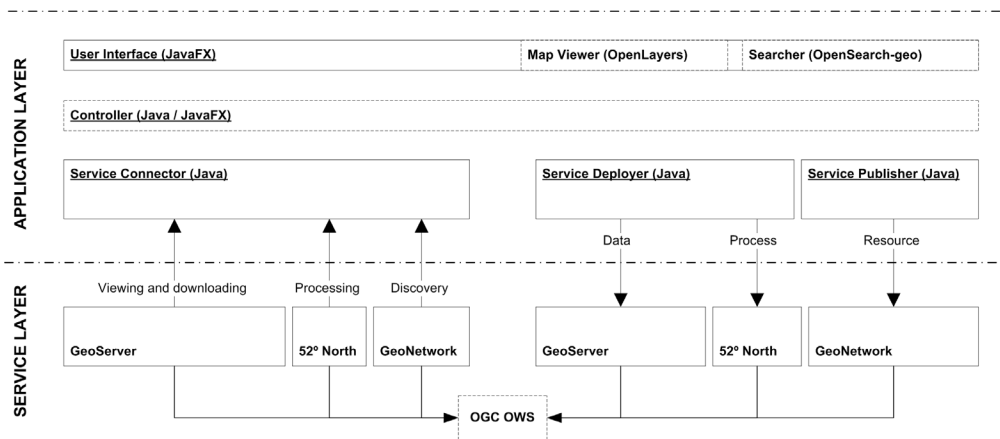
Application Layer

The Application layer as we can see in both Figures 2 and 3 is composed of several sub-layers and components. At the top of the Application Layer, we find the components that interact directly with the user. Next we describe each of these components regarding the interfaces and the technology used.

User Interface, Map Viewer and Searcher

Interface. The main purpose of the User Interface is to facilitate the user interaction with the application. The easiness and comfort of use and the level of interaction are some factors that determine the success or failure. The User Interface integrates the other two components that interact directly with the user: the Map Viewer and the Searcher. The former component offers a common map-based interface for data visualization and exploration. The latter, the Searcher component, follows an OpenSearch³ interface. This simple specification proposes a standard search interface, used in most modern browsers to manage custom searches. In addition, a geospatial extension (OpenSearch-Geo)

Figure 3. Technology used to implement the components



has been proposed that adds geographic filtering capabilities and support for geographic response formats, obtaining then an optimal approach to search for geospatial data.

Technology. As we shown in Figure 3, the interface is implemented using RIA technology. It proposes a new evolution of the client server paradigm that allows executing a portion of the application on the user's local system. This reduces the communication with the server and increases the user's autonomy (Weaver, Gao, Chin, & Iverson, 2009).

From among the wide range of valid options like FLEX, SilverLight or JavaFX, we have selected JavaFX because of its fully integration with Java which allows us to easily integrate the components reused and developed in our framework. JavaFX is structured according to the classical concept of scene graph for the implementation of graphical applications, with a primary scene node on which iteratively hang the rest of nodes. These nodes represent mainly behavior animation, transition effects, visual content and user interaction (Weaver, Gao, Chin, & Iverson, 2009).

In the open source community there exist several map viewers components available. In our work OpenLayers has been the chosen one. OpenLayers is a JavaScript client that offers interesting features like very good performance, the ability to consume many different

geographic formats, and client side support of standards such as OGC Web Map Service (WMS) and Web Feature Service (WFS). The communication between the interface and the map viewer component is done through calls to specific methods (using the JavaFX Applet-StageExtension class) that are interpreted by the JavaScript runtime engine in the browser.

Furthermore, OpenLayers has been the container extended with a new self developed control called OpenSearch-geo control which has been integrated in OpenLayers for searching purposes (Fonts, Huerta, Díaz, & Granell, 2010). This new control in the client translates user queries to the standard OpenSearch protocol, using geospatial parameters and supported formats, and offers user the retrieved documents. As in the previous case, the JavaFX Applet-StageExtension class is the preferred interface.

Controller

Interface. Following the MVC pattern, the logic of the application is located in the Controller component. This responds to events produced by the user, and invokes requests to the model by delegation to the lower components of the Application Layer (Service Connector, Service Deployer and Service Publisher).

Technology. This component is developed using the Singleton pattern (Freeman, Sierra, &

Bates, 2004) so that any instance requested on it to access the same content and execution. The interactions between the view (user interface) and the controller are performed via a new feature in JavaFX that synchronizes variables between the user interface and the controller. Examples are the list of local resources loaded (localFileResources) as a view update and the method loadLocalFile() as an input to the controller (and therefore execution of actions on the model).

