OSGeo Journal Volume 8

February 2011

FOSS4G 2009
Conference Proceedings

OSGeo Community
News & Announcements
Case Studies
Integration Examples

FOSS4G

DENVER 2011

SEPTEMBER 12-16

The Annual International

FREE & OPEN SOURCE
SOFTWARE FOR GEOSPATIAL

Conference Event

2011.FOSS4G.ORG

OSGeo
Your Open Source Compass

Volume 8 Contents

From the Editor

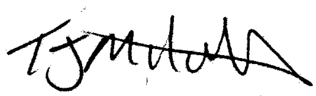
OSGeo has just past its 5th birthday, along with this 8th volume of the OSGeo Journal! With this edition we bring a few news headlines from the past couple months, a few general articles and, most significantly, several top papers from the **FOSS4G 2009** conference event held in Sydney, Australia.

The Journal has become a diverse platform for several groups and growth in each area is expected to continue. The key groups that read and contribute to the Journal include software developers sharing information about their projects or communities, power users showing off their solutions, academia seeking to publish their research and observations in a peer-reviewed, open source friendly medium. OSGeo also uses the Journal to share community updates and the annual reports of the organisation.

Welcome to those of you who are new to the OSGeo Journal. Our Journal team and volunteer reviewers and editors hope you enjoy this volume. We also invite you to submit your own articles to any of our various sections. To submit an article, register as an "author" and sign in at `http://osgeo.org/ojs`. Then when you log in you will see an option to submit an article.[1]

We look forward to working with, and for, you in the upcoming year. It's sure to be an interesting year as we see OSGeo, Open Source in general and all our relate communities continue to grow. Nowhere else is this growth more apparent than at our annual conference: **FOSS4G 2011 Denver**, September, 2011.[2] Keep an eye on your OSGeo mailing lists, blogs and other feeds to follow the latest FOSS4G announcements, including the invitation to submit presentation proposals.[3] It will be as competitive as ever to get a speaking slot, so be sure to make your title and abstract really stand out.

Wishing you the best for 2011 and hoping to see you in Denver!



Tyler Mitchell
`tmitchell@osgeo.org`
Editor in chief, OSGeo Journal
Executive Director, OSGeo

[1]The direct URL for article submission is: `https://www.osgeo.org/ojs/index.php/journal/author/submit`
[2]FOSS4G 2011 Denver: `http://2011.foss4g.org`
[3]FOSS4G 2011 Abstract Submission: `http://2011.foss4g.org/program`

FOSS4G 2009 Conference Proceedings

From the Academic Track Chair

Prof. Thierry Badard

The FOSS4G 2009 academic track aimed to bring together researchers, developers, users and practitioners – all who were carrying out research and development in the free and open source geospatial fields and who were willing to share original, recent developments and experiences.

The primary goal was to promote cooperative research between OSGeo developers and academia, but the academic track has also acted as an inventory of current research topics. This track was the right forum to highlight the most important research challenges and trends in the domain and let them become the basis for an informal OSGeo research agenda. It has fostered interdisciplinary discussions in all aspects of the free and open source geospatial domains. It was organized to promote networking between the participants, to initiate and favour discussions regarding cutting-edge technologies in the field, to exchange research ideas and to promote international collaboration.

In addition to the OSGeo Foundation[23], the ICA (International Cartographic Association) working group on open source geospatial technologies[24]) was proud to support the organisation of the track.

The coordinators sought to gather paper submissions globally that addressed theoretical, technical, and practical topics related to the free and open source geospatial domain. Suggested topics included, but were not limited to, the following:

- State of the art developments in Open Source GIS
- Open Source GIS in Education
- Interoperability and standards - OGC, ISO/TC 211, Metadata
- Spatial Data Infrastructures and Service Oriented Architectures
- Free and open source Web Mapping, Web GIS and Web processing services
- Cartography and advanced styling
- Earth Observation and remote sensing
- Spatial and Spatio-temporal data, analysis and integration
- Free and Open Source GIS application use cases in Government, Participatory GIS, Location based services, Health, Energy, Water, Urban and Environmental Planning, Climate change, etc.

