International Journal of
Geo-Information

Article

Cloud-Based Architectures for Auto-Scalable Web Geoportals towards the Cloudification of the GeoVITe Swiss Academic Geoportal

Ionuț Iosifescu-Enescu *, Claudia Matthys, Charalampos Gkonos, Cristina M. Iosifescu-Enescu and Lorenz Hurni

Institute of Cartography and Geoinformation, ETH Zurich, Zurich CH-8093, Switzerland; cmatthys@ethz.ch (C.M.); gkonosc@ethz.ch (C.G.); cristina.iosifescu@karto.baug.ethz.ch (C.M.I.E.); LHurni@ethz.ch (L.H.)

* Correspondence: iosifescu@ethz.ch; Tel.: +41-44-633-30-28

Academic Editors: Bert Veenendaal, Maria Antonia Brovelli, Serena Coetzee, Peter Mooney and Wolfgang Kainz
Received: 30 May 2017; Accepted: 22 June 2017; Published: 25 June 2017

Abstract: Cloud computing has redefined the way in which Spatial Data Infrastructures (SDI) and Web geoportals are designed, managed, and maintained. The cloudification of a geoportal represents the migration of a full-stack geoportal application to an internet-based private or public cloud. This work introduces two generic and open cloud-based architectures for auto-scalable Web geoportals, illustrated with the use case of the cloudification efforts of the Swiss academic geoportal GeoVITe. The presented cloud-based architectural designs for auto-scalable Web geoportals consider the most important functional and non-functional requirements and are adapted to both public and private clouds. The availability of such generic cloud-based architectures advances the cloudification of academic SDIs and geoportals.

Keywords: cloudification; cloud-based geoportal architecture; functional requirements; non-functional requirements; scalability; GeoVITe; academic SDI

1. Introduction

The availability of information technology (IT) infrastructure in the form of Web services, a feature which is nowadays generally known as cloud computing [1,2], has transformed the way in which Spatial Data Infrastructures (SDI) and Web geoportals are designed, implemented, deployed, administered, and maintained [3].

The interest of the academic community in harnessing and shaping cloud computing for geospatial sciences started more than half a decade ago [2]. Cloud computing significantly redefined the possibilities of Digital Earth applications [3], and thus geoportals were one of the first geospatial applications that benefitted from the availability of cloud computing for geospatial sciences. This is due to the five essential characteristics of cloud computing: on-demand self-service, rapid elasticity, broad network access, resource pooling, and measured service limiting the upfront costs of traditional computing infrastructures [1–5].

An early example of the adoption of cloud computing for geoportals and SDIs is provided by the Swiss Federal Office of Topography, swisstopo. In 2008, only two years after the public launch of the Amazon Web Services (AWS) [6], swisstopo used AWS in order to meet the needs of one of its key customers for a Web portal. Later, it even migrated considerable parts of the Swiss Federal Spatial Data Infrastructure (FSDI) to the AWS cloud. The main reasons listed by swisstopo for the cloudification of their Web portal are the performance and capacity problems encountered with on-premise infrastructures, while the main benefits are the shorter time for allocating new servers due to elastic resources and the simplification of the server infrastructure through standardization and automation [7,8].

Approaches for integrating self-describing geoprocessing packages into elastic computational environments on the Web were on the research agenda of geographic information science even before the advent of cloud computing [9]. However, the wide availability of cloud computing supported the emergence of Geographic Information Systems (GIS) as a service [10]. Cloud-based GIS such as ArcGIS online have also greatly multiplied the use of maps in the context of citizen science by fostering important global trends such as geo-awareness, geo-enablement, or storytelling with story maps [11]. Currently, cloud computing is used to address big geospatial data challenges [4,5,12–14] such as time-constraint geodata analysis [15] or high-performance image processing for geospatial applications [16], and evolves towards the self-management of heterogeneous computing resources [17], bringing us closer to the longstanding vision of autonomic computing [18].

The feasibility of cloud computing for supporting geoportals is unquestioned and has been thoroughly researched in previous literature [3]. The benefits of elasticity and scalability for geospatial processes such as spatial queries [15] or image processing [16] have also been proven. However, the cloudification of existing geoportals, namely their migration into a cloud computing environment, has not yet been properly addressed. The analysis of the most important functional and non-functional requirements for geoportals is also not available in the context of cloud computing. Open and generic cloud-based geoportal architectures are currently not available and therefore, there exists no useful information for migrating a full-stack geoportal application to a cloud computing environment. This work fills this gap by openly publishing generic cloud-based architectures, adapted to both public and private clouds, which can support the cloudification efforts of existing geoportals.

As a consequence, this paper focuses on open and generic cloud-based architectures for auto-scalable Web geoportals. In Section 2, we review relevant geoportal architectural aspects, based on an existing geoportal use case. This includes functional requirements (FRs) and non-functional requirements (NFRs), as well as a non-cloud, three-tier geoportal architecture. Section 3 introduces two cloud-based architectural designs for auto-scalable Web geoportals, with different levels of technical difficulty and resulting geoportal quality. Section 4 concludes the work with discussions related to the hidden costs of cloudification, and reviews the most notable advantages of cloud-based, auto-scalable Web geoportals, which are capable of fulfilling important NFRs such as reliability and security.

2. Multi-Perspective Review of Architecturally-Relevant Aspects for Geoportals

2.1. The Use Case of the Swiss Academic Geoportal GeoVITe

The inspiration for researching generic and open cloud-based geoportal architectures stems from the cloudification efforts of an existing geoportal, namely the "Geodata Versatile Information Transfer environment" (GeoVITe). GeoVITe is the geoportal of the corporate academic spatial data infrastructure of ETH Zurich, currently in the process of becoming the national academic geoportal for data visualization and download in Switzerland as part of a comprehensive national service offered by geodata4edu.ch [19,20]. The national portal for geodata in teaching and research, geodata4edu.ch, was established within the scope of a collaborative project between the ETH Library, the Institute of Cartography and Geoinformation of ETH Zurich, and the University of Applied Sciences Rapperswil (HSR) [19].

The comprehensive geodata4edu.ch Swiss academic service represents a recent addition to the academic SDI landscape. The main goal of an academic SDI is to facilitate the availability of, the access to, and the efficient use of spatial information for academia [21]. This definition is in line with existing SDI definitions provided by the Global Spatial Data Infrastructure (GSDI) community [22] and swisstopo [23]. As a consequence, geodata4edu.ch shares a similar raison d'être, and it was inspired by established academic SDIs such as EDINA (Edinburg University Data Library) with its Digimap Web mapping service. EDINA was established for the dissemination of spatial information to British academia through the provision and authenticated access to standardized services, data, and knowledge and support [20,24]. Similarly, the motivation behind geodata4edu.ch was to meet the demand for geodata in various academic disciplines of Swiss academia. Its aim was to establish a comprehensive and efficient national service for the targeted location, access, presentation, downloading, and processing of geodata for research and teaching at Swiss universities [19]. This goal was achieved by building the national service based on the existing SDIs from ETH Zurich and HSR and by providing three interconnected components: (1) the project Webpage centered around a metadata search portal, (2) geospatial Web services for GIS professionals, and (3) the GeoVITe geoportal that provides the geodata download service for all members of participating Swiss universities [19].

