Modern open-source, cloud-native, serverless web-mapping:

A Case Study for the Agriculture Industry

Last Revision: April 12, 2020

Amy Farley

afarley@austin.rr.com

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

# 

**Table of Contents**

[Overview 1](#_Toc37505693)

[Definitions 1](#_Toc37505694)

[Project Purpose 1](#_Toc37505695)

[Agriculture Industry Trends 6](#_Toc37505696)

[Existing Solutions 10](#_Toc37505697)

[Common components of a web-mapping solution 13](#_Toc37505698)

[Open-source technology 18](#_Toc37505699)

[Open-source GIS 19](#_Toc37505700)

[Cloud computing 23](#_Toc37505701)

[Cloud-native 24](#_Toc37505702)

[Serverless 25](#_Toc37505703)

[Existing Cloud GIS solutions 30](#_Toc37505704)

[Serverless GIS solutions 32](#_Toc37505705)

[Project Objective 33](#_Toc37505706)

[Proposed Methodology 35](#_Toc37505707)

[Technology 35](#_Toc37505708)

[Software Development Methodology 38](#_Toc37505709)

[Expected Challenges 38](#_Toc37505710)

[Anticipated Results 39](#_Toc37505711)

[Project Timeline 40](#_Toc37505712)

[Remaining Coursework 40](#_Toc37505713)

[Project Milestones 40](#_Toc37505714)

[Possible Presentation Venues 41](#_Toc37505715)

[References 43](#_Toc37505716)

# Overview

This project title is full of technical buzzwords, each of which is significant: 1) open-source, 2) cloud-native, 3) serverless, 4) web-mapping solutions, and 5) Agriculture Industry. The purpose of this project is not to complicate nor convolute a topic. Instead, it is to recognize a convergence of industry trends and technology advancements that can be combined to create solutions for geospatial data management and visualization that are more accessible, affordable, scalable, configurable, and, most importantly, valuable. Within this document, the project purpose, background, and justification will be clarified by reviewing each of the five contributing aspects of the project. By understanding the key topics and tying them together, the possibilities that can be achieved by pushing beyond the boundaries of familiar geographic information system (GIS) tools and services come to light.

# Definitions

To establish a common understanding, brief definitions and context are provided for the lengthy project title.

1. Open-source – In technology, this is code that is publicly available and free to use, modify, and distribute. (Red Hat, 2020b)
2. Cloud-native – In the context of software applications, this refers to how software applications are designed and built to take advantage of the benefits of cloud-based computing. (VMWare Tanzu, 2020)
3. Serverless – In cloud-based computing, serverless simply means a method of running software applications without provisioning the underlying infrastructure. The cloud-computing provider allocates computing resources on behalf of the software application builders. (Cloudflare, 2020a)
4. Web-mapping – Using the World Wide Web to access the Internet, geographic data can be delivered dynamically in the form of a map on a website. (Axis Maps, 2017; Technopedia, 2020)
5. Agriculture Industry – The focus for this project is specifically on crop growers and organizations that access and/or manage field data for growers such as farm managers, agriculture retailers, grower associations, crop consultants, and agronomists.

# Project Purpose

Understanding the project purpose requires acknowledging the proliferation of georeferenced data being created today. This can be examined more closely with an industry use case. Shimonti Paul wrote for Geospatial World in 2018 about how the GIS is empowering the agriculture sector. Specific to types of geospatial data, he wrote:

*“Sensors in fields, on tractors and on satellites high above farms are constantly collecting data. Advanced technologies are able to turn this data into information that farmers and land managers can use to make more informed and timely decisions. This, in turn, boosts productivity and reduces environmental impacts.*

*Farming is getting smarter with the availability of advanced technologies like precision equipment, the Internet of Things (IoT), sensors and actuators, geo-positioning systems, Big Data, Unmanned Aerial Vehicles, robotics etc.”* (Paul, 2018)

GIS professionals understand the value that geographic data brings to a topic especially when it can be aggregated and visualized. When visualizing data spatially, patterns emerge and a story can be immediately conveyed. A web map of wheat crop locations in the United States overlaid with locations of extreme drought for the year instantly reveals national wheat production impact. The value of this type of data lies in its organization, management, and visualization. Without that, it is little more than the tabular, spreadsheet-style data that is time-consuming and difficult to analyze.

Spatial data is inundating the industry. Geographic data management solutions are being created, but the industry has not traditionally been GIS-savvy, and these solutions have a price and generally serve a very specific purpose.

This project reflects personal relationships with people in the agriculture industry, more than twenty years of professional experience developing software, and engagement with academic research relevant to geospatial technologies, both open-source and proprietary. This experience, by fostering a knowledge of technologies in general and GIS-related, ultimately informs an innovative approach to web-mapping solutions. It is an approach sensitive to migrating technical infrastructures and applications to cloud-computing providers and refactorization of software applications into cloud-enabled architectures.

As progress continues, the technology industry still has room for improvement. Organizations may not yet be taking full advantage of cloud computing by building true, cloud-native solutions. Often during the migration of legacy applications, cloud computing providers are treated like a traditional data center. Actual machines are allocated with specific hardware and operating system specifications. It will take time to completely rewrite many of the older-style, monolithic applications into smaller microservices that can be managed by cloud computing services eliminating the need for actual machine provisioning. In other words, the need to keep these legacy systems operational slows the progress towards “serverless” solutions.

As with the software industry, the agriculture industry is experiencing a paradigm shift in regard to how to build and deploy technical solutions, the agriculture industry is as well. Precision agriculture (PA) is the most recent example of how that industry is adopting a different approach to farming. The software company Trimble describes PA in the following terms:

*“The primary goal of precision agriculture is to strive for profitability, efficiency and sustainability on the farm. This is achieved through a combination of PA technology and PA equipment. First, PA technology gathers and analyses data from every action performed on your operation and helps guide both your immediate and future decisions: what seed to plant in what field, or where exactly you need a precise amount of fertilizer or chemical. Then, once you have an idea of what needs to be done on your farm, PA equipment puts your plan into action. For example, you can prep your land by using automatic section or variable-rate application control. Or, you can precisely maneuver your tractor and implements with a hands-free steering system, like the Autopilot*[*™*](https://agriculture.trimble.com/precision-ag/products/steering-systems/)*Automated Steering System. And, with the right farm management software, you can manage complex prescriptions across a variety of productivity zones all in one platform.*

*The PA movement started in the 1990s with the introduction of Global Positioning Systems (GPS) and Geographic Information Systems (GIS). A wide range of sensors, monitors and controllers were also developed during that time and were used on ag equipment like shaft monitors, pressure transducers and servo motors**[[2]](https://agriculture.trimble.com/blog/what-is-precision-ag/" \l "_ftn2). With the rapid introduction and adoption of mobile computing, high-speed Internet and reliable satellites, the reach and usage of PA has grown immensely over the past decade, so much so that it now touches almost every area of a farm operation.”* (Trimble, 2020)

Trimble summarizes the benefits of PA as improved overall efficiency, reduction in labor and crop input costs, reduction in water usage, and reduction in time spent planning each growing season. (Trimble, 2020)

This is certainly not the first time the agriculture industry has experienced significant change. In 2017, Ernst & Young Global Limited published a series of articles defining digital (or precision) agriculture as the “the third great revolution of modern agriculture. “ They refer to this as Ag 3.0 and recognize earlier paradigm shifts in agriculture history.

