# Using Cloud-Based Geographical Information System to Support Earthquake Research and Disaster Response

Jun Wang[1*], Marlon E. Pierce[1], Yu Ma[1], Geoffrey C. Fox[1,2], Andrea Donnellan[3], Jay W. Parker[3], Margaret Glasscoe[3]

[1]Pervasive Technology Institute
Indiana University
2719 East 10th Street
Bloomington, IN 47408

[2]School of Informatics
901 E. 10th St.
Bloomington, IN 47408

[3]Jet Propulsion Laboratory
National Aeronautics and Space Administration
4800 Oak Grove Drive
Pasadena, CA 91109

[*] Corresponding Author
Email: wang208@indiana.edu
Phone: 812-856-1937
Fax: 812-856-7972

Classification: Physical Sciences, Computer Sciences

# Using Cloud-Based Geographical Information System to Support Earthquake Research and Disaster Response

**Abstract**: As earthquake research has expanded beyond the geophysics community to include the requirements of rapid emergency response and disaster management, the computational infrastructure that supports this research must also expand to deliver both data and analysis tools to a wide variety of users. In this paper, we investigate Cloud-based Geographical Information System (GIS) technologies that enable an open-architecture cyberinfrastructure to provide standards-compliant data products and computing services for both earthquake research and disaster planning and response. We evaluate this infrastructure with examples from two earthquake study projects: QuakeSim and E-DECIDER. Based on these case studies, we discuss gaps and research opportunities in Cloud Computing and GIS technology.

**Keywords: Cloud Computing, GIS, Earthquake, Web Service**

## Introduction

The high demand for international collaboration on the rapid emergency response and disaster management from the recent major earthquake disasters in Haiti (2010) and Japan (2011) has put the study of earthquakes into a new perspective. These events demonstrate that geophysicists must not only work effectively through different research groups but also find ways to rapidly and effectively deliver their knowledge, tools, and results to emergency planners and responders. As participants in the emergency response efforts on Japan 2011 earthquake organized by International Charter (http://www.disasterscharter.org/), our experiences highlighted the lack of infrastructure for timely distributing and processing the huge amount of geospatial data for emergency response. It is important to build the computational infrastructure to support this transformation.

Cloud Computing [1,2] provides elastically provisioned computing, software, and service infrastructure, typically implemented on a foundation of virtual machine and virtual data storage technologies. Commercial offerings include the Amazon Web Services suite (S3, EC2, EBS, and many others), Microsoft Azure, Rackspace, and Google App Engine. Open source software for building clouds includes OpenStack (http://openstack.org), Eucalyptus (http://www.eucalyptus.com), and Nimbus (http://www.nimbusproject.org). Prominent cloud research efforts include NASA's Nebula and the NSF's FutureGrid. This elasticity allows users to outsource their computing infrastructure, growing or shrinking it as necessary. Cloud GIS is an emerging technology which further develops Internet GIS using a Service Oriented Architecture approach in a Cloud computing environment. Several desirable characteristics make Cloud GIS a useful framework to meet the growing needs of connecting earthquake research community and emergency responders.

- Data as a Service (DaaS): Earthquake research involves numerous types of spatial data, such as seismicity, GPS time series and optical images. Many of these are distributed in file formats which are not widely supported outside the geophysics community. It is necessary to integrate data from multiple sources and produce standards-compliant geospatial products through a DaaS. Also, DaaS must support the series of operations associated with the remote data, including projection support, format conversion, and data fusion.

- Software as a Service (SaaS): It is very common that earthquake researchers relay on in-house software packages to analyze the specific type of data. For example, QuakeSim project developed Daily Regularized Deterministic Annealing Hidden Markov Model (RDAHMM) packages [3] for analyzing GPS daily time series data and the Simplex tool to find a dislocation fault model that best accounts for observed GPS and InSAR deformation data. SaaS not only hosts the applications accessible through Internet on the elastic cloud infrastructure, but also facilitates intuitive web interfaces for the broad range of end users if coupled with a well-designed service programming interface. Combined with DaaS, SaaS makes the outputs usable by downstream applications.
- Infrastructure as a Service (IaaS): Cloud computing is designed to supply elastic computational resources, hence a natural fit for the emergency response, since the infrastructure usage levels are very low on average but spike immediately after the earthquake events. With national or global-scale replication and content distribution, it is also possible with IaaS to provide access to application servers that provide better network access than centralized servers to distributed researchers and responders. Data providers could provide virtual images packed with the essential data and software that can be deployed automatically or instantiated by emergency responders on a Cloud once the earthquake happens.