Due to their significant number of potential users—staff and students—academic geoportals such as GeoVITe are prime candidates for cloudification. Considerations related to future system characteristics such as the performance, scalability, cost-effectiveness, or self-management and maintainability triggered the cloudification efforts of the Swiss academic geoportal GeoVITe and its associated academic SDIs. Therefore, GeoVITe represents a real use case for open and generic cloud architectures that can enable geoportals to take advantage of the cloud functionalities currently available. Based on the GeoVITe use case, we can first provide a multi-perspective review of architecturally-relevant aspects for geoportals, including FR, NFR, and traditional three-tier architectures.

2.2. Geoportal Functional Requirements

In academia, geodata are needed in a wide range of disciplines such as environmental management, architecture, urban and landscape planning, and medicine [25]. Unfortunately, until recently, the actual access to geodata has largely remained a cumbersome task for users not familiar with cartography or geographic information [26].

In this context, geoportal technologies enable a more intuitive and user-friendly access to geodata. The GeoVITe geoportal is providing the download services of the geodata4edu.ch national service. It aims to redefine the way in which researchers and students access geospatial information by providing a user-friendly Web interface [27] for instant access to spatial datasets based on the GeoAdmin [28] and OpenLayers [29] frameworks. Consequently, the guiding FRs are related to the access, visualization, and download of geospatial datasets. A user should be able to visually navigate the available data (spatially, thematically, and temporally), select the desired dataset and area of interest (AoI), and also be able to directly download the required data in a straightforward manner through a user interface accessible with a standard Web browser.

The fulfilment of these FRs determines the utility of the GeoVITe geoportal as perceived by its users. Thus, through the implemented Web browser interface of the GeoVITe geoportal (shown in Figure 1), users can indeed visually browse or search the needed spatial data, select the extent of the AoI, and immediately download the selected datasets.

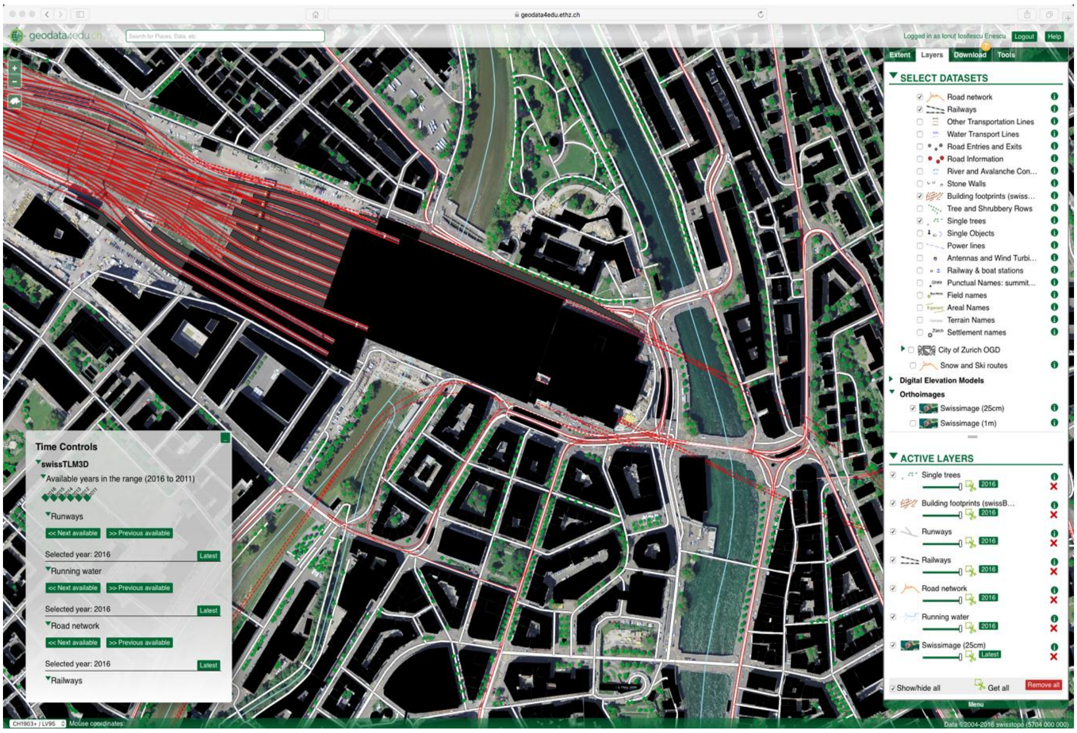

Figure 1. The GeoVITe geoportal provides a download service for the Swiss national portal for geodata in teaching and research—geodata4edu.ch; Geodata$^{©}$ 2016 swisstopo (JD100042).

2.3. Non-Functional Architectural Requirements and Constraints

NFRs play a key role in the architectural design of any information system, including geoportals and, as a consequence, NFRs have the power to drive key architectural decisions. In GeoVITe, we have considered several generic NFR categories mainly based on the taxonomies of Rashwan [30] and Roman [31]. More specifically, we refer to usability, reliability, security, maintainability, life-cycle, political, and economic NFRs. In this context, it is important to highlight the fact that NFRs are synonymous to architectural design constraints [31] and that they impact the quality of such information systems [32].

Cost-effectiveness is perhaps the strictest constraint in the life-cycle of an academic geoportal. Due to the inflexibility of budgets assigned for geoportal development in an academic environment, it is inherently directing the development towards a minimum viable product. Nevertheless, economic constraints can have dual outcomes. On the one hand, economic considerations are inherently constraining efforts focusing on addressing functional or other non-functional requirements. On the other hand, the strive for cost-effectiveness is definitely a factor that favors the swift adoption of public clouds such as AWS [6,8].

The perceived quality of a geoportal goes beyond its proven utility that is expressed by the fulfillment of its FRs. Usability, reliability, and security are equally critical factors influencing the quality of a geoportal, as perceived by its users. Among them, in our opinion, the most important NFR is usability. Although the economic factors are always of concern, focused usability studies with 10 to 40 participants have proven to be reasonably cost-effective [33], because they allow the implementation team to focus their limited resources on the most relevant features. As a consequence, we are the proponents of such usability studies on a regular basis, for example, every one to two years. We recommend the continuous improvement of a geoportal's user interface through a "wheel of design" approach that considers the interplay between usability, utility, and user feedback [33].

Reliability is the second important NFR that needs to be considered in the design of a geoportal. It ensures that the information system is performing well in a consistent manner. This NFR is generally quantified using the mean time between failures or the failure rate (number of failures in time) and has several associated facets, of which the most important are scalability, performance, and availability. Scalability is maybe the main challenge encountered by Web applications such as geoportals. A well-rounded geoportal architecture needs to take into account the increasing loads on the system, such as increases in the number of users and the associated increases in traffic and Web requests [34]. Therefore, a scalable geoportal architecture should allow the flexible upgrade and downgrade of serving infrastructure by planning for vertical, horizontal, or mixed scalability. In the case of vertical scaling, computing resources such as CPU cores or memory can be added to individual server nodes, thus increasing the performance of the existing nodes, while in horizontal scaling, more identical server nodes are provisioned, thus increasing the overall system performance [15,16].

The scalability of a geoportal directly influences its overall reliability. Under an increasing load, any non-scalable Web application will exhibit a degrading performance and possibly more or less extensive periods of unavailability. The main measure to ensure a reliable operational behavior for a geoportal is represented by overprovisioning computing resources. These spare resources can not only be used to ensure uninterrupted availability through failover on redundant servers, but they can also be used for scalability. Unfortunately, as the provisioning of spare resources is particularly expensive, the trade-off between scalability and cost-effectiveness needs to be carefully considered.