*‘The introduction and implementation of mechanization (1900 to 1930) and genetic modification (1990 to 2005) are referred to as Ag 1.0 and Ag 2.0 respectively. Both revolutions drove efficiency, yield and profitability to levels previously unattainable, and are now conventional in developed countries.*

*While Ag 1.0 and Ag 2.0 drove significant changes in agriculture, we believe Ag 3.0 will be the most transformative and disruptive, not only on the farm but across the entire agriculture and food value chain.”* (EY Global, 2017)

They posit that “digital (or precision) agriculture and big data hold answers to the problem of how to feed a growing world sustainably.” These articles attempted to address the effects of these changes on agribusinesses specifically. The series continues by suggesting that digital agriculture offers an “opportunity for new business models.” The article recognizes that agriculture retail typically includes agronomic advice, crop scouting, custom fertilizer/chemical applications to crops, as well as product sales. These businesses serve as “both advisor and supplier.” They see an opportunity in shifting “from selling products to selling outcomes.” (Dongoski et al., 2017)

A December 2019 article in CropLife continues this theme and offers advice to agriculture retailers. The article recognizes that data is the new “risk management tool.” Regarding the relationship between the retailer and grower, the article suggests that the data must revolve around guaranteeing grower success. Farm Market iD CEO Steven Rao is quoted as saying, “Farm and grower data — not agronomy data — can help ag retailers increase efficiencies and insights to better serve the grower. In a time of sustained low commodity prices, farmers have substantially limited spend potential compared to even a decade ago. The retailers who can best connect their products and services to the farmers’ specific needs and context will be the ones who win the game.” (Skernivitz, 2019)

Actual data management solutions for crop consultants and agriculture retailers have yet to be fleshed out. Interviews with two Texas agronomists were conducted for an on the ground perspective. Jim Farley of Comanche County, Texas has been an agronomist with an agriculture retail business his entire adult life. Along with him, Bob Whitney has served as an agriculture extension agent in various Texas counties and currently consults and does research work in agriculture and community development. Whitney’s current clients include Texas Pecan Board and Texas Peanut Producers Board plus Jim Farley’s business.

Between the two of them, they have over seventy years of crop consulting experience combined. Farley manages thousands of grower fields. Whitney’s consulting group is “dedicated to helping the world's poor living in rural areas better themselves through agriculture, community development and youth training.” He has overseen programs in over twelve countries. He works with universities, government agencies, and non-governmental organizations (NGOs) “in projects to help alleviate hunger and poverty and improve agriculture and wealth creation for producers. ” (J. Farley, 2020a; Whitney, 2019)

According to observations of Farley’s business, in 2018, Farley uses whatever services he can find freely (or very inexpensively) to manage customer crop data. Google Earth Pro is used for mapping out crop locations. Subscriptions to imagery services specific to the agriculture industry are employed. Beyond that, there are still filing cabinets full of customer soil sample records, fertilizer applications, and various other historical information. Both Whitney and Farley are using GIS without realizing what they are doing. Whitney has described many of the current challenges in getting all the data coming in from disjointed sources such as drone imagery, satellite imagery, soil data, and crop locations into one place where it can be overlaid, analyzed, and attributed. He talked about some of the organizations’ goals and ideas, but also the many challenges related to the management and usefulness of information that cannot be aggregated together in a meaningful way. Whitney has only a rudimentary knowledge of GIS products like ArcGIS and QGIS, but lacks thorough understanding of how to layer the disparate data. (Whitney, 2019)

Farley puts it in very simple terms by stating, “Whoever is going to be producing food for the world in the future is going to need technology. Things like yield monitors, soil tests, GIS layers, irrigation monitors.” (J. Farley, 2020b) Both Farley and Whitney realize the importance of the data. They attempt to take advantage of solutions that are affordable and intuitive.

There are advantages to being a bit behind the technology curve. Unlike the traditional software industry, the agriculture industry is still new to production guided by data and software. It is not littered with legacy systems that inhibit the adoption of modern technology. They have not yet locked themselves into sticky relationships with vendor-specific software. Solutions engineers can start from a clean slate when building out data management solutions for the agriculture industry.

All of this together presents a unique opportunity to build something new. Unencumbered by existing system requirements, a truly cloud-native GIS solution deployed in a serverless fashion to a cloud computing provider platform could meet the data management and visualization demands of the agriculture industry. Controlling costs for both growers and agriculture retailers is always important, but especially so as commodity prices drop. As new sensors and equipment are introduced in the agriculture sector, the data not only becomes more varied, but also larger in the sense that it will continue to require more storage space. Keeping any solution in the open-source spectrum ensures accessibility, configurability, and control. With respect to needs for cost containment, this also provides an affordable solution.

The following sections of the document will explore the five parts of the project proposal in more depth. Within certain sections, existing literature and examples of how open-source, cloud-native architecture, serverless computing, and web-mapping applications have already been used and even tied together will be reviewed. The goal of this study is to provide insight into the problem and justification for the project, thereby clarifying the reasons driving the need to move toward a geospatial data management and visualization solution that is open-source, cloud-native, deployable in a serverless fashion to a cloud computing provider platform, and beneficial to the agriculture industry.

# Agriculture Industry Trends

Digital revolution aside, there are other industry trends driving the need for changes in the way food is produced. Consumer demand for sustainable food production and commodity traceability is on the rise. Grocers like Whole Foods cater to customers concerned with sustainable food production and food supply chain traceability. Only products that meet their rigid quality standards are sold in the stores. (Whole Foods Market, 2020) Transparency into how the food is produced is essential to ensure that the standards have been met. This requires capturing and managing data related to the food production in a digital manner that can be passed to companies selling the products. (Farmobile Editors, 2019)

The December 2019 Farm Journal AgTech Expo in Indianapolis, Indiana defined some of these trends. Presentation topics varied. Speakers covered growing hemp, precision agriculture techniques for increasing yield, irrigation technology, and, most interestingly, farm data management. A key theme emerged from conference keynote and breakout session speakers. Sustainability, traceability, and accountability around the food purchased by consumers is becoming increasingly important. (Cubbage, 2019; Griffin, 2019; Heneghan, 2019; Ruppert, 2019)

Concerns for transparency in the modern food system are not new. Upton Sinclair exposed issues in the meat packing industry in 1906 when he published *The Jungle*. President Theodore Roosevelt conducted his own investigation after reading an advance copy of the book. This accelerated the passage of the Meat Inspection Act of 1906. (Rouse, 2020)

A fourth-generation farmer, frequent Farm Journal columnist, and precision ag consultant, Steve Cubbage works with farmers to implement and manage hardware and data and bridge the gap between the two. At the conference, his presentation elaborated on how grocery stores are changing. He stated that over 60% of modern grocery stores are committed to fresh foods while the packaged/processed sections are shrinking. He went on to state that customers are demanding transparency in regard to food sources. In his professional role, he works to implement universal data standards and digital exchange of farm data. (Cubbage, 2019)

At this same conference, presenter Jim Heneghan of Gro Intelligence discussed the various public agency sources of supplemental data that can be overlaid with grower-owned data for enhanced analysis. Gro Intelligence comprises data scientists focused on aggregating data at a global level to provide insights into agriculture trends. Lance Ruppert of Growmark, Inc discussed the ag retailer’s role in adopting technology.

In short, data management was a hot topic at both the individual grower level and the crop consultant/agriculture retailer level. This supported earlier concerns over data management articulated by the Farley business in addition to the input from consultant Bob Whitney.

In January 2020, Steve Cubbage wrote an article for Farm Journal’s AgWeb continuing his discussion about the changing consumer demand for agriculture products. He states, “future profitability may depend more on others knowing how you produce food and not simply focusing on producing more of it.” He continues by pointing to two examples of industry programs acknowledging this emotional side of food production. Land O’Lakes SUSTAIN pushes for sustainability and creating scorecards for major corporations. The Indigo Terraton initiative is a “push to transform agriculture as the champion fighting climate change and reclaiming one trillion tons of carbon from our atmosphere.” Cubbage completes the article by asserting that data is at the center of all these programs. He concludes with, “data transparency about the farmer, the fields and the food that they grow will likely be the ticket you’ll want to have in this new decade.” (Cubbage, 2020)

The data is coming from a variety of sources including satellite imagery, drone imagery, GPS sensors on farm equipment, field sensors, weather sensors, soil samples, as well as the old-fashioned visual observations. The data is both vector and raster data formats and has an important temporal component. This is an intimidating amount of data for any individual grower to negotiate. For an agriculture consultant, growers’ association, or agriculture retailer servicing multiple growers, this amount increases exponentially and the management/analysis of it all requires a robust, scalable solution. Managing the data, visualizing this data, and performing analysis on this data must be affordable and customizable so that agronomists can focus on growing crops instead of becoming computer technologists as well.