In this article, we report our efforts to use Cloud GIS technology in QuakeSim and E-DECIDER, two related earthquake research projects, and demonstrate the process of building on-line tools with two case studies. We concentrate primarily on DaaS and SaaS considerations.

## QuakeSim and E-DECIDER Project

QuakeSim and E-DECIDER are NASA-funded earthquake research projects involving earthquake researchers and computer scientists from several research institutes. The goal of QuakeSim [4,5] (http://www.quakesim.org/) is to couple multiple observation sources with both forward and inverse modeling applications for investigating both individual earthquake events and complex interacting fault systems. QuakeSim data sources include GPS, seismicity, geometric fault models, Uninhabited Aerial Vehicle Synthetic Aperture Radar (UAVSAR) and Interferometry Synthetic Aperture Radar (InSAR) images. Application codes include dislocation models (Disloc and Simplex), full finite element fault simulations (GeoFEST), probabilistic forecasting tools (Virtual California), and time series data mining tools (RDAHMM).

E-DECIDER (Earthquake Data Enhanced Cyber-Infrastructure for Disaster Evaluation) (http://www.e-decider.org/) is a downstream project that evolved from QuakeSim. It is a bridging effort to provide decision support for earthquake disaster management and response utilizing NASA remote sensing and other available remote sensing data in conjunction with the modeling software developed in QuakeSim project and other sources. The overall goal of the project is to deliver these capabilities as standards-compliant GIS data products through a web portal/web services infrastructure that will allow easy use by decision-makers.

## Cloud GIS Architecture

The design of Cloud GIS services aims to serve both data providers and end users through a series of web-services that are accessible through as many platforms as possible. Figure 1 shows a simplified architecture of Cloud GIS systems. The right-hand side consists of clients, and the left-hand side is the cloud deployment, in which data collections (DaaS) and applications (SaaS) are made available on cloud servers (IaaS). The two major components of the GIS Cloud (top left of Figure 1) are the implementations of various GIS core capabilities and the more specialized Web Service layer, which extends these for specific applications. Cloud GIS servers typically come as pre-configured virtual machines that can be deployed on both private and public Cloud-based IaaS, and each virtual machine can be configured to run only a certain type of service according to user demands.

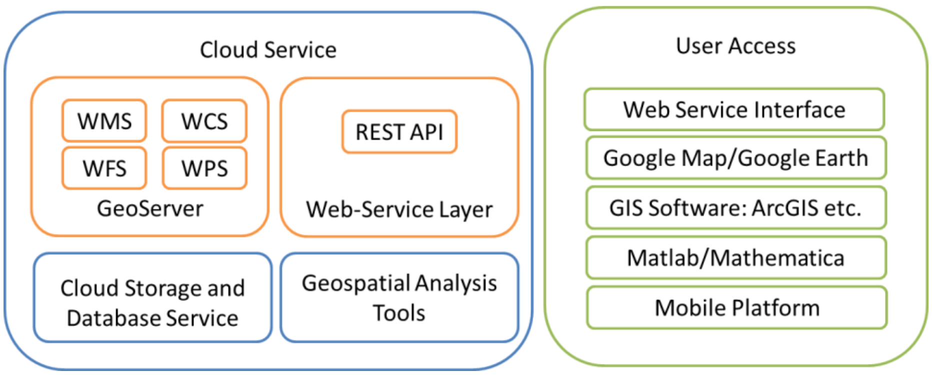


**Figure 1 Cloud-GIS Architecture**

In our evaluations of Cloud GIS, we chose GeoServer (http://geoserver.org) to provide core GIS capabilities of Figure 1. GeoServer is a community-maintained open source GIS server that allows users to share and edit geospatial data. It publishes data from any major spatial data source using the Open Geospatial Consortium (OGC) standards (http://www.opengeospatial.org/standards). These include the following.