Security is another factor that is important to geoportal users. Information security has many components, and among them, confidentiality, data integrity, and data availability are critical. Confidentiality refers to the protection of information against disclosure, data integrity ensures data protection against corruption or even loss, and data availability ensures continuous access to information resources by protecting them against interference [34]. In this context, another aspect of information security is that data availability is also inherently linked to the reliability of an information system. This means that a geoportal may not only become unavailable due to an increasing load and failures as described above, but also due to malicious intent and hacking.

Finally, it is important to note that not all NFRs can be optimally addressed. As a consequence, key design decisions for geoportals require complex trade-offs, mainly among limiting economic factors such as cost-effectiveness, expected functionality (utility), and the most important NFRs such as usability, reliability, and security. This is visualized in the pyramid introduced in Figure 2, containing a proposed hierarchy of the most important constraints for academic geoportals, as synthetized from our experience with developing GeoVITe. The pyramid illustrates that cost-effectiveness (in red) is the largest constraint that may limit the achievable geoportal quality, while paying attention to the other constraints, in the order from top to bottom, will significantly increase the geoportal quality.

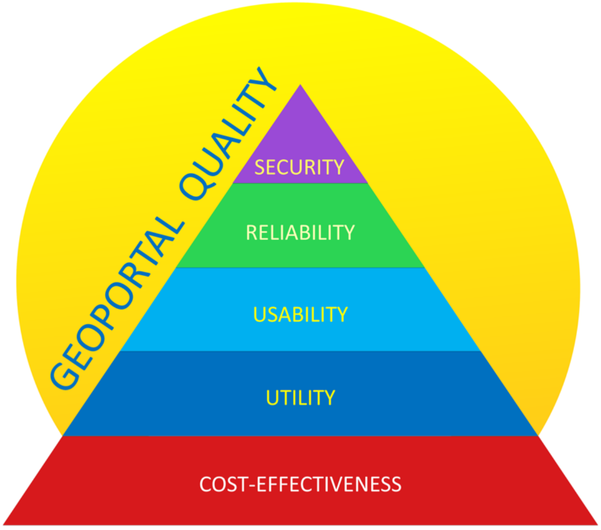

Figure 2. A hierarchy of the most important constraints for academic geoportals.

2.4. The GeoVITe Traditional Geoportal Architecture

The presentation of a traditional architectural design of a geoportal (non-cloud-based), as implemented in GeoVITe, lays the foundation needed for the systematic description of the generic cloud-based architectures presented later in this work. It also helps to demonstrate that the cloudification of a traditional geoportal can be a reasonably straightforward process.

GeoVITe is based on a three-tier architecture [27], with a clear separation among the data layer (implemented by back-end data servers), the application layer (consisting of server-side geo-services), and the presentation layer (front-end; client-side user interface), as illustrated in Figure 3.

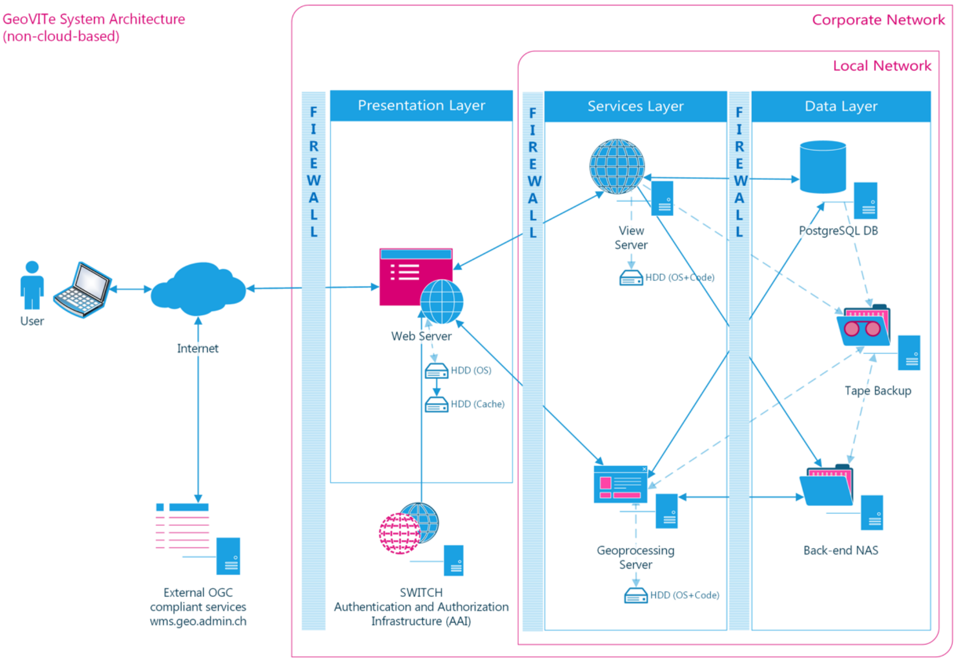

Figure 3. The traditional system architecture of the GeoVITe geoportal.

The data management layer of GeoVITe hosts available vector and raster datasets in back-end systems such as PostgreSQL geodatabases and Network Attached Storage (NAS) shares, where databases and folder structures are organized after a defined data management schema. In PostgreSQL geodatabases, each available vector product has its own dedicated database. Furthermore, for security reasons, the GeoVITe application logic accesses the geodatabases in a read-only mode. Due to similar security reasons, the data stored in NAS shares are accessible from the GeoVITe services layer exclusively through read-only credentials [27].

The server-side application layer was developed around servers hosting geoprocessing and view services based on well-known open source software and libraries such as QGIS Server, GDAL/OGR, and GeoTools. Furthermore, the application layer is extended by several Java EE technologies that create an Application Programming Interface (API) middleware around the application layer and handle API calls to the geoprocessing and view services [27].

Lastly, the GeoVITe Graphical User Interface (GUI) is built using well-known Web standards such as HTML, CSS, and JavaScript on top of the GeoAdmin and OpenLayers frameworks and organizes the available layers, products, and services in an object-oriented manner [27]. The entire GeoVITe GUI is served to the Web by an Apache Tomcat Web server, because the user interface is enclosed and enhanced by additional Java server technologies such as Java Server Pages (JSP) and Java servlets. These ensure proper communication with the services from the application layer, enable the integration of an external authentication and authorization mechanism, and enforce application security by sanitizing user requests and securely routing such requests to the underlying geo-services.

In this context, we note several considerations related to information security permeating the GeoVITe architecture. Although the implementation of extensive information security mechanisms is expensive, there exist certain cost-effective measures that can mitigate some of the security risks associated with allowing unrestricted access to resources. The use of an independent Authorization and Authentication Infrastructure (AAI), such as SWITCHaai [35] in the case of GeoVITe, alleviates the need for managing user names and passwords, thus protecting confidentiality. Another cost-effective way of protecting confidentiality is by enforcing Hypertext Transfer Protocol communications over Transport Layer Security (HTTPS) geoportal connections. Furthermore, data and computing resources can be protected in a cost-effective manner by individually firewalling and enforcing strict access rights between all architectural tiers of a geoportal, thus restricting the traffic that can enter and leave each architectural tier.

The GeoVITe geoportal is entirely service-driven. Users only have access to the Web-based GUI, which handles the majority of user interactions by sending requests and listening to responses from visualization and geoprocessing services. These services then access the corresponding data in the database, complete the necessary processing, and send the responses back to the user interface. The visualization services return vector and raster tiles in compressed GeoJSON and GeoTIFF formats, respectively. These tiles are generated on-demand and are then delivered to a tile cache hosted on the Web server; thus, subsequent identical visualization requests are served directly from the cache. The download services return the requested vector and raster geodata files in the extents, projections, and formats specified by the users. From a software engineering point of view, these background services are invisible to the users; they only see that the GUI reacts to their commands such as navigating through the different products available, zooming or panning the map, selecting the right area for download, and finally downloading the selected data, while they mostly remain unaware of the processes that are running on the back-end servers.