In response to the call for papers, 25 articles were submitted to the academic track. The submissions were highly diversified, and came from USA, Canada, Thailand, Japan, South Korea, Sri Lanka, Australia, New Zealand, Italy, Denmark, France, Germany, Switzerland, Romania and Turkey. Selection of submissions were based on the full papers received. All submissions were thoroughly peer reviewed by two to three members of the international scientific committee and refereed for their quality, originality and relevance. The scientific committee selected 12 papers (48% acceptance rate) for presentation at the FOSS4G 2009 conference. From those, 6 papers were accepted for presentation in the proceedings of the academic track, which are published in this volume of the OSGeo Journal. They correspond to the 6 best papers assessed by the international scientific committee.

The accepted and published papers covered a wide

[23]OSGeo: Open Source Geospatial Foundation: http://osgeo.org
[24]ICA open source working group: http://ica-opensource.scg.ulaval.ca/

range of cutting-edge research topics and novel applications on Free and Open Source Geospatial technologies. I am particularly proud and happy to see some very high quality scientific contributions published in the OSGeo Journal. This will undoubtedly encourage more interesting research to be published in this volume, as our OSGeo journal is an open access journal. In addition, it helps draw attention to this important project of the OSGeo Foundation. I hope the publication of these proceedings in the OSGeo journal will encourage future scientists, researchers and members of academia to consider the OSGeo Journal as an increasingly valuable place to publish their research works and case studies.

As a concluding note, I would like to take the opportunity to thank the individuals and institutions that made the FOSS4G 2009 academic track possible. First, I would like to thank the international scientific committee members and external reviewers for evaluating the assigned papers in a timely and professional manner. Next, I would like to recognize the tremendous efforts put forward by members of the local organising committee of FOSS4G 2009 for accommodating and supporting the academic track. Finally, I want to thank the authors for their contributions, efforts, patience and support that made this academic track a huge success.

January, 2011
Prof. Thierry Badard
Laval University, Canada
Chair, FOSS4G 2009 Academic Track
Co-chair, ICA Working Group on Open Source Geospatial Technologies

Geoprocessing in the Clouds

Bastian Baranski, Bastian Schaeffer, Richard Redweik

Abstract

Cloud Computing is one of the latest hypes in the mainstream IT world. Spatial Data Infrastructures (SDIs) with its classical publish-find-bind paradigm have not been affected yet by this emerging trend. This paper reviews this novel technology and tries to identify the paradigm behind it. In particular, the scalability aspect for a cloud enabled 52°North Open Source Web Processing Service is challenged and proven in the exemplary Google Cloud. On this basis, future direction for SDIs and the Cloud Computing paradigm are identified.

Introduction

Cloud Computing is one of the latest trends in the mainstream IT world (5). The term Cloud Computing uses a cloud metaphor to represent the internet or other large networking infrastructures. From a provider perspective, the key aspect of the cloud is the ability to dynamically scale and provide computational power, storage, and other applications, even complete infrastructures in a cost efficient and secure way over the internet. From a client perspective, the key aspect of a cloud is the ability to access the cloud facilities on-demand without managing the underlying infrastructure and dealing with the related investments and maintenance costs.

However, existing Spatial Data Infrastructures (SDI) are mostly focused on data retrieval and data visualization (8). Migrating the data processing part from classical desktop application to a distributed environment could be regarded as the next step. The step after migrating to a distributed environment would be the adoption of Cloud Computing principles. While the processing part in SDIs has already been tackled (12) (3) (13), Cloud Computing has not been regarded in the context of SDIs yet. This was the starting for this paper to explore the capabilities of Cloud Computing with a special focus on the processing part in SDIs.