Service Connector

Interface. The Service Connector component in the application layer contains components that implement the client side interfaces of the standard services deployed in the Service layer. For instance it contains the component Web Processing Service Application Programming Interface (WPSAPI) that implements the OGC WPS client interface to be able to access and run distributed processing services. Another module developed in the Service connector is the CSWAPI (Catalog Service for Web API).

Technology. Currently, Service Connector modules like WPSAPI and CSWAPI are included in a self-developed Java library. For instance WPSAPI offers a programmatic way to send and receive *getCapabilities* request using Java objects. It implements objects to send and receive Process description requests to get the description of processes in order to see the parameters and the way a process within a WPS has to be invoked. Finally, it permits sending an execute request to invoke one of the available processes and it is able to retrieve and parse the response in order to extract the process results. The object model is scalable to support new versions. Currently versions 0.4 and 1.0.0 are supported. This component is programmed also under open source license.

Service Deployer

Until now the described components offered accessing capabilities to resources deployed

in the infrastructure. The aim of the Service Deployer component is to assist the user to deploy resources as services in the infrastructure. These resources can be either data resources or processes. In this paper we focus on the deployment of data resources.

Interface. The Data Wrapper component allows users to upload geospatial data to be deployed and delivered as standard OGC data services, such as WMS, WFS and WCS. The Data Wrapper interface contains two methods: *WrapFile* and *WrapData* to deploy data available in a File or included as a parameter in the request.

To deploy data as an OGC Service, this component should connect to OGC Service instances available in the infrastructure. Some of these OGC standards (such as WFS) include transactional operations in their interface specification. In this case, the resources can be deployed by implementing the transactional interface in the client side. This is the ideal case, because the component implementation is independent of service implementations. In other cases like deploying data for visualization purposes as WMS, the specification does not consider transactional operations; therefore this component has to deal with the concrete services implementation and their own protocol and interface. Currently, in our prototype implementation this is the case of the Data Wrapper component when deploying data in WMS implementations using the Geoserver RESTful API⁴.

Technology. The Data Wrapper component is a self developed Java library in form of API exposing the described interface. In order to deploy the data, we have first associated this component with an instance of OGC Data Service that implements WMS, WFS and WCS interfaces. This OGC Services implementation is an open source project called Geoserver, which offers a RESTful API to deploy content. Geoserver is then the responsible for publishing data through OGC interfaces, transforming the resources processed in WMS for viewing and WFS and WCS for download purposes.

Service Publisher

Service Publisher component is in charge of deploying data and process descriptions to an open catalog, so they can be found in the infrastructure.

Interface. Like the Data Wrapper, the Service Publisher needs to implement an interface to deploy content in a specific Service type. In this case the content is not data but metadata and the service type is the Discovery Service Type, which recommended interface according to INSPIRE is the OGC CSW specification. In this case CSW offers a transactional operation in its interface to deploy new metadata in the service, thus opposite to the Data Wrapper the interface implemented by Service Publisher is a standard interface, the CSW-T. For these transactional requests, we build and send a HTTP POST request where the body is a standard metadata element included in the standardized XML structure according to the OGC service.

On the other hand once deployed the metadata, it will be available for searching purposes through the CSW interface. Since the CSW implementation of choice offers other interfaces, the deployed metadata will be available not only through the CSW but also through the Open Search interface.

Technology. The Service Publisher is again a self-developed Java library delivered with open source license. The Service Publisher uses the Service Connector to perform the two main steps of its workflow: first, uses the Service Connector to send *GetCapabilities* request to the services in which we have deployed the resources and get a minimum set of metadata elements (available in the Capabilities document) and second, it uses the Service Connector CSWAPI to connect to the catalog (OGC CSW) and deploy this metadata using the transactional interface, which provides methods to insert, delete, and update records in the catalog.

Service Layer

As we based our infrastructure on INSPIRE technical guidelines, the service layer must

provide services for discovery, viewing and downloading of data available in the infrastructure.

Interface. The INSPIRE technical architecture shows that the different service types defined by the Directive are placed in the service layer resulting in the so-called Service Network. Each type defines common capabilities offered by a group of services. Regarding this functionality, the INSPIRE Service Types we consider are the Discovery (INSPIRE Network Services Drafting Team, 2009b), View (INSPIRE Network Services Drafting Team, 2008) and Download (INSPIRE Network Services Drafting Team, 2009a) service types. In addition we also consider the Processing type to add processing capabilities to the SDI (Díaz, Granell, & Gould, 2008).