In this chapter, we have shown that the functional and non-functional requirements of a geoportal directly influence its system architecture and its overall quality. In this context, the cloudification of existing geoportals is able to massively improve their reliability, scalability, availability, and security, and at the same time, their overall performance.

3. Generic Cloud-Based Architectures for Geoportals

3.1. Cloudification of Geoportals

Geoportals are prime candidates for successful cloudification due to the fact that many geoportals feature loosely-coupled, service-driven architectures similar to GeoVITe. The cloudification of a geoportal represents the migration of a full-stack geoportal application to an internet-based private or public cloud, which provides on demand access to a shared pool of configurable computing resources that can be rapidly provisioned.

The main benefits of cloudification, such as the on-demand self-service and rapid elasticity, are usually discussed in the context of the top public clouds mentioned by the Gartner's Magic Quadrant for Cloud Infrastructure as a Service Worldwide report [36], namely AWS [6], Microsoft Azure [37], and Google Cloud Platform [38]. Nonetheless, due to many reasons varying from a required compliance with local organizational policies [1] to the aversion of hidden costs of public cloud usage [39], private clouds are also becoming a trend in academia. In private clouds, the main benefits of cloudification remain valid due to the inherent management of the physical infrastructure resources through proprietary (such as provided by VMWare [40] or Nutanix [41]) or open source (such as OpenStack [42] or Apache CloudStack [43]) cloud management and hyper-converged platforms.

A cloud platform, irrespective of its type (public or private), provides managed cloud services such as compute, networking, and storage services [6,37,38,42]. The main enabling technology is the virtualization of physical computing resources. For example, virtual servers are managed and provisioned by hypervisors, either proprietary ones, such as VMware vSphere ESXi [44] or Microsoft

Hyper-V [45], or open source ones, such as Linux KVM [46]. As there is a rich offer of cloud services that may be confusing for geoportal decision makers, we are going to focus on a core selection of relevant cloud computing services. As a result, we introduce two generic cloud-based architectures for geoportals, with different levels of technical difficulty and resulting geoportal quality.

3.2. Essential Cloud-Based Architecture for Geoportals

An essential cloud-based architecture for geoportals demonstrates the basic cloudification of a traditional geoportal architecture. As shown in Figure 4, such an architecture is obtained in a straight-forward manner by migrating traditional server and storage solutions to essential compute, networking, and storage cloud services.

The essential compute services needed by a cloud-based architecture for geoportals are: (1) virtual servers and (2) autoscaling. The virtual servers are in fact virtual machines (VMs) provisioned by hypervisors that are able to run geoportal software. In addition to the on-demand provisioning of virtual servers, the autoscaling technology is paramount for fulfilling the reliability NFR of a geoportal. The reliable operational behavior of a geoportal can be achieved through elasticity and scalability [15,16], by provisioning additional virtual servers during high-load spikes and automatically decreasing the compute capacity when not needed, thus reducing costs.

The network service needed to enable scalability and make use of the autoscaling technology of clouds is represented by the availability of managed load balancers. Load balancers are highly reliable services provided by the cloud infrastructure that distribute the load over available virtual servers that can start and stop automatically due to autoscaling. They can also handle failover, by detecting and eliminating failed virtual servers from their member pool, which further improves the reliability and availability of cloud-based geoportals.

In addition to compute and network services, we need to carefully consider the impact of available storage services for the architecture of geoportals and SDIs, as they might manage very large geodatasets. It is not uncommon for a geoportal to offer access to geospatial dataset collections starting from tens or hundreds of terabytes, which possibly contain individual raster tiles of over one gigabyte, as well as complex vector datasets with tens of millions or even billions of features. In this context, essential cloud storage services for geoportals are: (1) virtual server disks, (2) shared file storage, (3) Database as a Service (DBaaS), and (4) archiving services that need to scale to the above-mentioned specifics of geodata storage.

Virtual server disks are basic storage services that allow virtual servers to function. For each individual server, they contain the operating system and additional geoportal software. However, geodata cannot be stored on such virtual server disks because virtual servers need to scale automatically. Storing large amounts of data on such disks will, in practice, hinder autoscaling, since copying large datasets on new virtual servers is not only slow, but it also consumes bandwidth unnecessarily. As a consequence, geodatasets need to be stored in scalable shared file storage that can be mounted to any virtual server with traditional file access protocols.

The scalability of the shared file storage is a crucial factor for supporting the autoscaling of compute resources. An appropriate shared file storage cloud service will automatically scale with the number of connected virtual servers, while non-scalable shared file storage will exhibit a degrading performance when an increasing number of file-sharing clients use it. Similar considerations are also valid when geoportals need to use spatial databases. Although it is possible to install a spatial database on virtual servers, the scaling of such spatial databases with an increasing load becomes problematic. Therefore, another essential cloud service is Database as a Service (DBaaS), and more specifically, DBaaS for PostgreSQL. Such managed database services will automatically scale with the number of connections and eliminate the need for patching or for performing other tedious database administration and maintenance tasks. Finally, the last service is archiving. It provides low-cost but also low-performance storage for backups or data rarely accessed, as for example, in the rare case of disaster recovery.

The above-mentioned compute, networking, and storage cloud services are available under different names from public and private cloud platforms. Table 1 provides an overview of the essential services made available by the top public clouds mentioned in the Gartner report for cloud infrastructure as a service [36]. More specifically, these are AWS, Microsoft Azure, and Google Cloud Platform. Furthermore, we describe a possibility, one of many, for making such services available in a private cloud setup that uses a mix of virtualization technologies from VMware and other cloud platforms such as OpenStack or Nutanix.

Table 1. Essential Cloud Services for Geoportals.

Cloud Service	AWS	Microsoft Azure	Google Cloud	Private Cloud
Virtual servers	EC2	Azure VMs	Compute Engine	VMware ESXi VMs
Autoscaling	AWS AutoScaling	Azure Machine Scale Sets	Managed Instance Groups	Senlin-ceilometer/Aodh
Load Balancers	Elastic Load Balancing	Load Balancer	Google Cloud Load Balancing	OpenStack LBaaS v2/HAProxy
Virtual Server Disks	Elastic Block Store	Azure Storage Disks	Persistent Disks	Virtual Hard Disks (VHDs)
Shared File Storage	Elastic File System	Azure File Storage	ZFS/Avere	Nutanix Acropolis File Services
PostgreSQL DBaaS	Amazon RDS for PostgreSQL	Azure Database for PostgreSQL	Cloud SQL for PostgreSQL	OpenStack Trove with PostgreSQL
Archiving	Amazon Glacier	Azure Cool Storage	Cloud Storage Nearline	Glance, Nova Backup, VHD clones