In general, there are two options for realizing Cloud Computing in SDIs. First, adopting Cloud Computing principles and standards to SDIs. Second, migrating SDI elements amongst other services on top of a Cloud Computing infrastructure. Following the first option, the geodomain would once again create their own separate standards and markets and therefore establishing new barriers for utilizing SDIs. From our perspective the second option would be more effective and would allow the Geoinformation (GI) domain to be open to the mainstream IT world and thereby broaden the limited GI market. By leveraging these core propositions, we believe that the paradigm behind the Cloud Computing buzzword is promising for geospatial applications in order to enable new and promising business models for building up, operating and utilizing SDIs. In order to get hands-on experience, the 52°North WPS implementation [25] was migrated as a proof-of-concept study into the Google Cloud (namely the Google App Engine platform).

The remainder of this paper is structured as follows. First, a review of the basic concepts and related technologies is provided. This is followed by a description of the technical concept of the WPS migration into the Google Cloud. In the next section, our technical concept is evaluated in terms of scalability as one of the key aspects of Cloud Computing. Finally, the paper ends with a conclusion about the described framework and a discussion about interesting topics for a further research agenda.

Background

This section provides a review of related work in the context of Cloud Computing and SDI concepts.

Cloud Computing

Cloud Computing is one of the latest trends in the mainstream IT world (4) (5) and several companies such as Amazon, Google, Microsoft and Salesforce have already build up significant effort in this direction. The term Cloud Computing uses a cloud metaphor to represent the internet or other large networking infrastructures and the paradigm behind the buzzword hints at a future in which the storage of data and computations are no longer performed on local computers, but on distributed facilities operated by third-party storage and computational utilities (2). The term Cloud Computing overlaps with some concepts of Distributed Computing and Grid Computing (6). Grid Computing and Cloud Computing are both scalable infrastructures and provide sufficient computational resources like storage or computational power. But the target audience of Grid Computing is typically the scientific community running large-scale simulations and resource- and time-consuming applications (for example a global climate change model or the aerodynamic design of engine components), whereas with Cloud Computing small and medium-sized companies can scale their web-based applications in an instant fashion without having to invest in infrastructure for storing or processing large amounts of data (10). Furthermore, national

[25] http://www.52north.org/wps
[26] http://lcg.web.cern.ch/LCG/

and international Grid infrastructures (for example the Worldwide LHC Computing Grid [26]) are typically governmentally funded and driven by international joint research projects (for the example the Large Hadron Collider, LHC project at CERN [27]), whereas cloud infrastructures are operated by large enterprises under economic aspects.

Characteristics

The key characteristics of the cloud are the ability of datacenter providers to scale and provision computational resources, storage, and other applications even complete infrastructures dynamically in a cost efficient and secure way over the internet. Besides the consumer is given the ability to use these resources without having to manage the underlying complexity of the technology. These characteristics open up new perspectives for tackling different problems and lead to the following set of core value propositions.

Efficiency Cloud Computing enables IT organizations to increase hardware utilization rates enormously and to scale up to massive capacities in an instant without heavily investing in infrastructure in advance. Datacenter providers are now able to utilize their infrastructure more efficiently by dynamically distributing their applications and processes to free available resources.

Outtasking By outtasking software and data to scalable facilities operated by third parties, users and customers don't have to operate their own datacenters anymore. Therefore, enterprises of all types - from Web 2.0 startups to global enterprises - can decrease their infrastructure costs enormously. They can take advantage of transforming their fixed IT costs into variable costs as a business advantage by focusing on their core business (rather spending time on developing mature software and innovative business models than managing their physical hardware and purchasing costly licenses for rarely used software).

Scalability The allocation of cloud resources (for example high capacity storage or computing power) is done in real-time and most cloud infrastructures scale the deployed applications automatically on demand (for example in case of high request rates). This gives cloud users and cloud application providers the option for handling peak load very efficiently without operating their own datacenter and without managing their own infrastructure. For example, load-balancing or the development of high availability solutions for their software does not need to be regarded because it is provided by the cloud implicitly. By deploying their software and data in the cloud, they are automatically able to scale up their business capacities (for example from a few to hundreds of servers) in an instant and on demand fashion.