Since OGC standards are proven to be mature, they have been chosen to implement the interfaces of the services available in the Service layer as the INSPIRE directive recommends in its implementation rules (see parenthesis content: such as OGC CSW (to implement Discovery Service type), OGC WMS (View Service Types), OGC WFS, OGC WCS (Download Service types) and OGC WPS (pointed out to be useful to implement distributed environmental applications (Granell, Díaz, & Gould, 2010).

Technology. There are many implementations of OGC specifications in the open source community. GeoServer is the project of choice to deploy data for visualization and download purposes, that is, through WMS, WFS and WCS specifications. It offers these interfaces and in its latest version GeoServer RESTful API allows the Data Wrapper component to deploy resources and configure the services.

Geonetwork is another open source project broadly extended thanks to its implementation of the OGC CSW specification. This is the project of choice to deploy the discovery service type in our framework. Regarding WPS specification, we choose 52North WPS framework because it is an open source Java framework that enables easily design, implementation and deployment of WPS-based services.

USE CASE

To evaluate our proposal and illustrate with a practical example the behavior of the described components we present a particular use case focused on the evaluation of burned regions on protected environmental areas in Valencia (Spain). The analysis of fire impact requires a real time interaction between different datasets, processes and users. For these reasons it represents a perfect case study to look at the overall functionality of the framework.

A technician must assess the impact of the some summer fires in the protected areas of Valencia, she should find and access the protected areas dataset, the burned areas dataset (for a certain year) and perform the intersection process. Traditionally the result of the assessment can be used to write a report or make a decision; here we only describe how this technician can deploy it easily in the underlying SDI. Figure 4

shows a sequence diagram with the operations and actors involved in each step of this scenario.

The first step is to find the initial data sets, in our case the protected areas of Valencia and the burned areas in 2005. For the first data entry (protected areas), we use a local data provider using a standard OGC data service. These data can be found using the OpenSearch-geo client component.

If the data is not accessible through an OGC service and it is only available locally like the input data related to the burned areas in 2005, Service Framework provides the functionality to upload local datasets to be deployed as standard OGC data service. Thus, the user can upload it in the application. Figure 5 shows a screenshot of how this step can be performed in the user interface.

The second step shown in the sequence diagram (Figure 4) is the execution of a distributed process to perform the intersection and the generation of the assessment results