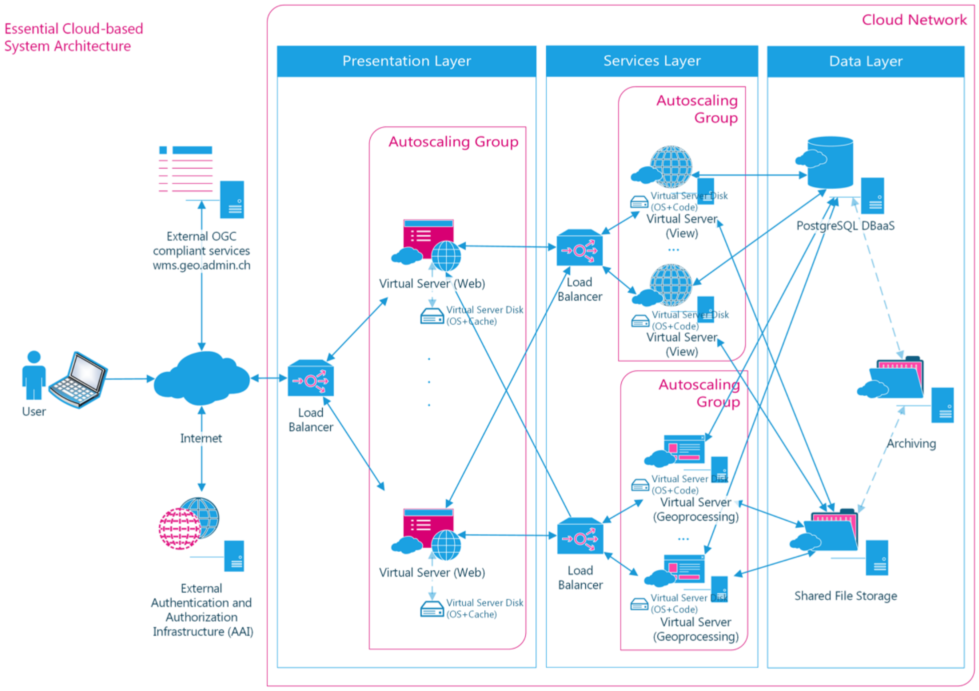
Essential Cloud-based System Architecture
Cloud Network
Presentation Layer
Services Layer
Data Layer
Autoscaling Group
Autoscaling Group
Autoscaling Group
External OGC compliant services wms.geo.admin.ch
User
Internet
External Authentication and Authorization Infrastructure (AAI)
Load Balancer
Virtual Server (Web)
Virtual Server Disk (OS+Cache)
Virtual Server (Web)
Virtual Server Disk (OS+Cache)
Load Balancer
Load Balancer
Virtual Server Disk (OS+Code)
Virtual Server (View)
Virtual Server Disk (OS+Code)
Virtual Server (View)
Virtual Server Disk (OS+Code)
Virtual Server (Geoprocessing)
Virtual Server Disk (OS+Code)
Virtual Server (Geoprocessing)
PostgreSQL DBaaS
Archiving
Shared File Storage

Figure 4. Essential cloud-based system architecture for geoportals.

The cloud-based architecture presented in Figure 4 can be easily implemented by existing geoportals. It features a low level of technical difficulty due to the straightforward replacement of the traditional computing components with their cloud-based counterparts. The services and the presentation layers of a geoportal are hosted on autoscalable groups of virtual servers, with the load being distributed among them by load balancers, while the associated software and information are stored on persistent virtual server disks. Furthermore, cloud storage services are able to successfully replace traditional solutions: PostgreSQL DBaaS can supersede individual PostgreSQL servers, scalable shared file storage replaces NAS shares, and archiving cloud services are able to substitute traditional tape back-ups.

3.3. Serverless Cloud-Based Architecture for Geoportals

We have seen in the essential cloud-based architecture that the geoportal migration from traditional on-premise hosting to a cloud environment can be straightforward. However, in addition to essential cloud services, there are several other relevant services, mainly available from public cloud providers, which are advantageous, but technically more demanding, for integration in a cloud-based geoportal architecture.

In a cloud-based architecture for geoportals, scalable and reliable application logic should be built following a serverless computing model. In this model, the server-side code can be scaled up without being bound to any virtual server or infrastructure, which means that the provisioning and administration of virtual servers becomes unnecessary. The serverless computing model is implemented in well-known public clouds such as AWS, Microsoft Azure, or Google Cloud Platform. In a private cloud setting, a purely serverless model is not yet available but can be approximated using application containers under a Container as a Service (CaaS) model. Light-weight and fast containers are designed to package and run dedicated applications with a high level of isolation because they are run in insolated user spaces on the same kernel of a hypervisor. Such application containers are also the products of advanced operating system virtualization.

Application logic that can be implemented from scratch, such as the geodata download application logic from GeoVITe, can be developed following a pure serverless model. Conversely, existing and complex software, such as the map visualization services from GeoVITe, can be packaged using the container technology. Moreover, in order to be effective, the serverless computing model needs to be implemented in conjunction with other scalable storage cloud services that can provide: (1) reliable message queuing, (2) object storage, and (3) caching. This is due to the fact that a serverless code is inherently limiting the input/output options for an application logic, because in general, a serverless code does not have access to any internal storage.

In a serverless geoportal architecture, with the possible exception of certain special use cases where virtual (Web) servers are still needed to serve Web clients, all application logic should be either containerized or migrated to a pure serverless compute model.

In a serverless geoportal architecture, input messages are retrieved from reliable message queues and the output can be stored in independent serverless storage such as object storage. Moreover, we recommend the introduction of performant cloud services specialized in providing serverless caching services for all operationally-critical resources. An overview of the cloud services recommended for a serverless cloud-based geoportal architecture is presented in Table 2.

Table 2. Additional Useful Cloud Services for Geoportals.

Cloud Service	AWS	Microsoft Azure	Google Cloud	Private Cloud
Containers (CaaS)	EC2 Container Service	Azure Container Service	Google Container Engine	OpenStack Magnum/Zun
Serverless Compute	Lambda	Azure Functions	Google Cloud Functions	Custom solutions [1,2]
Object Storage	Amazon S3	Azure Blob Storage	Google Cloud Storage	Swift;
Reliable Message Queuing	Amazon SQS	Azure Queue Storage	Google Cloud PUB/SUB	Zaqar
Caching	Amazon ElastiCache	Azure Redis Cache	Redis Cloud	Nutanix Acropolis [3]

[1] custom Murano application package for Java serverless computing using a managed Java cluster with high availability and autoscaling; [2] containers (e.g., Docker, Kubernetes) can also be used for building microservices in a private cloud setting; [3] Varnish or RAM drives can also be used caching in a private cloud setting.

Furthermore, when additionally considering security aspects, we highlight an important difference between using the network services of a public cloud and hosting a private cloud. A private cloud operates in a trusted network, while a geoportal running in a public cloud lacks the inherent protection of such a trusted network. On the one hand, as illustrated with the traditional architecture of the GeoVITe portal shown in Figure 3, the services and back-end data servers can be individually firewalled, managed, and protected in a local network. As a consequence, they perform their function in the local network without being connected to the internet and without being exposed to the associated security risks. Furthermore, the Web servers, which need to be located in the demilitarized zone of the corporate network in order to be accessible from the internet, can also be adequately firewalled in order to restrict access from the outside to HTTP traffic. This is equally true for any cloud-based geoportal architecture implemented in a private cloud. On the other hand, public clouds will assign public IP (internet protocol) addresses to compute instances, when not using more advanced network services. As a consequence, a secure serverless cloud-based architecture for geoportals should ideally include the following additional network and security cloud services: (1) virtual cloud networking, (2) a managed firewall, (3) security assessment, and (4) managed Distributed Denial of Service (DDoS) protection.

The virtual cloud network offers an isolated and secure managed network for running cloud resources such as virtual servers. Managed firewalls should be deployed before the load balancer of a Web-facing container or virtual server. Furthermore, an automated security assessment that assists to uncover commonly exploited application vulnerabilities is heavily recommended when using public clouds. Finally, the ability to repel Distributed Denial of Service (DoS) attacks would also be desirable for any academic geoportal hosted in a public cloud environment. An overview of these security-related services in the top public cloud platforms is provided in Table 3. In private cloud platforms, although a Firewall-as-a-Service (FWaaS) is, for example, available in OpenStack, traditional network security practices are still applicable.