Affordability is important to both growers and retailers as commodity prices fluctuate and the ownership of farmland continues to change. Farley predicts that it is going to continue to be difficult for agribusiness because it is increasingly harder for farmers to pay for services. Drops in commodity prices, increase in land prices, and equipment costs are driving what Farley considers “true” farmers out of business. In his experience, he sees land ownership transferring to people who are buying up farmland as a way of diversifying investments. Generally, these new land owners only produce enough to retain agriculture exemptions or they are part of a larger corporation. (J. Farley, 2020a)

Expanding on Farley’s observation, Willie Vogt wrote in January 2020 Farm Progress article about the changing customer impacting ag retail. While Farley is an independent retailer, Vogt interviewed the digital lead, Sol Goldfarb, of a major market supplier Nutrien Ag Solutions. The company is attempting to leverage digital tools to provide convenience to the grower.

*“He acknowledges that farming is a high-stakes business, and there are a lot of decisions that you have to get right the first time. The aim of the new digital platform the company has been working on is to help growers improve their operations and achieve their goals.*

*These tools also help to maximize the farmer-retailer relationship by providing the information you want when you need it; and by helping you leverage data you collect for improved outcomes.”*(Vogt, 2020)

The article concludes with Goldfarb explaining 2020 plans to build out tools that capture grower data so they can “demonstrate to the grower that we can help them improve the outcome on the farm using this information.”(Vogt, 2020)

As growers continue to take advantage of GPS devices, sensors in the field on equipment, imagery from drones and satellites, and practicing precision agriculture, the data will continue to expand and change. Agility in terms of being able to quickly take advantage of any new information being fed into the system will be critical. Referring back to the assessment from Ernst & Young, “As the industry evolves, disruption will follow. It is essential for agribusinesses to transform their business and themselves to differentiate and provide more value to customers.” (EY Global, 2017)

There is still another trend that is driving the need for inexpensive, customizable solutions. In December of 2019, American food system news source Civil Eats interviewed small-scale farmers in the  community-supported agriculture (CSA) business. These farmers need low-cost technology. They acknowledge that there is a dense field of ag-tech companies. The article states, “In 2018, tech startups along the entire agriculture and food supply chain securing nearly $17 billion in funding, with farm management and sensing startups alone attracting $945 million, according to AgFunder.” The issue is that these solutions have upfront costs and subscription fees that are simply too expensive for these small-scale growers.

*“Want to use satellite technology and other data to make decisions on irrigation and fertilizer application? Try Cropio, which costs 40 cents to $2 per acre per year. Seeking a business management solution to measure revenue and analyze field-by-field profits? Harvest Profit’s software runs $1,500 per year. (A consulting package that includes personalized calls twice per month goes for $10,000.)*

*Need a full-service solution that manages everything from worker punchcards to spraying events to RFID harvest tracking? Croptracker offers a per-service pricing plan that starts as low as $5 per user per month, but can easily run to hundreds of dollars per month for more comprehensive packages. Major names such as Granular, Conservis, Agworld, and others offer only custom quotes.”*(Orlowski, 2019)

The article advocates for open-source solutions not only for controlling costs but also for the ability to modify the underlying code and customize the solution to meet the needs of the user. This group of farmers also recognizes the need for independence. By using open-source technology, the farmers “are running the software themselves. They are not dependent on another company, because companies come and go.” Finally, the article makes the argument that open-source technologies help small farmers compete against large corporations. (Orlowski, 2019)

Regardless of size, revenues, and market shares, the people that produce food are changing together with the tools used to drive that production. Consumer demands are changing. Food production sustainability and traceability are important enough that some grocers may only stock items meeting specific standards. Transparency into the supply chain from farm to store means digital data must be captured, organized, and made accessible to meet the needs of this changing industry. Existing agribusiness is already facing survival challenges. In order to adapt to this new data-driven method of farming, the solutions need to be inexpensive, customizable, and, most importantly, reliable. This agriculture industry can benefit from open-source solutions that do not have subscription nor usage fees, give ownership of the data and underlying code to the user, and can be quickly modified as the industry evolves.

## Existing Solutions

Data management, specifically geospatial data (for the purposes of visualization and analysis), is quickly becoming a necessity in the agriculture industry. More than just storing it, the geographic attributes of this data call for it to be visualized and managed in a web-mapping system to easily identify locations, spatial patterns, and relationships to other data. It is worth investigating existing solutions available today.

The need for solutions has not gone unnoticed by the technology industry. At the AgTech Expo, both IBM and Microsoft were mentioned by multiple presenters as technology leaders creating solutions for the agriculture industry. The IBM Watson Decision Platform for Agriculture aggregates and analyzes “terabytes of multi-layer geospatial data using machine learning and advanced analytics” aimed at improving crop yield. (Mello & Raghavan, 2018) Microsoft’s Azure FarmBeats solution offers a comparable solution aiming to achieve four goals: 1) Aggregate agricultural data from different sources, 2) Fuse different agricultural datasets from sensors, drones & satellites, 3) Rapidly build AI/ML models using the fused datasets, and 4) Build customized digital agriculture solutions. (Microsoft, 2020a) These two examples represent powerful, potentially life-changing solutions to global problems. The ultimate intention of solutions such as these, all of which are funded by the large, recognizable technical industry leaders, are far broader than simple geospatial data management. They aim to feed the planet.

At a smaller scale, solutions exist to serve a specific purpose for the industry. Some are designed specifically for the industry while others have been adopted by the industry. As noted earlier, one example of a tool designed for generic purposes that was adopted by Farley in his business is Google Earth Pro. He uses this specifically to create crop boundary polygon files that he then saves in Keyhole Markup Language (KML) format on the file hosting service Dropbox for future use. When interviewing crop consultant Bob Whitney, he listed a few other industry-specific tools that provide geospatial data and/or services. DroneDeploy stitches together georeferenced drone imagery for visual inspection. Granular provides satellite imagery of certain crops. Land IQ provides insights related to plant health by “pairing land-based scientific knowledge with advanced analytical modeling technologies and remote sensing.” (Land IQ, 2018; Whitney, 2019)

A more expansive list of solutions offering some form of data management follows. It is worth noting that ALL of them are paid services, and that pricing is based on number of fields being managed, amount of data being stored, and/or some other use-limiting factor.

Technical consultants at the AgTech Expo represented a product called FieldView offered by the company Climate Corporation. This offering advertises data storage and data visualization capabilities. The entire system incorporates cloud data storage with the ability to upload historical data, a device that plugs into equipment to capture data, and data visualization. Beyond that, the solution can provide an analysis of the data related to crop performance and field health. It can also provide seeding prescriptions and crop fertilization plans. The data connectivity feature even allows for transfer of data from other industry partners such as DroneDeploy. While comprehensive, this offering has a tiered pricing structure based on feature additions. The solution is tailored to benefit the individual grower or farming operation. Pricing tiers differ based on the services desired. (The Climate Corporation, 2020) Research into this offering resulted in a couple of unfavorable conclusions. The data storage limits and pricing structure are unclear. The purchase of a FieldView plug-in device for farm equipment without sensors or for equipment with sensors that are not part of the FieldView partner network is required. Any other sensor may not collect data that can be uploaded to the FieldView system. Above all of that, adopting this solution means relinquishing control of the data management and ability to create a custom solution.

A quick Google search yields a few different lists of the top crop management software applications. Predictive Analytics and G2 list Croptracker. (G2, 2020a; Predictive Analytics Today, 2020) Granular is listed again. FarmLogs, Conservis, Farmer Core from Trimble Ag, and Agrivi also show up multiple times. All of these solutions are far more encompassing than merely managing and overlaying geospatial data. Often these products are farming enterprise solutions including financial management, planting and harvesting management, inventory and resource management, integrated health and weather monitoring, and additional farming activities such as dairy and livestock farms. (Software Connect, 2020) For an individual grower or one farming enterprise, these solutions may make economic sense. They are an overkill for the consultant with geospatial data that spans many growers and fields.