Figure 4. Sequence diagram

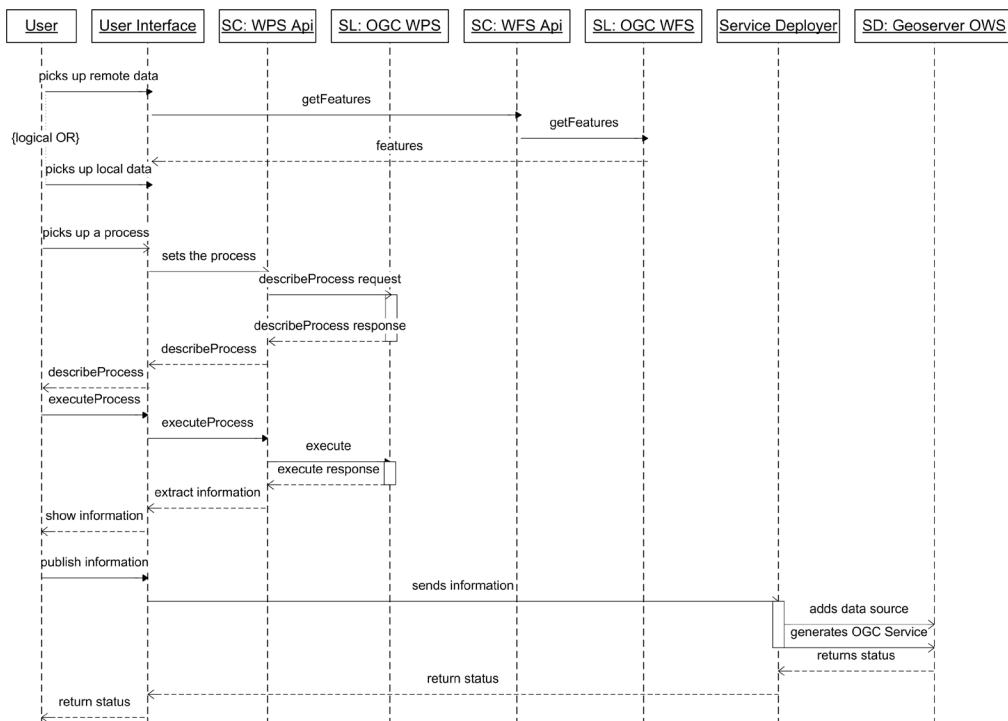
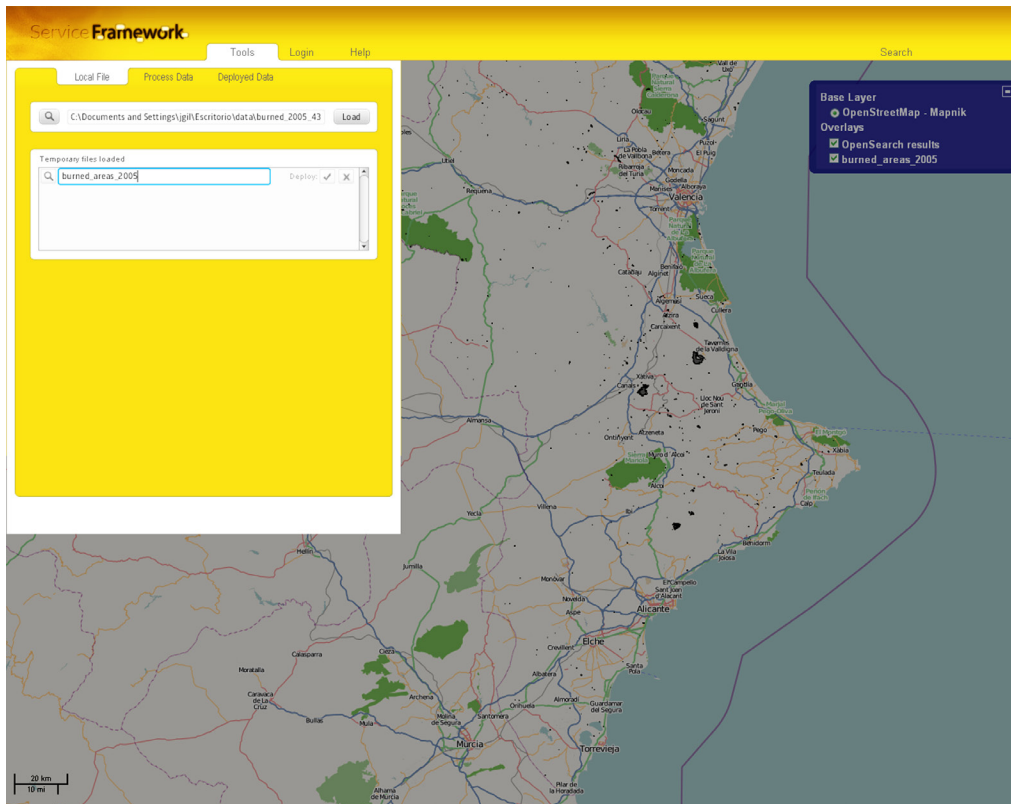