Table 3. Network and Security Public Cloud Services for Secure Geoportals.

Public Cloud Service	AWS	Microsoft Azure	Google Cloud
Virtual Cloud Networking	Amazon Virtual Private Cloud (VPC)	Azure Virtual Network	Google Virtual Private Cloud (VPC)
Managed Firewall	AWS Web Application Firewall (WAF)	Application Gateway WAF	VPC Firewall, Brocade Virtual WAF [1]
Security Assessment	Amazon Inspector	Security Center	Google Cloud Security Scanner
Managed DDoS	AWS Shield	N/A [1,2]	N/A [1]

[1] not available from the respective cloud provider, but available from third-parties; [2] some comparable functionality configurable in Azure App Service with "dynamicIpSecurity/denyByRequestRate".

Figure 5 introduces a secure serverless cloud-based geoportal architecture that is defined by the serverless computing model. It features a high level of technical difficulty due to the replacement of traditional computing components with serverless compute (CaaS, serverless compute) and storage (object storage, reliable message queuing, caching) cloud services. As visible in Figure 5, the presentation layer of the geoportal is handled by an autoscalable group of Web containers using serverless caching services, with the load being distributed among them by managed load balancers. Similarly, all application logic, depending on its complexity level, is either containerized or implemented as pure serverless code with the additional support of reliable message queuing for communication. Furthermore, due to the fact that serverless computing resources lack additional information storage on persistent virtual server disks, the serverless cloud-based architecture will rely even more on managed serverless storage services such as the PostgreSQL DBaaS, scalable shared file storage, and object storage.

The list of the cloud services used by the proposed geoportal architectures introduced in this chapter is not exhaustive, but rather limited to the ones that are relevant for the core implementation of a geoportal. There are many additional useful services, such as services dedicated to the management of cloud resources (e.g., building, deploying, operating), the monitoring of cloud resources, analytics services, performance management services, billing-related services (e.g., monitoring, forecasting, reporting), and many others. Although highly useful, from a technical perspective, they are only nice in terms of providing supplements for the cloud-based geoportal architectures and as a consequence, there is no need further address them in this work.

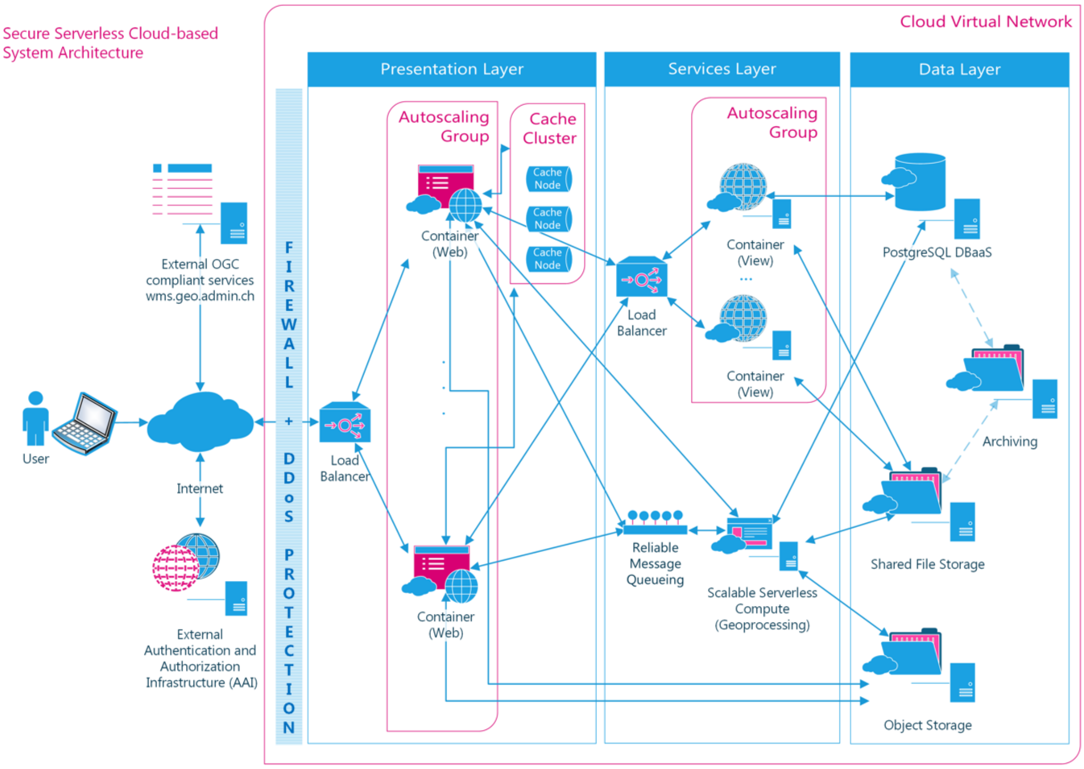

Figure 5. Secure cloud-based geoportal architecture using a serverless computing model.

4. Discussion

4.1. Security Concerns and the Use of Private Clouds

Security is an NFR with a profound impact on a Web geoportal, due to the fact that malicious attacks can severely affect the availability and reliability of any information system that ignores basic security practices. Fortunately, identity and access management (IAM), encryption, key management service (KMS), and even security assessments are available by default in the top public clouds. Furthermore, public cloud platforms employ dedicated personnel tasked with ensuring the security of the cloud and as a consequence, such public clouds are able to offer, by default, a higher level of security monitoring compared to what is commonly available in a private cloud setting. Nevertheless, private clouds can be managed and operated due to reasons such as data security, risk management, or compliance with local organizational policies [1]. As a consequence, private clouds can be encountered in various academic institutions where the organizational policies still prohibit the management of certain data in public clouds. Nonetheless, the main benefits of cloudification for geoportals, i.e., the on-demand self-service and rapid elasticity, remain valid even when deployed in private clouds due to the inherent management and automation of the physical infrastructure resources.

4.2. Discussion of Cost-Effectiveness in Cloud-Based Architectures for Geoportals

The cloudification efforts of GeoVITe are also relevant for the entire academic community dealing with academic SDIs and geoportals because the cloud-based architectures were designed for cost-effectiveness. First, the on-demand self-service enables technical personnel to allocate the needed computing resources for scalable geoportals on-demand, thus eliminating the lengthy traditional process of resource provisioning. As a consequence, it allows the IT administrators to focus on their core task of maintaining, instead of provisioning, computing resources. Second, the rapid elasticity of cloud computing allows geoportals to scale rapidly with the user load on the platform by adapting, in most cases automatically, the compute resources used by the geoportal to the variations in the daily and weekly traffic patterns. This automatic adjustment to low and high traffic times positively influences the overall running costs by reducing them. Third, the use of public clouds limits the upfront costs of traditional computing infrastructures. As a consequence, due to economies of scale, the yearly hosting costs for an academic geoportal should be less expensive in a public cloud than in a private cloud.