In contrast to that are solutions that offer a very specific service. Satellite and earth imaging company Planet offers field health monitoring with daily streams of 3-5 meter resolution imagery. (Planet Labs Inc, 2020) Geospatial mapping solutions company Tierra Plan “is developing a custom field management solution for an independent agronomy provider built around the Esri ArcGIS platform. Called Field Analyst, it is a scalable GIS solution for managing and analyzing field data so growers can optimize soil treatments and yields. A suite of ArcGIS desktop tools and web maps lets agronomists and growers manage their soil data, yield, as-applied, EC data, generate nutrient maps, prescription files, management zones, and share data using online maps.” (Tierra Plan, 2019) A 2019 research paper describes the technical specifications of the CropSight solution for plant phenotyping. This system has a PHP Hypertext Pre-processor and structured query language-based server platform that provides automated data collation, storage, and information management through distributed IoT sensors and phenotyping workstations. (Reynolds et al., 2019) The authors describe the architecture, software frameworks, database, query language, data transfer protocols used to create the system features. They even provide the link to the source code repository.

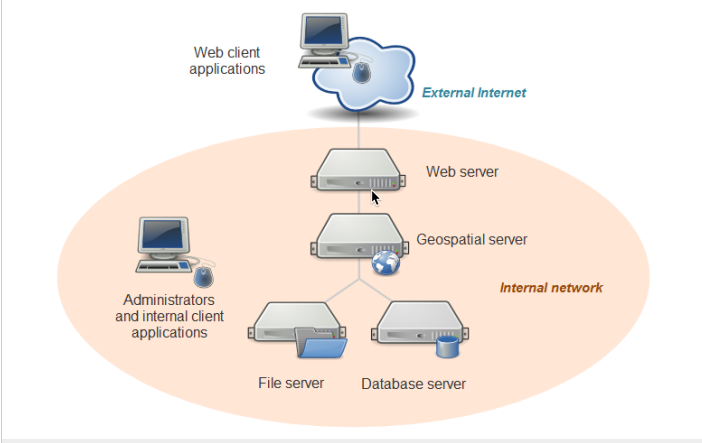
One company, AgriXP does offer a product “for agronomists, consultants, farm workers and multi-farm managers has been carefully designed to provide real experience with crop monitoring, farm works and scouting.” (AgriXP, 2020) A quick look at the pricing plans reveals that even the most expensive plan limits users to sharing data with only 100 growers by individual account creations. It is not clear just how much data can be saved nor what the geospatial capabilities are. The information about the product implies that most of the data is limited to tabular data.

In summary, there are many software solutions with mapping and/or geoprocessing capabilities available today. Some of them may be ideal solutions for particular individuals or enterprises. The drawbacks are that they come with a price tag and none focus solely on leaving the management of the data in the data owner’s control. Furthermore, consumers are limited to the visualization and analysis options provided by the solution. They have no control of the code and cannot configure it to scale or operate differently. Going back to the earlier observation of Farley’s enterprise, the existing solutions result in a hodgepodge collection of free and paid services to manage the constant swelling of geospatial data. This is neither efficient nor sustainable. As data accumulates from a variety of sources, the tools to analyze that data in a meaningful way constantly change. Paying for additional services to manage or analyze this data will drive up operation costs. Some of the companies providing these services could go out of business. Agribusinesses need to harness solutions that allow them to maintain ownership of data and data management/visualization solutions for business continuity and cost containment.

# Common components of a web-mapping solution

At this point, it is helpful to review what actually makes an information system “geographic” as opposed to just an information system. Industry leading GIS software supplier ESRI defines a GIS as “A geographic information system (GIS) is a framework for gathering, managing, and analyzing data. Rooted in the science of geography, GIS integrates many types of data. It analyzes spatial location and organizes layers of information into visualizations using maps and 3D scenes. ​With this unique capability, GIS reveals deeper insights into data, such as patterns, relationships, and situations—helping users make smarter decisions. ” (ESRI, n.d.) At the core of a GIS is the presence of data attributed with geographic coordinates and accompanying coordinate system(s). The data may have one or more pairs of longitude/latitude values. The underlying horizontal (and possibly vertical) coordinate system(s) must also be known in order to accurately position the data. This data is visualized and analyzed on a map. In this way, the data is immediately meaningful to the end user. Locations are clear and spatial relationship patterns among the data emerges. Within the GIS industry, professionals will often hear phrases such as “maps can tell a story.”

A common solution used to work with geospatial data is a web-based mapping application. These may also be called web GIS applications. While similar to any web-based application, the web GIS applications have a distinct component that allows geographic data to be viewed (and potentially analyzed) on a map. Different geographic data may be layered, and users can often interact with the data in some way. A web-mapping (or web GIS) application will include a few basic components within a fairly standard application architecture. ESRI asserts that there are “five essential elements in every web GIS application.” These elements are 1) a web application, 2) digital basemaps, 3) operational layers, 4) tasks and tools in the web GIS application, and 5) one or more geodatabases. (ESRI, 2019) Pennsylvania State University (PSU) curriculum lists three elements: basemaps, thematic layers, and interactive elements. (Quinn, 2019) Within the same PSU coursework, a diagram of the system architecture for a web-mapping application is presented. Figure 1 is the diagram highlighting the common parts of the application.

  
**Figure 1.** System architecture for web-mapping applications presented by PSU

© Penn State College of Earth and Mineral Sciences, John A. Dutton e-Education Institute. All rights reserved.

Reproduced here for educational purposes only.

Key pieces of the system exist in the “Internal network” portion of the figure. Geospatial data is stored in files and/or databases. The geospatial server retrieves the geospatial data and transforms it to be drawn on a web. This becomes the operational or thematic layers that are viewed on top of a basemap. The web server hosts the user interface to the application. The interface will have one or more maps that generally consist of an underlying basemap that is either aerial imagery or a planimetric map. The geographic data being served from the geospatial server is overlaid on the basemap. (Quinn, 2019)

In a 2017 proposal for Managing and Interrogating Multiyear and Multiscale Bridge-Inspection Images, researchers proposed a web GIS called *BridgeDex*. They described the basic components of a web GIS with an accompanying figure. Figure 2 is the diagram presented by the research team.

A screenshot of a cell phone

Description automatically generated

**Figure 2.** Basic components and workflow of a web GIS as presented by *BridgeDex* researchers

© 2017 American Society of Civil Engineers. All rights reserved.

Reproduced here for educational purposes only.

This team of researchers specifically called out the GIS Server as the “critical component” of the application. (Javadnejad et al., 2017)

An earlier research paper from 2012 by researchers at the University of Brasilia ﻿presented an architecture for webGIS based on Web 2.0 and interoperability among different geographical data. Section 2 of the paper introduced the basic concepts of webGIS.

*“﻿Web GIS is any GIS that uses Web technologies. The simplest form of webGIS should have at least a server and a client, where the server is a Web application server, and the client is a Web browser, a desktop application, or a mobile application.*

*With regard to the architecture of a webGIS, the architecture based on three layers is most commonly used: User Interface Layer, Application Server Layer and Database Layer. Some authors considered four layers, where the integration layer is added on the architecture webGIS, which is based on web services.*

*The User Interface layer serves as a graphic user interface (GUI) to present the result of spatial data, allowing the end users to interact with the backend services*

*The Application Server layer communicates with multiple data sources via the data integration layer, and interacts with end users to analyze and manipulate data coming from data provider services.*

*The Database layer of data provider services, is a set of remote data provider services for data sharing. Each data provider service offers a set of interfaces through which client applications can pull remote data in and manipulate the data.”* (Pascaul et al., 2012)

While the basic components are high-level enough that there is nothing to discern this from any other web application, the abstract architecture figure proposed in this paper lists very specific options for each layer. It even splits up the high-level layers to distinguish the geospatial components. Figure 3 is the model presented by the Brazilian research team.

A screenshot of a social media post

Description automatically generated

**Figure 3.** Abstract model of proposed webGIS architecture as presented by University of Brasilia researchers

© 2012 GEOProcessing 2012 : The Fourth International Conference on Advanced Geographic Information Systems, Applications, and Services. All rights reserved.

Reproduced here for educational purposes only.

A notable part of the 2012 research as depicted in Figure 3 is that the team went to great lengths to propose options that are both proprietary and open-source for the different layers. At the database layer, both proprietary ArcSDE is listed with open-source PostgreSQL. The team does this again when listing map servers. Proprietary ArcGIS Server is listed as an option with open-source GeoServer. This is important because it demonstrates that for every proprietary option, there is also an open-source option to build out an entire web-mapping system.