- Web Map Service (WMS) is a standard for generating maps on the web for both vector and raster data, and rendering images in a number of possible formats: JPEG, PNG, GeoTIFF, GeoRSS, KML, and KMZ. WMS's ability (and GeoServer's implementation) of these multiple output formats are very useful for building interactive user interfaces in our case studies (described below).
- Web Coverage Service (WCS) provides a standard interface for requesting the raster source (raw images) and information over Internet. One of WCS's most important capabilities is data subsetting, which is particularly useful for downstream applications that need to manipulate large imagery catalogs.
- Web Feature Service (WFS) is the interface for vector data sources that include plotting definitions for features, such as the geometrical definitions of earthquake faults, and extensions that can capture non-plotting metadata, such as earthquake slip rates and source descriptions.

- Web Processing Service (WPS) provides rules for standardizing inputs and outputs (requests and responses) for geospatial processing services. It is an efficient way to turn GIS processing tools into SaaS for Cloud environment.

All of these basic OGC services expose network accessible request/response programming interfaces. However, it is usually useful to extend these generic interfaces. The Web Service layer provides easy-to-use network service (typically REST-based) API [6] for various specialized tasks. It can be not only used in web applications but also integrated into standalone applications that enable users to use the processing power in Cloud GIS server. It includes functionalities in the following categories:

- GIS Protocol Reflector API: GIS service request URLs (which must be constructed by clients in REST invocations) can be quite long and cumbersome. A reflector provides a much simpler URL call format. It also further enhances GeoServer to automatically adopt the configurations for the different user platforms. For example, the URL to call a WMS to deliver KML that includes images in lower resolution format suitable for mobile platforms is http://server/wms?layer=layername&format=kml&target=mobile.
- Server-Service API: this service API allows administrators to programmatically configure and manage the data and services on Cloud-GIS server.
- Geoprocessing API: provides web interface for common GIS functionalities and other standalone analysis tools. It also handles the input/output of in-house applications that require specific data formats. It enables users to run a set of geoprocessing tools, including some computing intensive applications, in Cloud environment.

**Cloud Testbed:** We have initially developed our Cloud GIS services on Indiana University's Intelligent Infrastructure, which is based on VMWare's vSphere software. Our prototype consists of a single virtual machine running 64-bit Red Hat Enterprise Linux Server (RHEL5) with 8GB RAM and 1TB disk space. It hosts the GeoServer and all necessary data for Cloud GIS tools. While vSphere provides basic VM hosting, it is not as "elastic" as, for instance, Amazon's Elastic Computing Cloud. We discuss the implications of this in our conclusions.

## Cloud GIS tool for UAVSAR Analysis

UAVSAR is a NASA project to use an uninhabited aerial vehicle (UAV) equipped with the synthetic aperture radar (SAR) system for rapid repeat-pass interferometry measurements of Earth's surface. The UAVSAR data portal (http://uavsar.jpl.nasa.gov/) distributes the SAR image products in the single-band binary files; the size of one single image ranges from several hundred megabytes to several gigabytes. It also supplies pre-rendered images stored as KML or KMZ, which can be visualized in Google Earth and Google Map. The disadvantage of this file-based distribution system is that users have to download the complete raw data and use special software capable of handling the UAVSAR format to analyze it. Without specific domain knowledge on SAR images, it is difficult for general users to extract useful information for emergency response.

We have developed Cloud GIS services to automatically scan the metadata and import the SAR images into Cloud storages. The images, after conversion to GeoTIFF, are distributed through WMS service;

Figure 2 shows several examples. We also provide WCS services that allow users to access the raw data in the interested study area at user-specified spatial resolutions. UAVSAR ground-projected products are in an equiangular geographic projection corresponding to 6 m x 6 m pixels; it is very common that users are only interested in the partial image at a much lower spatial resolution such as 30m x 30m rather than the whole original image. Through the on-demand WCS protocol, we can greatly reduce the burden of downloading large images to desktops and to remote servers in distributed processing workflows and pipelines.