In practice, however, the decision to migrate a traditional geoportal architecture to the cloud should be carefully analyzed, because there are two cost-related caveats that need to be taken into account: hidden costs and additional implementation efforts. Therefore, we recommend reviewing three main sources of hidden costs when planning the deployment of geoportals to public cloud infrastructures: data transfer charges, extensive backup, and changes in the planned capacity. Regarding the first hidden cost, even if data transfers into the public cloud are generally free, there may be significant outbound transfer charges. Furthermore, transferring tens of terabytes of data over the internet would take several weeks without the use of import services such as the AWS Snowball. Concerning the second hidden cost, backups may also take significant, and therefore expensive, storage space, when old full point-in-time snapshots of persistent virtual disks are not regularly deleted. Finally, for the third point, it is customary to reserve and pay compute resources in advance and over a fixed term (e.g., three years) in order to take advantage of significant discounts compared to on-demand pricing. However, if the requirements for the discounted virtual servers change during the contract term, not all of the discounted machines can be fully utilized until the end of the contract term.

Furthermore, additional implementation efforts might be required in order to incorporate certain cloud-specific services such as scalable serverless compute or message queues into existing geoportal implementations. These additional cloud migration costs are upfront costs, since application logic needs to be re-implemented using cloud-specific APIs before the architectural deployment in the cloud. Therefore, these costs need to be carefully estimated and budgeted, otherwise they will also increase the final migration bill, in the same way as hidden costs. Albeit such costs, the cloudification of geoportals brings considerable advantages related to the fulfillment of NFRs such as reliability, scalability, and security, thus achieving an overall significantly higher quality geoportal deployment.

5. Conclusions

Cloud computing has redefined the way in which SDIs and Web geoportals are designed, managed, and maintained. This work introduced two open and generic cloud-based architectures for geoportals, which are able to support geoportal providers in their cloudification efforts. These cloud-based architectures are illustrated with the use-case of the GeoVITe geoportal, demonstrating the applicability of the presented general architectures to real use cases.

Regarding the cloudification process, the presented cloud-based geoportal architectures have different levels of technical complexity. The former features a low level of technical difficulty due to the straightforward replacement of the traditional on premise servers with autoscalable groups of virtual servers, which are connected to persistent virtual server disks and selected cloud storage services. In this essential cloud-based architecture for geoportals, the virtual compute services replace architectural components such as Web servers, visualization and download servers, and even back-end systems such as databases in a straightforward manner. The latter, namely the serverless cloud-based geoportal architecture, supports the exclusive cloudification of traditional server-based computing to serverless compute and other managed cloud storage services. However, it also features a higher level of technical difficulty for the geoportal's migration, because it requires additional implementation effort to write the serverless code, containerize the services, and use other necessary cloud services such as reliable message queuing services or object storage in the application logic.

The presented cloud-based geoportal architectures also assist interested readers with some key design decisions. Considering a fixed geoportal functionality, the NFRs such as reliability, security, and cost-effectiveness, have a significant impact on the final geoportal quality. Therefore, when cost-effectives is a major concern, the essential cloud-based architecture should be considered because it covers the fundamental needs of autoscaling for a high-performing geoportal. The secure serverless cloud-based architecture is the highest quality option that should be considered when sufficient upfront funding is available for the migration to serverless code and cloud-specific APIs.

In both the cloud-based architectures presented in this paper, autoscaling and managed compute resources demonstrate some of the most important technical benefits of cloud platforms. On the one hand, the autoscaling enables a geoportal to react automatically to changes in traffic and usage, thus making the manual administration of geoportal infrastructures superfluous. Horizontal scaling, such as that normally provided by cloud platforms with autoscaling groups, is also more versatile than vertical scaling, while unhealthy compute instances can be automatically detected and replaced. On the other hand, managed compute resources further reduce the administration requirements for cloud-based geoportal infrastructures. Therefore, serverless compute, managed databases, and scalable storage are the most important components that need to be included in any high-quality cloud-based geoportal architecture.

Acknowledgments: This work was partly supported by swissuniversities SUC-P5 (geodata4edu.ch). ETH Zurich covered the costs to publish in open access. The authors are grateful to Magnus Heitzler, Nadia H. Panchaud, Angeliki Tsorlini, and the three anonymous reviewers for their constructive comments and suggestions.

Author Contributions: Ionuț Iosifescu-Enescu designed the generic cloud-based architectures and led the writing of the paper with inputs from all authors; Claudia Matthys provided inputs regarding virtualized and hyper-converged server infrastructures; Charalampos Gkonos first suggested opening some work in progress towards the cloudification of GeoVITe to the scientific community through journal publications; Cristina M. Iosifescu-Enescu authored all figures except for Figures 1 and 2, which were authored by Ionuț Iosifescu-Enescu.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Mell, P.; Grance, T. The NIST definition of cloud computing. *United States Natl. Inst. Stand. Technol.* **2011**. [CrossRef]
2. Yang, C.; Goodchild, M.; Huang, Q.; Nebert, D.; Raskin, R.; Xu, Y.; Bambacus, M.; Fay, D. Spatial cloud computing: How can the geospatial sciences use and help shape cloud computing? *Int. J. Digit. Earth* **2011**, *4*, 305–329. [CrossRef]
3. Xia, J.; Yang, C.; Liu, K.; Gui, Z.; Li, Z.; Huang, Q.; Li, R. Adopting cloud computing to optimize spatial web portals for better performance to support Digital Earth and other global geospatial initiatives. *Int. J. Digit. Earth* **2015**, *8*, 451–475. [CrossRef]
4. Yang, C.; Yu, M.; Hu, F.; Jiang, Y.; Li, Y. Utilizing Cloud Computing to address big geospatial data challenges. *Comput. Environ. Urban Syst.* **2017**, *61*, 120–128. [CrossRef]
5. Li, Z.; Yang, C.; Liu, K.; Hu, F.; Jin, B. Automatic Scaling Hadoop in the Cloud for Efficient Process of Big Geospatial Data. *ISPRS Int. J. Geo Inf.* **2016**, *5*, 173. [CrossRef]
6. Amazon AWS. Available online: https://aws.amazon.com/about-aws/ (accessed on 15 May 2017).
7. Swiss Academy of Engineering Sciences. White Paper Cloud Computing. Available online: http://www.cloud-finder.ch/fileadmin/Dateien/PDF/News/2012-11-06_SATW_White_Paper_Cloud_Computing_EN_1_.pdf (accessed on 12 June 2017).
8. Amazon Web Services. Case Study: Swisstopo. Available online: https://aws.amazon.com/solutions/case-studies/swisstopo/ (accessed on 15 May 2017).