Understanding the architecture components is critical to a web-mapping application when examining how to host these applications outside of a traditional data center. Traditionally each layer in the architecture is hosted independently on an actual machine or a virtual machine designation. Backup or disaster recovery instances may also be configured and on stand-by. Each layer of the architecture must be configured by installing software and configuring communication links between the machines. The machines themselves must be maintained by monitoring the underlying hardware and continually patching the operating system. Any significant changes to the machines mean a reconfiguration of that layer of the web GIS architecture layer. Maintaining “bare metal” machines whether it be actual hardware or virtual machines requires continual maintenance, ongoing monitoring, and machine administration skills.

# Open-source technology

Red Hat, producer of the open-source Linux operating system platform defines open-source software as “code that is designed to be publicly accessible—anyone can see, modify, and distribute the code as they see fit.” (Red Hat, 2020b) It is important to understand that open-source does not necessarily mean “free” software, but it does mean free to use. PSU curriculum describes “free and open-source software” (FOSS) with “a common analogy describing the F in FOSS is ‘free as in free speech, not free beer.’ In other words, FOSS is ‘free’ in the sense that it is open and amenable to use and modification.” (Quinn, 2019)

At the time of writing this report, the 2020 coronavirus pandemic and global market recession is reminding the software industry of earlier recessions. Open-source has historically survived these recessions. Dries Buytaert, founder of open-source platform Drupal, recalls that the 2000-2001 dot-com crash compelled companies to adopt the open-source operating system Linux. This accomplished two things. It cut technology costs and allowed for innovation at speed. Fast forward eight years to the 2008-2009 recession, Red Hat, provider of enterprise open-source solutions including Linux, outperformed proprietary software giants Oracle and Microsoft. Buytaert contends that today “Open Source is still less expensive than proprietary software. In addition, Open Source has grown to be more secure, more flexible, and more stable than ever before. Today, the benefits of Open Source are even more compelling than during past recessions.” (Buytaert, 2020)

Matt Asay writes for TechRepublic that open-source could prove helpful during the pandemic. He also references earlier recessions and concludes that, “Open source adoption has been accelerating for a long time, but as open source vendors and communities experienced during the last recession, it tends to do even better as things get bad.” He elaborates by providing specific examples of open-source companies that are thriving such as Elastic, Cloudera, and MongoDB.

*“Today, buying into open source requires even less risk than it did back then, when companies were still testing the waters. Today for things like data infrastructure, open source is already recognized as the safe, innovative choice.”* (Asay, 2020)

Taking this into account with future market uncertainty and the ever-changing landscape in food production, the agriculture industry is a prime candidate for open-source technology. Assuming ownership of the data and technical systems driving agriculture systems not only ensures sustainability in terms of our ecological resources, but also sustainability of the growers and agribusinesses supporting them. Costs remain as low as possible and systems can be customized to meet demand in a timely fashion. This independence from proprietary software solution providers adds to the durability of the agriculture enterprise during economic downturns.

## Open-source GIS

Generally, open-source solutions cost less and surrender control of the underlying code to the users. (Red Hat, 2020b) Control of the code means that organizations can respond to changing needs and direction without having to consult or collaborate with external software providers. (Carey & Macaulay, 2019) Within the GIS industry, one of the most recognized providers of proprietary GIS software is ESRI. (G2, 2020b; GISGeography, 2020) This company continues to produce robust and stable products that meet the ever-expanding needs of geospatial data owners and users. Of course, these products and services are all licensed and must be purchased. There is also very limited to no control at all in terms of customization or freedom to use the product without potentially breaching contracts with the publisher. Debates around open-source versus proprietary software often center on weighing the costs of purchasing licenses for complete solutions that are “tried and true” versus paying engineers to develop custom applications and services. (Matteson, 2018)

As noted earlier in the discussion regarding the common components of web-mapping applications, proprietary GIS software provider ESRI has solutions for every layer in the web GIS architecture as seen in Figure 3. In fact, ESRI can provide the entire solution. In contrast, there also exists solutions in the open-source realm for every layer in the web-mapping architecture. The Open Source Geospatial Foundation (OSGeo) maintains a list of modern tools and technologies that are freely available to use under an open-source license. There is an abundance of web-mapping applications in production today built entirely with open-source solutions.

In 2009, researchers in China published a paper in the OSSC-2009 - Proceedings of 2009 IEEE International Workshop on Open-source Software for Scientific Computation journal titled “﻿Web GIS Server Solutions using Open-Source Software.” In this paper, the authors defined the important components of the web GIS server as, “﻿Web GIS Server formed by uDig, GeoServer, PostGIS and Eclipse. uDig is a mapping tool, GeoServer is map publishing, PostGIS is a spatial database and Eclipse is a programming tool.” (Xia et al., 2009)

In both professional and academic settings, solutions have been built using PostgreSQL/PostGIS as the database layer, GeoServer installed and configured as the geospatial layer, and user interfaces created with open-source frameworks such as Leaflet. Geospatial code libraries GDAL/OGR and GeoTools have been used for various projects. (A. Farley, 2019a) A Penn State Cloud and Server GIS class project involved configuring a complete web-mapping solution using PostgreSQL database with PostGIS extensions as the geospatial data storage layer, GeoServer to style and draw the geospatial data into layers for the web map, and Apache HTTP Server as the web server hosting the application built using the Leaflet framework. (A. Farley, 2019b) Hardly comprehensive, Table 1 lists common open-source tools that can be used at the different architecture layers within a web-mapping application.

**Table 1. Common Open-Source Geospatial Tools**

|  |  |
| --- | --- |
| Database Layer | PostgreSQL/PostGIS, MySQL, SQLLite/SpatialLite |
| Geospatial/Application Layer | GeoServer, MapServer, GeoTools, Python GDAL/OGR, Apache Tomcat |
| Web/User Interface Layer | Apache HTTP Server, Leaflet, OpenLayers, Mapbox.js, Turf.js, D3, OpenStreetMap |

(OSGeo, 2020; Quinn, 2019)

Specific to the agriculture industry, there are already examples of open-source solutions for managing and serving up geospatial data. A journal article published in 2019 in the An Giang University Journal of Science described building a WebGIS for agriculture production in a province of the Mekong Delta region of Vietnam. The proposed system was built using open-source technology. Differences between the aforementioned Penn State class project and this architecture reside in the web server and user interface framework. Both Apache Tomcat and Apache HTTP Server were used to serve the web application. Instead of Leaflet, Mapbox was used to build the web-mapping interface. Figure 4 shows that architecture. (Binh et al., 2019)

A screenshot of a cell phone

Description automatically generated  
**Figure 4.** WebGIS architecture as presented in “﻿Building Agricultural Information System In An Giang Province”  
© 2019 An Giang University Journal of Science. All rights reserved.  
Reproduced here for educational purposes only.

An earlier article published in 2016 by Italian researchers details a spatial data infrastructure (SDI) “﻿capable of integrating both geospatial datasets and time series information from multiple sources, e.g., multitemporal satellite data and Volunteered Geographic Information (VGI).” This project called Space4Agri was developed to support the agricultural sector in the Lombardy region. The paper begins by coining yet another term for web-mapping applications. “﻿Geospatial information on the Web, also named GeoWeb, is becoming more and more important not only among traditional users (mainly environmental researchers, geographers, and social scientists) but also among public authorities and citizens for the most diverse tasks: retrieval of Point of Interests (POIs), consultation of time series of meteorological and thematic maps for natural hazards, agriculture, etc.” (Bordogna et al., 2016) Their “GeoWeb” solution was also built on open-source technology. The architecture presented for this took into account different data sources and data types. It is worth noting that the data layer contains the PostgreSQL/PostGIS database accompanied by standard file storage. Figure 5 shows the architecture diagram of the system.

A screenshot of a cell phone

Description automatically generated

**Figure 5.** Architecture of the Space4Agri (S4A ) Spatial Data Infrastructure (SDI).

© 2016 ISPRS International Journal of Geo-Information. All rights reserved.

Reproduced here for educational purposes only.

Not only is the agriculture industry using GIS, but its solutions are built on open-source technology. By using open-source, cost may be coming down while control and customization go up. As stated previously though, regardless of whether or not the solution is built on open-source solutions, the differing layers that make up the web-mapping architecture are still siloed onto independent machines or a collection of machines. This could mean purchasing hardware or allocating hardware in a data center. Even if hardware is configured to allow the sharing by way of virtual machines, each virtual machine must be provisioned and maintained.