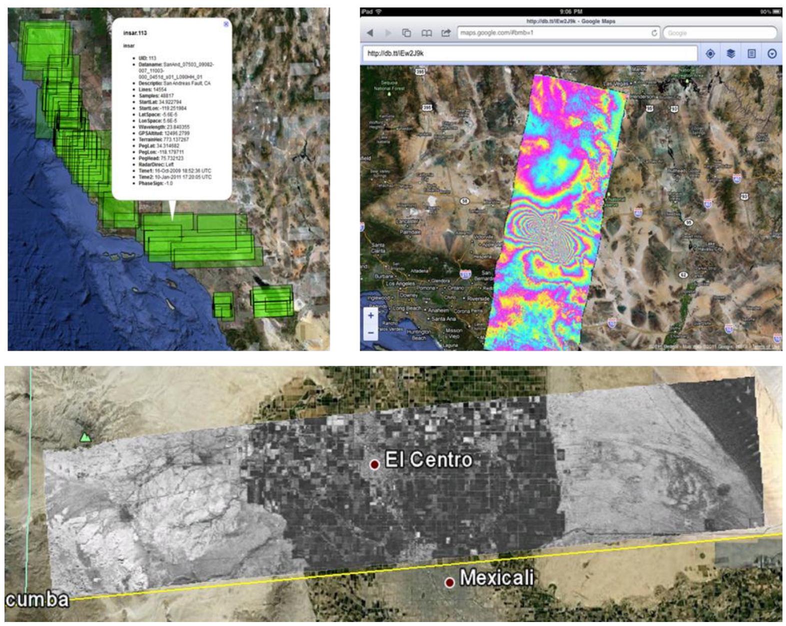


**Figure 2 UAVSAR Distribution example, Image Metadata (top left), Image on mobile platform (top right), Google Earth (bottom)**

In support of QuakeSim and E-DECIDER, we have built several on-line tools to help general users to analyze the UVASAR data directly through the web browser. We use the Line-Of-Sight (LOS) profile tool and the Simplex surface deformation inversion tool to demonstrate the structure of web applications based on Cloud GIS services. These examples show how powerful but complicated GIS capabilities can be wrapped with simpler, application-specific APIs that can be used to build mash-ups and other third-party applications.

**LOS profile tool** is used to calculate the cross-section of Line-Of-Sight displacement in a SAR interferogram. We implement the required process to extract the LOS values from a selected SAR image as an add-on REST service co-deployed with GeoServer. The user interface (Figure 3) is implemented on a separate server using Google Map and JQuery JavaScript libraries.

In the first step, the user is presented with a map with thumbnails of all available data sets. The user clicks the region of interest, and the server returns the list of images available in this area. Usually there are multiple overlapping images taken at different time periods, so we allow the user to refine the selection by presenting a table with all available images that contain the selected point (top right-hand side of Figure 3). The web service API for this operation has the form http://server/imagequery?location=lon1,lat1. Internally this service uses the WMS GetFeatureInfo function to query the vector layer that contains the bounding box and metadata of the InSAR images. Compared with the general but cumbersome WMS call URL, the web service API supplies a much simpler and cleaner interface to the web developer.

In the second step (lower half of Figure 3), the user is presented with a low-resolution version of the selected InSAR image. The user clicks the map and is presented with draggable starting and ending points. The user drags the points on top of the selected image and is presented with interactive plots that show the value of LOS displacement and corresponding Digital Elevation Model (DEM) data along the cross section. These values are extracted from the high resolution data; the low resolution image shown in the Web interface is for presentation only. The service API has the form http://server/profiletool?image=image_id&points=lon1,lat1,lon2,lat2. Internally, this wrapper service queries two images at the same time: one for the LOS calculation and the other for DEM. It generates a series of locations along the cross section, with the spatial sampling resolution decided by the balance between the plot quality and the length of the cross section. The profile tool service then calls the WMS GetFeatureInfo function for each location with the two images. LOS is calculated based on the query results from one image.