Figure 5. Local data deployment

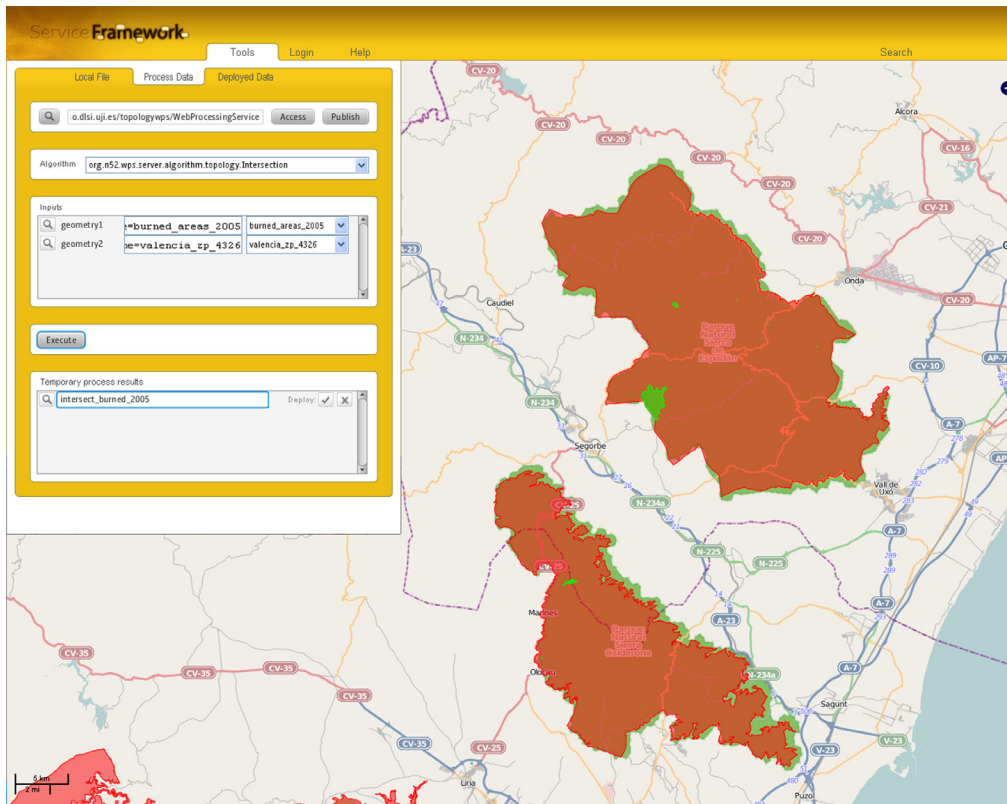


(Figure 6). The Service Framework interface allows the user to connect and execute to existing WPS services, rather than using heavy desktop applications. In this step we must select a WPS server and a specific process, as in our case the Intersection (a process that will return the protected areas that burned in 2005). All communication with the service is through the Service Connector component using the WPS API. In the diagram of Figure 4, we can see how internally several WPS interface calls are performed. First a *describeProcess* request to get the process description regarding input data and afterwards (after the user identifies the correct input data, in this case references to the WFS where we have deployed the data), the *execute* request to run the process.

The last step consists of deploying the processing results to explore the results (Figure

6). Users can thus view it and analyze it through the integrated map viewer component based in OpenLayers. Again as we described with the upload of the burned areas the most interesting feature of the Service Framework is that the Service Framework assists the user to deploy the processing results for visualization and download purposes into standard services, OGC WMS and OGC WFS respectively. The Service Framework uses the GeoServer RESTful API and creates a new data store accessible through standard services. After the data deployment, the Service Publisher component creates a minimum set of metadata elements to be published in an instance of CS/W so that the dataset can be found by other users through common discovery mechanisms in SDI.

Figure 6. Process inputs view and results



CONCLUSION

We have presented a framework that follows a hybrid SDI building methodology. We provide an

INSPIRE-based architecture, whose components assist users not only in accessing but also in deploying resources as INSPIRE-compliant services. In this way we provide an entry point for users to participate in SDI building and maintenance. As a result, users can massively deploy resources improving their availability in a standard manner.

To deploy resources automatically as standard services, the Service Framework uses a concrete implementation of the OGC WMS, WFS, WCS specification. The Service Framework uses Geoserver's own protocol to deploy resources. We still could migrate the solution

to other implementations, but some programming should be done. To avoid this, it would be useful to deploy resources using standard transactional operations. This would require to have the OGC WMS and WCS specifications including a transactional interface like WFS and WPS is considering in its current and future versions. This would detach the solution from the vendor implementation. In this sense OGC standards have shown to be mature enough to provide specifications to create interoperable web services to deploy and share resources and provide the functionality needed to create distributed applications on SDI.

The use of open source components has the advantage of giving us full control over the code to modify everything. A disadvantage, however, is that they have caused some inconveniences because most of them use libraries that are not

fully developed in the context of this project. Documentation scarcity is always a big issue to face. Therefore, there is a need to invest an important amount of time and effort in the development to have satisfying results.

ACKNOWLEDGMENTS

This work has been partially supported by the European project EuroGEOSS and by the CENIT “España Virtual” project through the Instituto Geográfico Nacional (IGN). We would like to thank Jorge Suarez from the *Conselleria de Medi Ambient, Aigua, Urbanisme i Habitatge* (Environmental, Water, Urban planning and living regional government of Valencia), who kindly granted the forest fire data for experimentation purpose.