On-demand Allocating cloud resources on a real-time and on-demand basis helps enterprises to scale up their business capacities in an instant and efficient way. The absence of long-term contracts in combination with pay-per-use revenue models allows the low-cost startup of new ideas for business models. The total cost of ownership (including hardware, software licenses, energy, fail-safety and technical engineers) of self-hosted datacenters minimizes start-up costs and helps enterprises to put new promising business models into the market.

Additional features of Cloud Computing infrastructures are the application of Service Level Agreements (SLA) defining service quality guarantees and contractual penalty clauses if the providers fail to meet the guaranteed service quality goals. Such contracts are important for cost-performance ratio transparency and therefore an essential skill for all kinds of IT and in this sense also IT based geospatial business models.

In essence, Cloud Computing is not a completely new concept. It moreover collects a family of well known and established methods and technologies (for example SaaS as a model for software packaging and deployment and Virtualization as an efficient hosting platform (7)) under the umbrella of the term Cloud Computing. Besides, it describes a paradigm of outsourcing applications and specific tasks to a scalable infrastructure and therefore consequently enabling new business models with less up-front investments. Keeping in mind that these technologies and general concepts existed in the IT industry for years, the emergence of high network bandwidth and mature virtualization technologies has now enabled this paradigm for a broader audience and leads to new application development models.

There are still a number of open issues for Cloud Computing. One deals with the general barriers of adopting Cloud Computing and is examined for example in the so-called "Open Cloud Manifesto". Beside data backup and recovery responsibilities the outsourcing of confidential and economically relevant data from data owners facilities to third party infrastructures is problematic in context of trust. Using public clouds as a deployment platform for applications and services in a risk management scenario is already a security issue in situations when the underlying cloud suffers an outage. But the problems regarding outsourcing of data and reliability of infrastructures are not specific only for cloud infrastructures. They must be addressed for all kinds of distributed architectures.

Projects and Initiatives

A lot of enterprise corporations are trying to get into the Cloud Computing business by offering services to ac-

[27] http://lhc.web.cern.ch/lhc/

cess their huge and over years grown infrastructures to the public. Microsoft with the Azure Services Platform [28] and its upcoming operating system Windows Azure [29] for operating cloud infrastructures, IBM introduced their "Blue Cloud" platform [30] and SUN for example offers Cloud Computing solutions as well [31]. In this chapter we describe the two cloud solutions from Amazon and Google more detailed, showing clearly that cloud providers could realize the different layers and characteristics of a cloud infrastructure at a different level of detail.

The Amazon Web Services (AWS) product is a collection of services that are offering Infrastructure as a Service (IaaS), Datastorage as a Service (dSaaS) and some aspects of Platform as a Service (PaaS). The Amazon Elastic Compute Cloud (Amazon EC2) provides a web service interface to manage virtual machines (IaaS) that are used to host customer specific applications and can be scaled on-demand to handle peak load. The Amazon Simple Storage Service (Amazon S3) provides a web services interface that can be used to store and retrieve large amounts of data (dSaaS). The Amazon Elastic MapReduce is a web service that offers computational power to process efficiently vast amounts of data. It utilizes the Hadoop [32] framework and dynamically distributes data and processing tasks across an automatically scaled cluster of computation nodes.