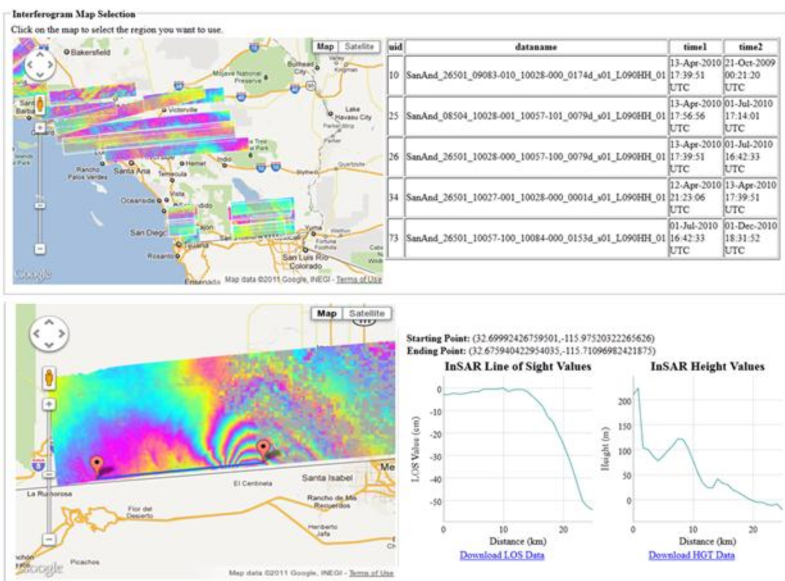


**Figure 3 Screenshot of LOS profile tool: selection of images (step 1, top), LOS profile (step 2, bottom)**

**Simplex** is a command line tool developed by collaborators at NASA JPL that optimally finds a dislocation model of fault slip that accounts for observed GPS and InSAR deformation data. In this case, we need to provide a separately running Simplex service (not co-located with our GIS services) with a subset of the observational data in a selected region of interest. It is possible to transfer the entire image file or files, but this is not optimal since Simplex only needs to be applied to a subset of the data (such as the region of interest showing displacement fringe patterns associated with an earthquake). Figure 4 shows the structure of the on-line simplex tool. The user examines the high resolution interferogram image on Google Map through GeoServer WMS protocol. The polygon coordinates *n* of the user-selected region are sent in as the parameter to the web service, http://server/simplex?image=image_id&polygon=lon1,lat1|lon2,lat2...|lonn,latn. The service is implemented as a python wrapper that pulls the data in the selected polygon region through the WCS protocol and generates the metadata file required by the command line simplex binary; it also reformats the outputs for the plotting service. The plotting service produces a KML file that contains the Simplex plot and sends the result back to the web interface. Users can also programmatically call the Simplex Web Service API directly to process multiple studies on Cloud GIS server and avoid downloading the data and running the simplex tools on the local computing resource. The Simplex tool itself is also under continuing development as new features are added. By providing Simplex as a service, we can ensure that the user has the latest version. Version information is also part of the service's standard output.