REFERENCES

- Ball, W. P., Brady, D. C., Brooks, M. T., Burns, R., Cuker, B. E., & Di Toro, D. M. (2008). Prototype system for multidisciplinary shared cyberinfrastructure: Chesapeake Bay environmental observatory. *Journal of Hydrologic Engineering*, 13(10), 960–970. doi:10.1061/(ASCE)1084-0699(2008)13:10(960)
- Bell, M. (2008). *Service-Oriented Modeling (SOA): Service analysis, design, and architecture*. Hoboken, NJ: John Wiley & Sons.
- Benthall, S., & Gill, S. (2010). *SDI Best Practices with GeoNode*. Retrieved June 1, 2010, from FOSS4G website: <http://2010.foss4g.org/tutorial10.php>
- Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., & Weerawarana, S. (2002). Unravelling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI. *IEEE Internet Computing*, 6(2), 86–93. doi:10.1109/4236.991449
- Díaz, L., Granell, C., & Gould, M. (2008). Case study: Geospatial processing services for web based hydrological applications. In Sample, J. T., Shaw, K., Tu, S., & Abdelguerfi, M. (Eds.), *Geospatial services and applications for the Internet* (pp. 31–47). New York, NY: Springer.
- Díaz, L., Granell, C., Gould, M., & Olaya, V. (2008). An open service network for geospatial data processing. In *Academic Proceedings of the Free and Open Source Software for Geospatial Conference*, Cape Town, South Africa (pp. 410-419).
- European Parliament and Council. (2007). *INSPIRE EU Directive*. Retrieved January 28, 2009, from http://eur-lex.europa.eu/LexUriServ/site/en/oj/2007/1_108/1_10820070425en00010014.pdf
- Fielding, R. (2000). *Architectural styles and the design of network-based software architectures* (Unpublished doctoral dissertation). University of California, Irvine, CA.
- Fonts, O., Huerta, J., Díaz, L., & Granell, C. (2010). OpenSearch-geo: The simple standard for geographic web search engines. In *Proceedings of the IV Jornadas de SIG Libre*, Girona, Spain. Retrieved September 1, 2010, from <http://www.sigte.udg.edu/jornadas-siglibre/uploads/Articles/a29.pdf>
- Freeman, E., Sierra, K., & Bates, B. (2004). *Headfirst design patterns*. Sebastopol, CA: O'Reilly Media.
- Giuliani, G., Ray, N., Schwarzer, S., De Bono, A., Peduzzi, P., & Dao, H. (2011). Sharing environmental data through GEOSS. *International Journal of Applied Geospatial Research*, 2(1). doi:10.4018/jagr.2011010101
- Goodchild, M. F. (2008). The use cases of digital earth. *International Journal of Digital Earth*, 1(1), 31–42. doi:10.1080/17538940701782528
- Goodchild, M. F., Egenhofer, M., Fegeas, R., & Kottman, C. (1999). *Interoperating geographic information systems*. Boston, MA: Kluwer Academic.
- Granell, C., Díaz, L., & Gould, M. (2010). Service-oriented applications for environmental models: Reusable geospatial services. *Environmental Modelling & Software*, 25(2), 182–198. doi:10.1016/j.envsoft.2009.08.005
- Granell, C., Gould, M., & Esbri, M. A. (2008). Geospatial web service chaining. In Karimi, H. A. (Ed.), *Handbook of research on geoinformatics* (pp. 189–195). Hershey, PA: Information Science Reference.
- Hey, T., & Trefethen, A. E. (2005). Cyberinfrastructure for e-Science. *Science*, 308(5723), 817–821. doi:10.1126/science.1110410
- INSPIRE Network Services Drafting Team. (2008). *Draft technical guidance to implement INSPIRE view services*. Retrieved from http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/network/Draft_Technical_Guidance_View_Services_v1.0.pdf
- INSPIRE Network Services Drafting Team. (2009a). *Draft technical guidance for INSPIRE download services*. Retrieved from http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/network/Draft_Technical_Guidance_Download_Services_v1.0.pdf