In contrast of AWS, the Google App Engine is an adequate example for pure PaaS. The Google App Engine provides a sandbox for running Java- and Python-based web applications. The web applications are deployed on the Google infrastructure and so they can take advantage of the same scalable and load balancing technologies that Google applications are built on. On the one hand, the key advantage of Google App Engine over AWS is that Google App Engine offers an easy way of deploying web applications in the cloud. In particular, the overhead of dealing with virtual machines and entire (virtual) server systems could be neglected. The Google App Engine offers also a data storage service (dSaaS) and different bindings to existing Google applications for authentication and accounting. Besides, the free default quota for testing purposes (for example data transfer and CPU time) lowers also the barrier for a first trial experiment. On the other hand, applications deployed in the Google App Engine are restricted to a specific (Java- or Python-based) application framework that runs in a restricted sandbox. This sandbox forbids the creation of threads and the web service request duration is limited to 30 seconds. Furthermore, the Google App Engine platform does not support the MapReduce programming model (1) or related methods for distributed processing and generating efficiently large data sets. Therefore, the Google App Engine platform is currently not suitable for performing large-scale and time-consuming geospatial processes.

Beside these and other commercial cloud providers, different projects and initiatives drive the general development of Cloud Computing technologies and especially the development of open standards for interoperability in clouds. The Open Cloud Consortium (OCC) [33] for example is an initiative dedicated to Cloud interoperability and initiated the Open Cloud testbed [34]. The Open Cirrus Project [35] is cloud computing research testbed between research and industry partners. In the Eucalyptus [36] initiative, an open source based implementation of the Amazon API is under development.

Web Processing Service

The Open Geospatial Consortium (OGC) Web Processing Service interface specification (11) describes a standardized method to publish and execute web-based processes for any type of geoprocesses. According to the WPS interface specification, a process is defined as any calculation operating on spatially referenced data. In detail, the WPS interface specification describes three operations, which are all handled in a stateless manner: GetCapabilities, DescribeProcess and Execute. GetCapabilities is common to any type of OGC Web Service and returns service metadata. In case of WPS it also returns a brief description of the processes offered by the specific service instance. To get more information about the hosted processes, the WPS provides process metadata through the DescribeProcess operation. This operation describes all parameters, which are required to run the process. Based on this information the client can perform the Execute operation upon the designated process. As every OGC Web Service, the WPS communicates through HTTP-GET and HTTP-POST based on an OGC-specific XML-message encoding. Besides this basic communication pattern, the WPS interface provides functionality for scalable processing such as asynchronous processing (implemented using the pull model), storing of process results and processing of data references encoded as URLs. The application of URL references as input for specific processes is a promising feature, as it limits the volume of data sent

[28] http://www.microsoft.com/azure/default.mspx
[29] http://www.microsoft.com/azure/windowsazurefordevelopers/default.aspx
[30] http://www.ibm.com/ibm/cloud/
[31] http://www.sun.com/solutions/cloudcomputing/index.jsp
[32] http://hadoop.apache.org/core/
[33] http://www.opencloudconsortium.org/
[34] http://www.opencloudconsortium.org/
[35] https://opencirrus.org/
[36] http://www.eucalyptus.com/

between client and service and allows the service to apply specific caching strategies. The service retrieves the data once and reuses it multiple times, by using the reference as an identifier for data.

Concept

On a technical level, the classical 52°North WPS is implemented as a Java Servlet. Due to platform independence gained by Java programming language and the Google App Engine Java support, the WPS components could be easily compiled with a standard Java compiler on a local machine and the resulting package could be deployed on the Google App Engine platform which runs with its own java virtual machine.

As our first test case, we implemented a simple buffer process, which takes two inputs. First, geographic features to be buffered encoded as GML (for example provided by an OGC Web Feature Service, WFS) and second, a distance for the buffer calculation. As a result, geographic features representing the buffers around the input geographic features are computed. The resulting dataset could be fetched either encoded as GML (as exercised with uDig [37]) or KML (as exercised with Google Earth [38]). According to the number of requests, the deployed application is able to scale up by means of the Google cloud mechanisms. Furthermore, native Google cloud services such as authentication could be used directly in the cloud from the deployed application.

In general, the deployed WPS provides geoprocesses to customer, which is the classical SaaS aspect. This is built on the PaaS aspect, which fosters the automatic scalability.