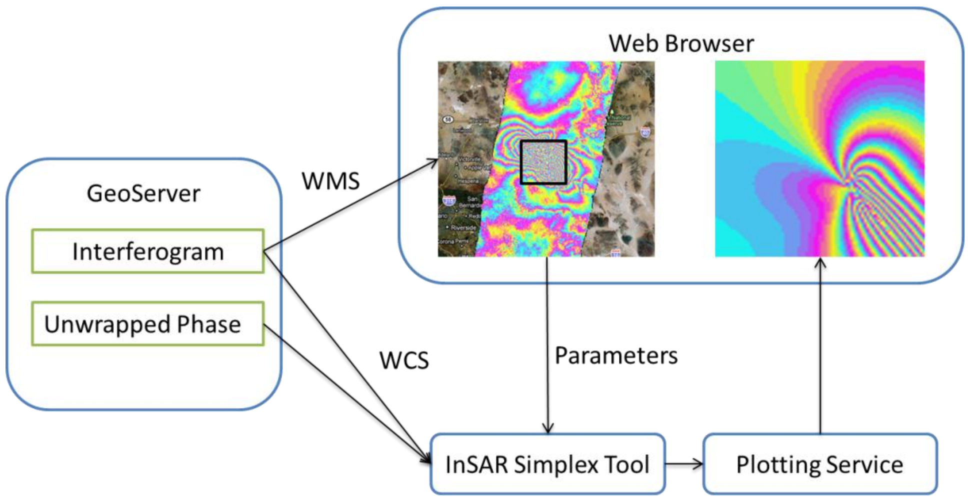


**Figure 4 the Structure of Simplex Tool**

## GIS Services for the E-DECIDER Project

The NASA-funded E-DECIDER project bridges the earthquake research and emergency response communities. The data products and tools from QuakeSim project are integrated with a broader range of services and workflows used by emergency responders. The Cloud GIS system for the QuakeSim project provides the infrastructure services for the E-DECIDER project. Geophysical modeling tools and results of earthquake forecasting tools from QuakeSim project along with remote sensing data are accessible through Web Service APIs. One of the services is HAZUS gadget, which allows users to generate scenario earthquakes for FEMA (Federal Emergency Management Agency) HAZUS based on the forecasting results from QuakeSim. FEMA HAZUS (http://www.fema.gov/plan/prevent/hazus/) is a nationally applicable standardized methodology that contains models for estimating potential losses from earthquakes, floods, and hurricanes. Emergency planners identify the interesting region with the information from the forecasted hot-spot areas, which are identified by the global forecast of future earthquake activity service from the Open Hazards Group (http://www.openhazards.com/). Fault model parameters are currently determined using a simple heuristic based on the magnitude of the earthquake event, but we can also obtain fault models through QuakeSim's QuakeTables service. Then the Web Service uses the OpenSHA framework [7] (http://www.opensha.org/) to generate the HAZUS input files for the scenario earthquake, which can be used for earthquake damage estimations. The simulation results can be shown on Google Map with the UAVSAR data to identify areas where the greatest deformation and damage has occurred and emergency services may need to be focused. Figure 5 shows the example of HAZUS gadget. Besides delivering the products both as web services through the E-DECIDER portal, we plan to further integrate Cloud GIS system with the FEMA Unified Incident Command and Decision Support (UICDS) framework. UICDS (http://www.uicds.us/) is information sharing middleware for FEMA NIMS (National Incident Management System) incident management that continuously receives and shares standardized data among many agencies during an incident.

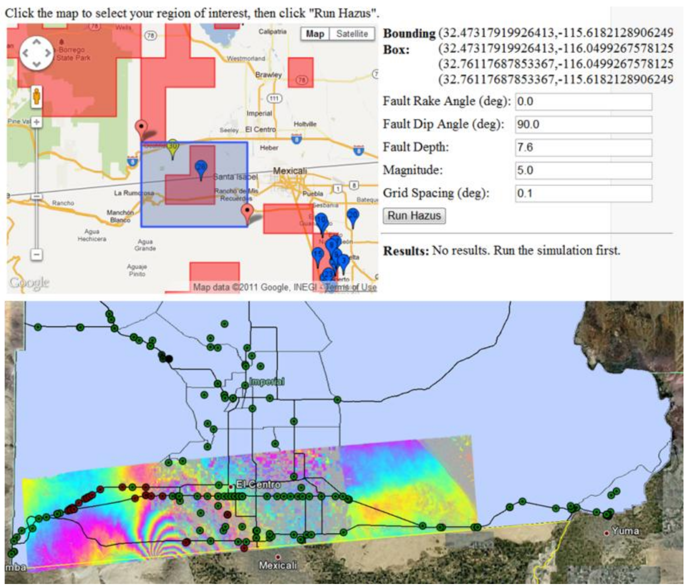


**Figure 5 HAZUS gadget, hot spot forecast and fault model (top) and HAZUS simulation result (bottom, red dots indicate damaged bridges)**

## Conclusions and Future Work

This paper presents the first steps for building the necessary infrastructure for the growing needs of collaboration efforts for earthquake research and disaster response. We use Cloud GIS technology to build the platform for earthquake researchers to explore large amount of data (particularly GeoTIFF-encoded SAR imagery) efficiently. It also gives researchers the tools to deliver their knowledge in a timely way to emergency planners and responders.  In this paper we have presented our first steps in deploying Cloud-based GIS services.  We plan to build on this to more fully exploit the elasticity and other capabilities of commercial clouds. We outline some of this future work below.