- INSPIRE Network Services Drafting Team. (2009b). *Technical guidance for INSPIRE discovery services*. Retrieved from http://inspire.jrc.ec.europa.eu/documents/Network_Services/Technical%20Guidance%20Discovery%20Services%20v2.0.pdf
- Lannotta, B. (2007). GEOSS: A global view. *Aerospace America*, 45(8), 36–40.
- Masser, I. (2005). *GIS worlds: Creating spatial data infrastructures*. Redlands, CA: ESRI Press.
- Mazzetti, P., Nativi, S., & Caron, J. (2009). RESTful implementation of geospatial services for Earth and Space science applications. *International Journal of Digital Earth*, 2(1), 40–61. doi:10.1080/17538940902866153
- Minsker, B., Myers, J., Marikos, M., Wentling, T., Downey, S., Liu, Y., et al. (2006). NSCA environmental cyberinfrastructure demonstration project: Creating cyber environments for environmental engineering and hydrological science communities. In *Proceedings of the ACM/IEEE Conference on Supercomputing*, Tampa, FL. New York, NY: ACM.
- Morissette, J. T., Richardson, A. D., Knapp, A. K., Fisher, J. I., & Abatzoglou, J. (2009). Tracking the rhythm of the seasons in the face of global change: Phenological research in the 21st century. *Frontiers in Ecology and the Environment*, 7(5), 253–260. doi:10.1890/070217
- Nerbert, D. (2004). *Developing spatial data infrastructure: The SDI cook book, version 2.0*. Victoria, Australia: GSDI.
- Open Geospatial Consortium Inc. (OGC). (2005). *Web feature service implementation specification 1.1.0*. Redlands, CA: Author.
- Open Geospatial Consortium Inc. (OGC). (2006). *Web map service implementation specification 1.3.0*. Redlands, CA: Author.
- Open Geospatial Consortium Inc. (OGC). (2007a). *Sensor observation service (Version 1.0.0)*. Redlands, CA: Author.
- Open Geospatial Consortium Inc. (OGC). (2007b). *Web processing service (Version 1.0.0)*. Redlands, CA: Author.
- Open Geospatial Consortium Inc. (OGC). (2007c). *OpenGIS® catalogue service implementation specification (Version 2.0.2)*. Redlands, CA: Author.
- Open Geospatial Consortium Inc. (OGC). (2008). *Web coverage service implementation specification 1.1.2*. Redlands, CA: Author.
- Papazoglou, M. P. (2008). *Web services: Principles & technology*. Essex, UK: Person Education.
- Pezzoli, K., Marciano, R., & Robertus, J. (2006). Regionalizing integrated watershed management: A strategic vision. In *Proceedings of the 7th Annual International Conference on Digital Government Research*, San Diego, CA (pp. 444-445). New York, NY: ACM.
- Pierce, M. E., Fox, G. C., Choi, J. Y., Guo, Z., Gao, X., & Ma, Y. (2009). Using Web 2.0 for scientific applications and scientific communities. *Concurrency and Computation*, 21(5), 583–603. doi:10.1002/cpe.1365
- Poch, M., Comas, J., Rodríguez-Roda, I., Sánchez-Marrè, M., & Cortés, U. (2004). Designing and building real environmental decision support systems. *Environmental Modelling & Software*, 19(9), 857–873. doi:10.1016/j.envsoft.2003.03.007
- Pumphrey, M. (2009). *OpenGeo Suite*. Retrieved June 1, 2010, from FOSS4G website: http://wiki.osgeo.org/wiki/FOSS4G_2009_Demonstration_Theatre
- Rajabifard, A., Binns, A., Masser, I., & Williamson, I. (2006). The role of sub-national government and the private sector in future spatial data infrastructures. *International Journal of Geographical Information Science*, 20(7), 727–741. doi:10.1080/13658810500432224
- Ramapriyan, H. K. (2008). NASA's Earth science data systems: A 'bit of history' and observations. In *Proceedings of the Cyberinfrastructure for Environmental Observations, Analysis, and Forecasting: A Cyberinformatics Forum*.
- Reichert, P., & Schuwirth, N. (2010). A generic framework for deriving process stoichiometry in environmental models. *Environmental Modelling & Software*, 25(10), 1241–1251. doi:10.1016/j.envsoft.2010.03.002
- Richardson, L., & Ruby, S. (2007). *RESTful Web Services*. Sebastopol, CA: O'Reilly Media.
- Secretariat, G. E. O. (2007). *The full picture: GEO*. Geneva, Switzerland: Tudor Rose.
- Steiniger, S., & Bocher, E. (2009). An overview on current free and open source desktop GIS developments. *International Journal of Geographical Information Science*, 23(10), 1345–1370. doi:10.1080/13658810802634956
- Weaver, J. L., Gao, W., Chin, S., & Iverson, D. (2009). *Pro JavaFX Platform: Script, Desktop and Mobile RIA with Java Technology*. New York, NY: APress.