9. Müller, M.; Bernard, L.; Kadner, D. Moving code—Sharing geoprocessing logic on the Web. *ISPRS J. Photogramm. Remote Sens.* **2013**, *83*, 193–203. [CrossRef]
10. O'Doherty, P. Cloud Computing: Future Belongs to "GIS as a Service". *Geospatial World*. 2010. Available online: https://www.geospatialworld.net/article/cloud-computing-future-belongs-to-gis-as-a-service/ (accessed on 12 June 2017).
11. Kerski, J.J. Geo-awareness, Geo-enablement, Geotechnologies, Citizen Science, and Storytelling: Geography on the World Stage. *Geogr. Compass* **2015**, *9*, 14–26. [CrossRef]
12. Schnase, J.L.; Duffy, D.Q.; Tamkin, G.S.; Nadeau, D.; Thompson, J.H.; Grieg, C.M.; McInerney, M.A.; Webster, W.P. MERRA Analytic Services: Meeting the Big Data challenges of climate science through cloud-enabled Climate Analytics-as-a-Service. *Comput. Environ. Urban Syst.* **2017**, *61*, 198–211. [CrossRef]
13. Li, Z.; Yang, C.; Jin, B.; Yu, M.; Liu, K.; Sun, M.; Zhan, M. Enabling Big Geoscience Data Analytics with a Cloud-Based, MapReduce-Enabled and Service-Oriented Workflow Framework. *PLoS ONE* **2015**, *10*, e0116781. [CrossRef] [PubMed]
14. Big Data: Techniques and Technologies in Geoinformatics. Available online: https://www.crcpress.com/Big-Data-Techniques-and-Technologies-in-Geoinformatics/Karimi/p/book/9781138073197 (accessed on 21 June 2017).
15. Huang, W.; Zhang, W.; Zhang, D.; Meng, L. Elastic Spatial Query Processing in OpenStack Cloud Computing Environment for Time-Constraint Data Analysis. *ISPRS Int. J. Geo Inf.* **2017**, *6*, 84. [CrossRef]
16. Kang, S.; Lee, K. Auto-Scaling of Geo-Based Image Processing in an OpenStack Cloud Computing Environment. *Remote Sens.* **2016**, *8*, 662. [CrossRef]
17. Drăgan, I.; Fortiş, T.-F.; Iuhasz, G.; Neagul, M.; Petcu, D. Applying Self-* Principles in Heterogeneous Cloud Environments. Available online: https://link.springer.com/content/pdf/10.1007%2F978-3-319-54645-2.pdf (accessed on 21 June 2017).
18. Kephart, J.O.; Chess, D.M. The vision of autonomic computing. *Computer* **2003**, *36*, 41–50. [CrossRef]
19. geodata4edu.ch. *Project Info 2017*; geodata4edu.ch: Zurich, Switzerland, 2017; Available online: https://www.geodata4edu.ch/en/about-geodata4edu-ch/project-info/ (accessed on 21 June 2017).
20. Iosifescu Enescu, I.; Gkonos, C.; Iosifescu Enescu, C.M.; Tsorlini, A.; Hotea, M.D.; Piguet, A.; Hurni, L. Guidelines for a Comprehensive Design of Geoportals based on Open Geospatial Software. In Proceedings of the 28th International Cartographic Conference, Washington, DC, USA, 27–28 July 2017.
21. Iosifescu Enescu, I. *Maps for Spatial Data Infrastructures (Service-Oriented Web Mapping)*; Lecture Notes; Institute of Cartography and Geoinformation, ETH Zurich: Zurich, Switzerland, 2016.
22. Nebert, D.D. (Ed.) *Developing Spatial Data Infrastructures: The SDI Cookbook v. 2.0, Global Spatial Data Infrastructure (GSDI)*. 2004. Available online: http://gsdiassociation.org/images/publications/cookbooks/SDI_Cookbook_GSDI_2004_ver2.pdf (accessed on 15 May 2017).
23. Swisstopo Spatial Data Infrastructure. Available online: https://cms.geo.admin.ch/www.swisstopo.admin.ch/archives/cms2007/internet/swisstopo/en/home/topics/geodata_inf.html (accessed on 15 May 2017).
24. Smyth, C.G. SDI–national to global: Perspectives from the UK academic sector. In Proceedings of the 27th International Cartographic Conference: Spatial Data Infrastructures, Standards, Open Source and Open Data for Geospatial (SDI-Open 2015), Rio de Janeiro, Brazil, 20–21 August 2015.
25. geodata4edu.ch. *Fields of Application 2017*; geodata4edu.ch: Zurich, Switzerland, 2017; Available online: https://www.geodata4edu.ch/en/service/possible-fields-of-application/ (accessed on 21 June 2017).
26. Iosifescu Enescu, C.M.; Iosifescu Enescu, I.; Jenny, H.; Hurni, L. GeoVITe—A Service-Driven Solution for an on-Demand, User-Friendly Web Access to Geodata. In Proceedings of the 25th International Cartographic Conference, Paris, France, 3–8 July 2011.
27. geodata4edu.ethz.ch. *Geodata Download Service Technology Overview 2017*; geodata4edu.ch: Zurich, Switzerland, 2017; Available online: https://geodata4edu.ethz.ch/documents/GeoVITe_Technology_Overview.pdf (accessed on 21 June 2017).
28. GeoAdmin API Documentation. Available online: http://api3.geo.admin.ch/ (accessed on 15 May 2017).
29. OpenLayers. Available online: http://openlayers.org/ (accessed on 15 May 2017).
30. Rashwan, A.; Ormandjieva, O.; Witte, R. Ontology-based classification of non-functional requirements in software specifications: A new corpus and SVM-based classifier. In Proceedings of the 2013 IEEE 37th Annual Computer Software and Applications Conference, Kyoto, Japan, 22–26 July 2013; pp. 381–386.

31. Roman, G.-C. A taxonomy of current issues in requirements engineering. *IEEE Comput.* **1985**, *18*, 14–23. [CrossRef]
32. Iosifescu Enescu, I.; Vescoukis, V.; Iosifescu Enescu, C.M.; Müller, F.; Panchaud, N.H.; Hurni, L. Hypercube-Based Visualization Architecture for Web-Based Environmental Geospatial Information Systems. *Cartogr. J.* **2015**, *52*, 137–148. [CrossRef]
33. Kellenberger, B.; Iosifescu Enescu, I.; Nicola, R.; Iosifescu Enescu, C.M.; Panchaud, N.H.; Walt, R.; Hotea, M.; Piguet, A.; Hurni, L. The wheel of design: assessing and refining the usability of geoportals. *Int. J. Cartogr.* **2016**, *2*, 95–112. [CrossRef]
34. Coulouris, G.F.; Dollimore, J.; Kindberg, T. *Distributed Systems: Concepts and Design*, 4th ed.; Addison Wesley: Harlow, UK, 2005.
35. SWITCHaai. Available online: https://www.switch.ch/aai/ (accessed on 15 May 2017).
36. Gartner. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide. Available online: https://www.gartner.com/doc/reprints?id=1-2G2O5FC&ct=150519 (accessed on 15 May 2017).
37. Microsoft Azure: Cloud Computing Platform & Services. Available online: https://azure.microsoft.com/en-us/ (accessed on 15 May 2017).
38. Google Cloud Computing, Hosting Services & APIs | Google Cloud Platform. Available online: https://cloud.google.com/ (accessed on 15 May 2017).
39. CloudCheckr. Revealed: The 7 Hidden Costs Every Public Cloud User Needs to Avoid. Available online: http://cloudcheckr.com/2017/05/revealed-7-hidden-costs-every-public-cloud-user-needs-avoid/ (accessed on 15 May 2017).
40. VMware. VMware Hyper-Converged Infrastructure (HCI)—VMware Products. Available online: http://www.vmware.com/products/hyper-converged-infrastructure.html (accessed on 15 May 2017).
41. Nutanix. Hyperconverged Infrastructure: The Definitive Guide. Available online: https://www.nutanix.com/go/what-is-nutanix-hyperconverged-infrastructure.html (accessed on 15 May 2017).
42. OpenStack Open Source Cloud Computing Software. Available online: https://www.openstack.org/software/ (accessed on 15 May 2017).
43. Apache CloudStack: Open Source Cloud Computing. Available online: https://cloudstack.apache.org/ (accessed on 15 May 2017).
44. VMware vSphere ESXi Bare-Metal Hypervisor. Available online: https://www.vmware.com/products/esxi-and-esx/overview.html (accessed on 15 May 2017).
45. Microsoft Hyper-V, Server Virtualization. Available online: https://www.microsoft.com/en-us/cloud-platform/server-virtualization (accessed on 15 May 2017).
46. Linux Kernel Virtual Machine. Available online: http://www.linux-kvm.org/page/Main_Page (accessed on 15 May 2017).