Even though Computing Clouds can meet the storage requirement during the peak data usage of a large earthquake event, it may not be realistic or necessary to collect the data from various data providers into one place. One solution is to distribute pre-configured virtual machines that data providers can host in their own private cloud. Distributed geospatial data can be collected and managed by catalog applications, such as GeoNetwork (http://geonetwork-opensource.org/). GeoSever supports loading data from remote WFS and WMS servers; the data pulled from the remote server can be cascaded through GeoServers and web services. It is also a straightforward way to exchange the data among data providers.

The computing resource, network bandwidth and latency requirements of QuakeSim and E-DECIDER tools as well as data products after large earthquake events are uncertain. Developing countries have much less data, especially the pre-earthquake data, when compared to developed countries, as we have seen in Haiti 2010 and Japan 2011 earthquakes. There are various techniques to scale up the GIS services on cloud computing platform. At the virtual machine level, one simple way is to run multiple VM instances that mount the same storage, each server has the same configuration and access the same data set.

Our prototyping is based on VM hosting at Indiana University using vSphere. This provides basic IaaS capabilities such as image hosting, but it lacks the advantages of very large, distributed commercial clouds. In particular, we have seen that client-to-service network performance is an important issue if we wish to provide interactive Web tools. Although Indiana University has excellent networking, it is not capable of providing wide-area or global content distribution networks and dynamic routing to the best available service instances; instead, all requests go to the same server or server cluster. A more interesting scenario that can be supported by commercial clouds is elasticity to meet increased demand from various research communities and agencies during large earthquake events. The huge amount of data gathered immediately after the earthquake is exceeding the computing resources for any independent organization. Commercial clouds have enormous computing resources to tackle this problem. A hybrid cloud model combing private and public cloud is a way to go. However, there are many complex issues related with hybrid clouds yet to be explored, such as system design, data security, and data sovereign/privacy; the last one is a particular sensitive issue as regards of geospatial data covering the sensitive areas/targets.

## Acknowledgments

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, Indiana University, University of Southern California, and University of California's Davis and Irvine campuses under contract with NASA. The work was sponsored by NASA's Advanced Information Technologies Program.

## References

1. F. Sullivan, "Cloud Computing for the Sciences," special issue, Computing in Science & Eng., vol. 11, no. 4, 2009, pp. 10–11.

2. L. Wang et al., "Cloud Computing: a Perspective Study," New Generation Computing, vol. 28, no. 2, pp. 137-146, Jun. 2010.

3. R. Granat, G. Aydin, M. Pierce, Z. Qi, and Y. Bock, "Analysis of streaming GPS measurements of surface displacement through a web services environment," in 2007 IEEE Symposium on Computational Intelligence and Data Mining, 2007, pp. 750-757.

4. A. Donnellan et al., “QuakeSim: Enabling Model Interactions in Solid Earth Science Sensor Webs,” in 2007 IEEE Aerospace Conference, 2007, pp. 1-8.

5. M. E. Pierce et al., “QuakeSim: Web Services, Portals, and Infrastructure for Geophysics,” in 2008 IEEE Aerospace Conference, 2008, pp. 1-7.A.

6. Roy T. Fielding, Richard N. Taylor: Principled design of the modern Web architecture. ACM Trans. Internet Techn. 2(2): 115-150 (2002)

7. E. H. Field, T. H. Jorda, and C. A. Cornell, “OpenSHA: A Developing Community-Modeling Environment for Seismic Hazard Analysis,” Seismological Research Letters, vol. 74, no. 4, pp. 406-419, 2003.