Westlund, S. (2010). The use of geospatial technologies in disaster management. *International Journal of Applied Geospatial Research*, 1(3), 17–30. doi:10.4018/jagr.2010070102

Wilkins-Diehr, N., Gannon, D., Klimeck, G., Oster, S., & Pamidighantam, S. (2008). TeraGrid science gateways and their impact on science. *Computer*, 41(11), 32–42. doi:10.1109/MC.2008.470

Yang, C., & Raskin, R. (2009). Introduction to distributed geographic information processing research. *International Journal of Geographical Information Science*, 23(5), 553–560. doi:10.1080/13658810902733682

Yang, C., Raskin, R., Goodchild, M., & Gahegan, M. (2010). Geospatial Cyberinfrastructure: Past, present and future. *Computers, Environment and Urban Systems*, 34, 264–277. doi:10.1016/j.compenurbsys.2010.04.001

Yang, P., Li, W., Xiao, D., Raskin, R., & Rambacus, M. (2007). Earth information exchange - Sharing earth science information through interoperable approach and cyberinfrastructure. In *Proceedings of the SPIE Conference on Geoinformatics: Geospatial Information Science*, Nanjing.

ENDNOTES

- ¹ All specifications for service interfaces defined by OGC are publicly available at <http://www.opengeospatial.org>
- ² Workshops, tutorials and presentations for every conference edition are publicly available at <http://foss4g.org>
- ³ The OpenSearch specification and its extensions can be found at <http://www.opensearch.org>
- ⁴ <http://docs.geoserver.org/stable/en/user/extensions/rest/index.html>

José Gil received his MSc in computer science from the Jaume I University in 2004. Currently he is a researcher at the Interactive Visualization Group at Jaume I University and he is pursuing his PhD in computer science from Jaume I University. His research areas include real-time visualization, computer simulations and Java technology.

Laura Diaz graduated in Computer Engineering (2000) (Universitat de València). She obtained a Master in Intelligent Systems (2008) and received her doctorate in Geospatial Science (2010) at the Universitat Jaume I of Castellón where she is currently a postdoctoral researcher. Her research lines are approaches for improving geoservices descriptions and deployment, distributed geoprocessing and service interoperability. She has participate in R&D projects in the Institute of Robotics of the Universitat de València and in GIS companies such as Geodan (The Netherlands) and Iver TI (Spain).

Carlos Granell, PhD (Universitat Jaume I, Spain), is a postdoctoral researcher interested in the study of Spatial Data Infrastructures (SDI), interoperability, geoprocessing services, workflow, and the composition and reuse of geospatial services applied to the environmental domain. He has taken part in several public funded research projects, both at Spanish and European levels. He has carried out research visits at SINTEF (Norway), ITC (the Netherlands), University of Nottingham (UK). Dr. Granell is currently affiliated with the European Commission, Joint Research Centre, Institute for Environment and Sustainability, Spatial Data Infrastructures Unit, Ispra, Italy.

Joaquín Huerta is an associate professor in the Department of Information Systems at UJI, where he teaches GIS and Internet Technologies. He holds a PhD in Computer Science from Jaume I University. He is currently leading two EU projects at UJI: EUROGEOSS (FP7) and eSDI-NET+ (FP6) and a Spanish project “España Virtual” funded by Cenit Programme. His current research interests are Geospatial Technologies and Computer Graphics. He is Director of the Master in Geospatial Technologies funded by Erasmus Mundus program. In addition to academic activities Dr. Huerta is founding board member of an Internet service provider and, thus, possesses considerable business experience as well as experience in systems integration.

CALL FOR ARTICLES

International Journal of Applied Geospatial Research

An official publication of the Information Resources Management Association

The Editor-in-Chief of the *International Journal of Applied Geospatial Research* (IJAGR) invites authors to submit manuscripts for consideration in this scholarly journal.

MISSION

The *International Journal of Applied Geospatial Research* (IJAGR) publishes research that exemplifies the usage of geographic information science and technology (GIS&T) to explore and resolve geographical issues from various application domains within the social and/or physical sciences. IJAGR is designed to provide planners and policy analysts, practitioners, academicians, and others using GIS&T useful studies that might support decision-making activities.

TOPICS OF INTEREST:

- Biogeography
- Business and marketing geography
- Climatology
- Economic geography
- Geography of crime
- Geomorphology
- Historical geography
- Medical geography
- Military geography
- Natural hazards
- Political geography
- Population geography
- Soil geography
- Tourism geography
- Transportation geography
- Other geographic subfields



ISSN 1947-9654
eISSN 1947-9662
Published quarterly

**All submissions should be e-mailed to:
Donald P. Albert, Editor-in-Chief
geo_dpa@shsu.edu**

Ideas for Special Theme Issues may be submitted to the Editor-in-Chief.

**Please recommend this publication to your librarian. For a convenient
easy-to-use library recommendation form, please visit:
<http://www.igi-global.com/ijagr>**