By deploying the WPS in the Google Cloud, the enduser still is able to find and bind a single URL representing the WPS, even though multiple instance exist on the sever side to maintain a scalable service. Therefore the classical publish-find-bind SDI paradigm (9) is not modified by using cloud technologies. However, the use of standardized interfaces such as a WPS ensures interoperability from the client perspective, cloud interoperability from a provider perspective is not given, since every cloud infrastructures has its own APIs and requirements.

Scalability Evaluation

Scalability is one of the key aspects of Cloud Computing. Therefore, we tested our approach and the Google cloud in this direction. We used a stress test to simulate a high demand of simultaneous requests and expected a constant response time by the WPS deployed in the cloud in contrast to a linear rising response time by a non-cloud setting.

Methodology

The WPS was stress tested with the simple buffer algorithm, deployed on the Google App Engine as well as on a local and non cloud enabled Tomcat installation. The geometric data for that process were also delivered via a web service (deployed at the Google App Engine platform in the first case and deployed on the local and non cloud enabled machine in the second case). A cumulative approach was used, starting with 1 and up to 200 requests that were sent simultaneously to the deployed services. The elapsed time from sending the request to receiving the response on its own, as well as for the cumulative sum of the requests/response times was measured. In order to compare the local setting with the remote cloud setting, the results are normalized by only regarding the response time relatively to the maximum/minimum interval of all requests to the specific machine.

Results

Figure 1 shows the normalized response time of the online as well as of the local deployed WPS over the number of simultaneously sent requests. The response time of the remote WPS (blue) stays nearly constant up to 200 simultaneous requests whereas the local WPS response time (red) grows linearly.

Evaluation

The performance evaluation shows to some degree that Google App Engine's scale at high request rates, as the response time for many simultaneous requests stays nearly constant in contrast to the non-cloud deployment.

The slight increase of the response time of the WPS deployed at the Google App Engine platform could be explained by some bottlenecks concerning the data allocation from an external server, laborious internal processing steps in the performance testing tool and high traffic at the local machine and in the local subnetwork when running the performance testing tool. A slight overhead for replicating new service instances on the server side could also be assumed.

Conclusion and Outlook

This paper presented and tested an approach of bringing the OGC Web Processing Service to the cloud. On a conceptual level, we showed that Cloud Computing is not a completely new concept and applied to SDI, the classical publish-find-bind pattern does not have to be modified. Therefore, we see a paradigm shift