# Cloud computing

Microsoft defines cloud computing as “the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet (the cloud) to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping you lower your operating costs, run your infrastructure more efficiently, and scale as your business needs change.” (Microsoft, 2020b) Progressing independent of GIS technology, on-demand computing services has become a reality. Cloud computing providers generally accommodate varying service models depending on the amount of control end users wish to have over the resources. Users can allocate machines (virtual machines to be specific) just like in a traditional data center with an Infrastructure-as-a-Service (IaaS) model or they can simply deploy applications to a readily available platform with a Platform-as-a-Service (PaaS) model. In a PaaS model, the cloud computing provider is responsible for the underlying hardware and virtual machine provisioning. The end user just takes advantage of a platform service designed to host the application. For vendors such as the ones listed previously in the discussion about existing software available in the agriculture industry, the ones that are cloud-based are offering their services based on a Software as a Service (SaaS) model. (Watts & Raza, 2019)

The biggest cloud computing providers today in terms of market share and capabilities are Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure (Azure). AWS was the first to offer IaaS and continues to develop its service offering to include artificial intelligence, database, machine learning, and serverless deployments. (Dignan, 2020)

The true benefits of cloud computing center around agility and elasticity. Provisioning resources on-demand and scaling resources up and down as needed provide the most cost-effective and reliable environments for applications whether they be GIS or not. (Amazon Web Services, 2020f)

As organizations move ecosystems from traditional data centers to cloud computing environments, the efforts are generally labeled as “lift and shift.” Basically, the organization takes advantage of the IaaS models and only provisions virtual machines in the cloud to match or better the existing data center hardware. This makes it easier to “lift and shift” existing tools and applications. Often the cost-benefit analysis shows that the cost to refactor applications for a PaaS model outweigh the benefits of the elasticity the model offers.

In order for applications to be truly elastic, they need to be designed, built, and deployed in a way that allows them to scale without keeping a virtual machine provisioned at all times. The Penn State Cloud and Server GIS class project was configured on AWS virtual machines. Despite little to no traffic at all, the fact that machines were provisioned meant that costs were still being incurred for reserving that space. An ideal application (even a web-mapping application) is deployed to the cloud provider’s platform without any infrastructure provisioning at all. There are two key concepts to understand in order to move from an IaaS model to a PaaS model. The concepts are: 1) cloud-native and 2) serverless.

## Cloud-native

VMWare Tanzu defines cloud-native as “an approach to building and running applications that exploits the advantages of the cloud computing delivery model. Cloud native is about how applications are created and deployed, not where. ” This approach centers around the idea of breaking apart traditional monolithic applications and deploying smaller reusable components called microservices independently. “Microservices is an architectural approach to developing an application as a collection of small services; each service implements business capabilities, runs in its own process and communicates via HTTP APIs or messaging. Each microservice can be deployed, upgraded, scaled, and restarted independent of other services in the application, typically as part of an automated system, enabling frequent updates to live applications without impacting end customers.”(VMWare Tanzu, 2020)

Software developers have spent years building out applications that may consist of multiple code libraries, but, ultimately, those independent code libraries are packaged together and deployed as a single unit. This makes it time-consuming and risky to make any changes to the applications. Often, these code libraries have dependencies upon one another too. When architecting cloud-native solutions, the first thing to remember is that each piece of functionality needs to be able to stand on its own. Ideally, that piece could be reused in an entirely different application without any modification. These independent pieces communicate with each other via standard messaging or web protocols.

Of course, getting to this point from a legacy application may never be feasible. This is another reason that the agriculture industry is a great case study. There are no legacy applications, so solutions can be built from scratch with a modern architecture that takes full advantage of cloud computing by ensuring right-size capacity that will minimize computing costs. VMWare Tanzu defines right-size capacity in these terms: “A cloud native application platform automates infrastructure provisioning and configuration, dynamically allocating and reallocating resources at deploy time based on the ongoing needs of the application. Building on a cloud native runtime optimizes application lifecycle management, including scaling to meet demand, resource utilization, orchestration across available resources, and recovery from failures to minimize downtime.” (VMWare Tanzu, 2020)

There are other aspects of cloud-native applications, but the one additional key feature to note is that the application has no dependencies on the operating system (OS) of the machine to which it is deployed. The application is “abstracted” from the OS and only needs a platform to operate. “A cloud native application architecture lets developers use a platform as a means for abstracting away from underlying infrastructure dependencies. Instead of configuring, patching, and maintaining operating systems, teams focus on their software. The most efficient means of abstraction is a formalized platform.” (VMWare Tanzu, 2020) In other words, the application should take advantage of a PaaS model instead of an IaaS model of cloud computing.

The way VMWare Tanzu suggests accomplishing “OS Abstraction” is by way of “containers.” VMWare Tanzu continues, “**Containers** offer both efficiency and speed compared with standard virtual machines (VMs). Using operating system (OS)-level virtualization, a single OS instance is dynamically divided among one or more isolated containers, each with a unique writable file system and resource quota. The low overhead of creating and destroying containers combined with the high packing density in a single VM makes containers an ideal compute vehicle for deploying individual microservices.” (VMWare Tanzu, 2020) One can think of containers as lots of tiny machines running on a host machine. While this does utilize the underlying VM more efficiently, actual machines still must be provisioned. Cloud providers have been quick to respond to this demand and have begun to offer services that handle the machine provisioning for the end user. Containers are actually the midway point between traditional machine provisioning and the most modern form of deploying software. To fully exploit the benefits of cloud computing, system architects are taking advantage of “serverless” options instead of using containers.

## Serverless

In contrast to containers (which can be thought of as tiny VMs), serverless computing delegates the responsibility of managing underlying resources back to the cloud computing provider. “Serverless” is a misnomer. There are still underlying machines. Those machines are provisioned and managed entirely by the cloud computing provider. In a 2018 paper by Dr. R. Arokia Paul Rajan of Christ University in Bengaluru, serverless computing is defined as “﻿an execution model in which the cloud service provider dynamically manages the allocation of compute resources of the server. The consumer is billed for the actual volume of resources consumed by them, instead paying for the pre-purchased units of compute capacity. This model evolved as a way to achieve optimum cost, minimum configuration overheads, and increases the application's ability to scale in the cloud.” (Rajan, 2018) In a 2019 article published in *Communications of the ACM*  magazine, researchers define serverless computing as “﻿a platform that hides server usage from developers and runs code on-demand automatically scaled and billed only for the time the code is running.” The authors continue by pointing out two key features of serverless are cost and elasticity. It is truly a pay-as-you-go model. Users only pay for resources used while the code is running. In terms of elasticity, resources to run the code are scaled up and down automatically by the cloud computing provider. (Castro et al., 2019) These authors continue the discussion by examining another cloud computing service model known as Function-as-a-Service (FaaS). Serverless computing is based on the idea that an application is broken down into independent functions, or pieces of code that perform a specific task. Researchers at UC Berkeley wrote in 2019, “﻿In any serverless platform, the user just writes a cloud function in a high-level language, picks the event that should trigger the running of the function—such as loading an image into cloud storage or adding an image thumbnail to a database table—and lets the serverless system handle everything else: instance selection, scaling, deployment, fault tolerance, monitoring, logging, security patches, and so on.” (Jonas et al., 2019)

Serverless computing comes the closest to realizing the true benefits of cloud computing. Because there are no machine reservations with serverless, users do not have to pay for unused computing resources. If particular functions are used more frequently or if function usage peaks and wanes, the resource scaling happens automatically to meet demand. The progression in cloud computing goes from 1) IaaS models where users provision and reserve VMs to 2) deploying containers to VMs or to a container service offered by the provider that is more like a PaaS model to 3) serverless or a FaaS model. Unless someone is building from scratch, getting to container deployments may be the best that can be achieved without significant rewriting of code. Cloud computing providers recognize this and offer those middle services that allow for container deployments in as close to a serverless ecosystem as possible. Table 2 illustrates the different service model options for computing capabilities from each of the top three cloud computing providers: AWS, GCP, and Azure. This table does list all service options. It lists only the options related specifically to computing resources.