[37] http://udig.refractions.net/
[38] http://earth.google.de/

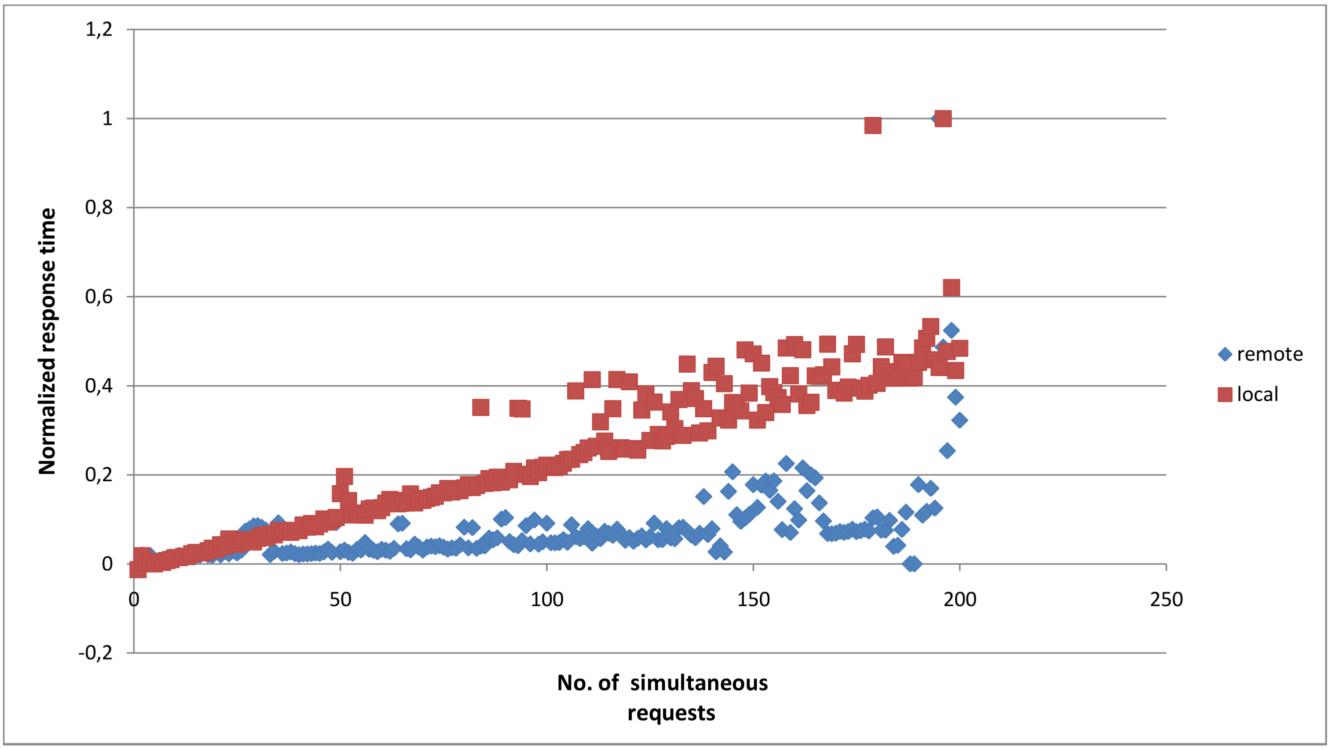


Figure 1: Comparison of normalized response time of remote (blue) and local (red) deployed WPS over number of simultaneously requests.

from technological to economical aspects in contrast to a complete paradigm change. On a technical level, our tests showed that by using the Google Cloud, response times could be held almost constant in contrast to a non-cloud approach. However, our tests also showed, that for the cloud approach, bottlenecks outside the cloud have to be taken into account and could eliminate the positive cloud effects if not carefully evaluated. Nevertheless, the tests showed that Cloud Computing keeps its promises and should be regarded further in sophisticated setups.

Thus, we plan in the next evolution phase to extend the described scenario, which mainly incorporates SaaS aspects, towards a more complex scenario, which takes near real-time air quality sensor data, stored already in the cloud (IaaS or dSaaS) and provided through standardized OGC interfaces (for example OGC Sensor Observation Service, SOS), and interpolate these data in the cloud (for example via WPS). Thereby, we aim at keeping the response time constant using efficient (despite possible high request rates). Besides, another goal for the next iteration phase will be the integration of existing Google App Engine services (for example Mail for alerting and Google Accounts for authentication) and efficient methods for distributed processing as well as storing large dataset (for example MapReduce and the Hadoop platform) into the framework.

Nevertheless, the presented approach is to our knowledge the first OGC compliant cloud service ever and could pave the way for a paradigm shift in SDIs. On the basis of our past experience we still believe that Cloud Computing is promising for building up, operating and utilizing SDI in an effortless way and promising for geospatial applications to enable new business models with less up-front investments. Furthermore, Cloud Computing could be potentially the missing element to popularize SDIs to a broader non-expert community (for example in an effortless way by means of Web 2.0 applications, such as mashups, open collaboration, social networking and mobile e-commerce). In particular, we could think of using OGC interfaces as the standardized way for obtaining geospatial resources (data/processing) similar to added-value services already provided in clouds such as Google Mail. However, by using OGC interfaces, cloud interoperability even from a provider perspective in regard to geospatial resources could be gained.