**Table 2. Cloud Computing Service Comparisons**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **IaaS**  **(VMs)** | **IaaS/PaaS Hybrid** | **PaaS**  **(can host containers)** | **PaaS/FaaS Hybrid**  **(containers)** | **FaaS (serverless)** |
| **AWS** | \*Amazon Elastic Compute (EC2)  \*Amazon Lightstail | \*VMWare Cloud on AWS | \*Amazon Elastic Container Service (ECS)  \*Amazon Elastic Container Registry (ECR)  \*Amazon Elastic Kubernetes Service (EKS) | \*AWS Fargate | \*AWS Lambda |
| **GCP** | \*Compute Engine | \*Google Kubernetes Engine (GKE) | \*App Engine | \*Cloud Run  \*Cloud Run for Anthos on Google Cloud | \*Cloud Functions |
| **Azure** | \*Virtual Machines | \*Service Fabric  \*Azure Batch | \*Azure App Service | \*Container Instances  \*Azure Kubernetes Service | \*Azure Functions |

(Amazon Web Services, 2020c; Google, 2020c; Microsoft, 2020b)

The biggest argument for moving to serverless is overall cost reduction. Because users do not have to pay for idle machines, there are multiple examples of drastic cost savings realized by migrating to serverless. (Jonas et al., 2019; Samdan, 2019) In 2018, Anna Helendi interviewed early adopters of serverless to find out exactly how much money an organization could save. In all three cases, the users saved significantly. One user moved from AWS Elastic Beanstalk to Lambda+APIG and saw a 90% reduction in cost. The second case study used EC2 instances for batch file processing. Even with the instances sized at the smallest option, the organization saw costs rising upward to $30,000. The team refactored AWS Lambda functions and saw costs reduce to $4000. The last story was similar in that the users went from a $10,000/month bill to $370/month by switching to serverless. (Helendi, 2018)

Of course, there are tradeoffs when moving to serverless computing. For all the earlier support for recapturing control of an application with open-source technology, promoting FaaS may be hypocritical. The primary negative consequence of serverless computing is “vendor lock-in.” In cloud-based computing, vendor lock-in is a situation where the cost or effort to switch to a different cloud computing provider is too high, effectively locking in the consumer to the provider. (Cloudflare, 2020b) Each vendor will have slightly different features, limitations, and workflows for deploying serverless applications. This could make it very difficult and expensive if the organization needs to move to a different cloud computing provider. (Samdan, 2019) For example, AWS Lambda has limits in terms of function size, memory allocation, and timeout. (Amazon Web Services, 2020c; Sarjeel, 2019) For each FaaS option offered by the top three cloud computing providers, the list of supported programming languages of the functions is limited. Each provider has specific requirements for how to write the functions. Software engineer and technical blogger Sarjeel Yusuf describes this as a loss in overall flexibility when compared to using AWS EC2 instances. Figure 6 is his depiction of the Spectrum of Operation Management and Flexibility of AWS Services.

A screenshot of a cell phone

Description automatically generated

**Figure 6.** Spectrum of Operation Management and Flexibility of AWS Services as presented by Sarjeel Yusuf

© 2019 Medium. All rights reserved.

Reproduced here for educational purposes only.

Over time, serverless computing options could become more uniform across cloud computing providers as consumer demand increases. Google already publishes documentation titled “Google Cloud for AWS Professionals” that “compares Google Cloud with AWS and highlights the similarities and differences between the two.” (Google, 2020a)

Software developers must also be mindful of the “stateless” nature of serverless computing and architect solutions around the concepts of events that trigger the function calls. This can be challenging for applications that need to share temporary data between function calls or chain function calls in a transactional manner that must be managed. (Jonas et al., 2019; Rajan, 2018)

A “stateful” application will store specific information about a user and his/her session with that application in some backend manner. As a user interacts with the application, requests are sent to the backend server and information is shared across those requests. This is common for applications that involve transactions such as banking applications or online purchasing. (Jones, 2018; Sandoval, 2017) In a stateless application, each request the user makes is completely independent of other requests. No information is shared between requests. This means that the entire application must be designed with that in mind. If requests must chain together to act as a single transaction, various strategies can be explored to force user data sharing between requests. Some cloud computing providers offer services to “stitch together” workflows composed of independent microservices. (Amazon Web Services, 2020b; Nolle, 2019) Other strategies involve directly connecting each function to a database and storing user session information upon each function call or writing functions as events that pass payloads to each other. (Taylor, 2019)

Despite the loss of flexibility, the cost savings and ease of deployment tend to outweigh any drawbacks. Moving to serverless is just an extension on the original reason CEOs and CFOs got behind cloud migrations in the first place. Once upon a time, purchasing hardware (and even software) came with the assumption that it would last several years.  Technology is moving so fast that today’s hardware and software are outdated long before companies finish paying for it. Cloud computing is a shift from the up-front capital expenditure (CAPEX) to ongoing operational expenses (OPEX).  (Rosenberg & Mateos, 2011) Serverless computing is a step further in that direction.

Research into existing use cases where serverless computing is used tend to highlight any sort of event-driven computing such as file uploading, live video encoding, IoT data processing of information received from smart devices, and MapReduce analytics. (Castro et al., 2019; Jonas et al., 2019; Rajan, 2018)

This research does not speak specifically to web-mapping applications (nor non-mapping web applications for that matter), but AWS provides a step-by-step tutorial on exactly how to build a serverless web application using a collection of services including AWS Lambda. (Amazon Web Services, 2020c) In comparison, Microsoft offers similar reference documentation for building a serverless web application on Azure with Azure Functions featured as a component. (Microsoft, 2019)

If web applications can be reimagined in an event-driven architecture, then a web-mapping application is nothing more than a special type of web application. To start, the geospatial/application layer must be re-evaluated and refactored into a collection of geospatial functions.

## Existing Cloud GIS solutions

All of these distinct elements can be brought together by recognizing that many cloud-based GIS solutions already exist. This is similar to the discussion about open-source GIS solutions. Just as there are already open-source GIS technologies and frameworks, there are many complete GIS solutions and solution templates for cloud environments. It is not exclusive to open-source technologies either. One need look no further than industry leader ESRI to see the cloud-based commercial products already available. In 2019, ESRI founder and president Jack Dangermond announced:

*“In 2019, we are introducing the term Esri Geospatial Cloud to describe the larger set of technology and products offered by Esri. This encompasses all our software and SaaS offerings, including the following:*

* *ArcGIS software (ArcGIS Desktop, ArcGIS Enterprise, apps, and extensions)*
* *ArcGIS Online (mapping and location SaaS)*
* *ArcGIS for Developers (APIs and SDKs)*
* *Geoenabled products (such as ArcGIS Indoors, ArcGIS Urban, ArcGIS Hub, and ArcGIS Business Analyst)”*

ESRI SaaS solution ArcGIS Online is a complete map making, geoprocessing, data sharing system. Users already familiar with ESRI desktop products can move to this cloud-based service without having to worry about underlying hardware or system configuration. Other paid SaaS services for web-mapping include offerings from Mapbox and CARTO. (Quinn et al., 2019)

This is just the tip of the iceberg. PSU curriculum covers in-depth steps for deploying common ESRI server and desktop applications to AWS EC2 instances. The coursework covers the entire process starting at the cloud computing provider service provisioning to getting an entire system accessible to anyone via a web browser. (Quinn et al., 2019)

For open-source solutions, the Penn State Cloud and Server GIS class project used AWS services and EC2 instances, by divvying up the different layers of the web-mapping application. (A. Farley, 2019b) Cloud GIS is here. Anything that has been traditionally installed on hardware can be installed on resources offered by a cloud computing provider following an IaaS service model.