To further advance the adoption and combination of Cloud Computing and SDI, the 52°North Geoprocessing Community members will continue their basic research by addressing the following questions and topics:

- How can Cloud Computing lower the barriers for building, operating and utilizing SDIs?
- How can Cloud Computing promote innovative and promising geospatial e-commerce models?
- How can Cloud Computing popularize geospatial applications to a broader and collaborating commu-

nity?

- How can SDI elements be mapped to the Cloud Computing paradigm?
- Development and implementation of a fully Cloud Computing enabled SDIs by extending our approach with other Cloud Computing aspects.
- Security aspects such as Authentication, Authorization, Accounting and Delegation.

Bastian Baranski
Research Associate, Institute for Geoinformatics at University of Muenster, Germany
baranski AT uni-muenster.de
http://ifgi.uni-muenster.de/

Bastian Schaeffer
Research Associate, Institute for Geoinformatics at University of Muenster, Germany
schaeffer AT uni-muenster.de
http://ifgi.uni-muenster.de/

Richard Redweik
Bachelor Student, Institute for Geoinformatics at University of Muenster, Germany
richard.redweik AT uni-muenster.de
http://ifgi.uni-muenster.de/

Bibliography

[1] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters., 2008.

[2] I. Foster, Y. Zhao, I. Raicu, and S. Lu. Cloud computing and grid computing 360-degree compared, 2008.

[3] A. Friis-Christensen, N. Ostlnder, M. Lutz, and L. Bernard. *Transactions in GIS*, chapter Designing service architectures for distributed geoprocessing: Challenges and future directions., pages 799–818. 2007.

[4] Gartner. Gartner says cloud computing will be as influential as e-business. Gartner press release, Gartner, 2008.

[5] Gartner. Gartner says cloud application infrastructure technologies need seven years to mature. Gartner press release, Gartner, 2009.

[6] K. Hartig. What is cloud computing? the cloud is a virtualization of resources that maintains and manages itself. *.NET Developers Journal*, 2008.

[7] Sun Microsystems Inc. Cloud computing at a higher level. Technical report, Sun Microsystems Inc., 2009.

[8] C. Kiehle, K. Greve, and C. Heier. *Transactions in GIS*, chapter Requirements for Next Generation Spatial Data Infrastructures-Standardized Web Based Geoprocessing and Web Service Orchestration, pages 819–834. 2007.

[9] I. Masser. *GIS worlds: Creating spatial data infrastructures*. ESRI Press, Redlands, California, USA, 2005.

[10] J. Myerson. Cloud computing versus grid computing - service types, similarities and differences, and things to consider. Technical report, IBM Corporation, 2008.

[11] OGC. Opengis web processing service, version 1.0.0. Technical report, Open Geospatial Consortium (OGC), 2007.

[12] B. Schaeffer, B. Baranski, T. Foerster, and J. Brauner. A service-oriented framework for realtime and distributed geoprocessing. In *International Opensource Geospatial Research Symposium*, 2009.

[13] A. Weiser and A. Zipf. *Lecture Notes in Geoinformation and Cartography*, chapter Web Service Orchestration (WSO) of OGC Web Services (OWS) for Disaster Management, pages 239–254. Springer, New York, 2007.

Imprint

Acknowledgements
Daniel Holt, LaTeX magic & layout support
Various reviewers & writers

The *OSGeo Journal* is a publication of the *OSGeo Foundation*. The base of this journal, the LaTeX 2$_\varepsilon$style source has been kindly provided by the GRASS and R News editorial boards.

Journal online: http://www.osgeo.org/journal

OSGeo Homepage: http://www.osgeo.org

Mail contact through OSGeo, PO Box 4844, Williams Lake, British Columbia, Canada, V2G 2V8

9 771994 189002 00

ISSN 1994-1897

OSGeo

Your Open Source Compass

osgeo.org