In reference to the scalability and elasticity of cloud computing, the GIS industry is already realizing the benefits as it relates to data and imagery processing. The entire Google Earth Engine platform is both a data repository and programming platform that is cloud-based. (Google, 2019)  Researchers in Canada published a paper in 2019 detailing their investigations into migrating to a cloud architecture for both performance and collaborative benefits for geospatial algorithm development. Performance gains were described as atmospheric correction algorithms on one year’s worth of Sentinel-3 imagery going from 15 to 20 days down to approximately 23 hours to process three years of data! The biggest gain for this team though was the collaborative aspect. Using cloud storage and containers, researchers did not have to worry about differing operating systems and code dependency problems across desktops. They could package up the entire ecosystem required for running scripts and pass it on to colleagues for continued development. In this same article, the researchers acknowledged that moving from VMs to a serverless architecture would further reduce cost and need for system administration skills. (Jacoby et al., 2019)

Specific to agriculture, many of the solutions mentioned previously are cloud-based. FieldView is cloud-based with a web interface. DroneDeploy offers powerful image processing using cloud infrastructure. The Granular Business software is cloud-based. Croptracker is cloud-based. AgriXP offers a data backup feature for storing data in cloud-based storage. (AgriXP, 2020; Croptracker, 2020; DroneDeploy, n.d.; Granular, 2020; The Climate Corporation, 2020)

In fact, any web application can be migrated to a cloud computing provider. There are a couple of examples of other web-based applications built for the agriculture industry that could be deployed to virtual machines in the cloud.

Developed by Penn State Cooperative Extension Land Analysis Lab, online tool PA*One*Stop includes a mapping module that can “extract data and generate high-quality maps that are required for completion of Nutrient Balance Sheets and Nutrient Management, Erosion and Sediment Control (E&S) and Manure Management Plans” helping farmers meet regulatory requirements. (Penn State Cooperative Extension et al., 2011) This application built in 2011 using Google Maps JavaScript API allows users to create accounts and map out field locations by outlining the boundaries and assigning attributes. Acreage is automatically calculated and other farm features such as streams and water wells can be outlined.

Open-source solution farmOS is designed as a web application that could be deployed to a web server running in a cloud environment. This all-inclusive system is designed to manage all assets and activities of a farming enterprise including livestock, equipment, and fields. Mapping is just one feature of this solution. Fields can be defined as points, lines, or polygons and can be organized in a hierarchical structure. (Red Hat, 2020a; Stenta, 2020)

## Serverless GIS solutions

Can the needle be pushed even further? Cloud computing resources exist. Cloud-based web-mapping solutions exist. There are even cloud-based, open-source web-mapping solutions. Can costs be further reduced and can ownership of data and code be returned to the users by moving away from IaaS and SaaS solutions toward FaaS models?

Some industry experts are already considering this. GIS application developers working for Timmons Group in Virginia delivered a presentation at GeoCon 2019 Virginia GIS Conference titled “﻿Enhancing GIS with Cloud Technology and Serverless Computing.” Their use cases integrated ESRI ArcGIS Online content and services, but they did review various services offered by AWS. Each use case featured a different technology stack of varying services provided by the cloud computing provider. The use case for serverless computing centered around exposing geospatial functions as an application programming interface (API), performing reverse geocoding at scale, and task scheduling of data processing. In the reverse geocoding at scale case, the AWS Lambda service was used. Costs decreased from $5,000/month on the old solution to only $20/month. The volume was reported to be approximately 3 million requests per month. For the scheduled tasks, the researchers also used AWS Lambda and reported a cost of only $10/month. Other key takeaways listed were low maintenance, zero licensing costs, high performance, and scalability. The presentation also included a use case for hosting traditional GIS web applications. For this case though, they reverted back to using a full web application and ArcGIS Enterprise on EC2 instances. (Dhanapal et al., 2019) If an organization is already committed to ESRI products though, these examples demonstrate ways to better architect solutions in the cloud for optimizing costs, performance, scalability, and overall control of the applications. (Dhanapal et al., 2019)

In his graduate research, Ed Santos presented “Considerations on Deploying Geoprocessing Workflows as Serverless Functions.” His focus was on intensive geoprocessing tasks, and he noted that some tasks that involve neighborhood operations or spatial relationship analysis may not be attainable with the current memory and timeout limits of serverless functions. He points to current aspects of serverless functions such as modest maximum processing power, potential bottlenecks with high input/output, and memory limits as items to consider when deciding on serverless functions for geoprocessing tasks. He proposes using more of a PaaS model to run containers as a hybrid approach when more capacity is required. The unfortunate side of this is that the containers are running continuously allocating computing resources that incur costs regardless of actual calls to the code hosted in the containers. (Santos, 2020)

Professional experience grants only cautious optimism. Challenges around actually pushing the entire web-mapping application to a serverless architecture are anticipated. The stateless nature of serverless computing could make getting geospatial data from the database to the web client to be used as a mapping layer problematic. This will likely require using client-side storage in coordination geographic data transformed into a common data interchange format to be passed to and from serverless functions and the database layer.

Performance is another unknown. It is curious that Timmons Group developers stopped short of building out a serverless web-mapping application. (Dhanapal et al., 2019) Helendi covers cost savings in her 2018 research, but performance is never mentioned. (Helendi, 2018) In a 2019 article that was part of a series *The enterprise architect's guide to application state management*, author Twain Taylor warns of the latency transporting data from one function to another. He also offers advice when using a database with serverless functions to force statefulness in an application. Constantly calling the database from the serverless functions introduces scalability issues and stress on the database layer. (Taylor, 2019)

# Project Objective

The project objective is two-fold. The first goal is to build a serverless web-mapping application. The second goal is to answer specific questions around viability and usefulness of this type of application:

1. *Can a serverless web-mapping application be achieved with a stateless architecture or will statefulness have to be enforced?*
2. *Is the performance of the serverless web-mapping application acceptable?*
3. *Is the serverless web-mapping application usable?*
4. *Is the solution affordable?*
5. *Is the solution configurable and reusable?*

To answer these questions, a prototype of a serverless web-mapping application will be constructed and deployed to a cloud computing provider’s platform. Common in software development, this prototype will be a working model of the software application with limited functionality. (Tutorials Point, 2020) The approach and basic requirements to test the validity of the proposed solution are defined in the following paragraph.

References provided by AWS for building serverless web applications coupled with professional experience in open-source web-mapping technologies will be used to build this serverless web-mapping application. The application will be built on open-source technology and frameworks. It will be deployable to a cloud computing environment in a serverless fashion. At each layer of the web-mapping application, the following functional requirements will be addressed:

1. Database layer
   1. Both vector and raster data will need to be stored
   2. The data will have a temporal component
   3. The data will be attributed
   4. The data will must be queryable by both location and attributes
2. Application/Geospatial layer
   1. The architecture used here must be serverless
3. Web/User Interface layer
   1. User interface must be delivered in a serverless fashion
   2. The user interface must be built using open-source frameworks
   3. User interface must retrieve geospatial data and add/update geospatial data
   4. User interface must include functionality to query data spatially and by attributes
   5. User interface must include functionality to add/update attribute data

The outcome of this project will be a working prototype of a serverless web-mapping application. The original questions posed will be answered by the prototype. This prototype will demonstrate whether or not this type of solution is valuable to the agriculture industry. If not valuable, it will answer the questions around why serverless web-mapping is not currently viable.

The prototype will include a template for completing the necessary registration and setup of required services for at least one cloud computing provider.

The prototype may have accompanying scripts and standalone programs for specific data management tasks or other utility services that complement the web-mapping application.

All program code, database schema designs, and configuration instructions will be available in a publicly-accessible code repository with an open-source license for anyone to use and distribute.

# Proposed Methodology

Building upon prior work and tutorials provided by cloud computing provider AWS, a web-mapping application will be produced in iterative cycles. Technology will be open-source. All code will be stored in a publicly-accessible repository. Only readily-available services provided by the cloud-computing provider will be used avoiding the use of VMs in the overall architecture. Figure 7 represents a sample serverless web application design that will serve as a reference and starting point for building out the serverless web-mapping application.

A screenshot of a cell phone

Description automatically generated

**Figure 7.** Application architecture for serverless web application as presented by AWS

© 2020 Amazon Web Services

Reproduced here for educational purposes only.

## Technology

The anticipated technology details that are critical to prototype completion are as follows:

* Cloud Computing Provider – AWS
* Code Repository – GitHub
* Cloud Computing Provider Services
  + Amazon Virtual Private Cloud (VPC)
  + Security Groups
* Database Layer
  + Amazon S3 Buckets
  + Amazon RDS
    - PostgreSQL
      * JSON data type
    - PostGIS extensions
      * ST\_AsGeoJSON function
* Application/Geospatial Layer
  + AWS Lambda
  + Amazon API Gateway
  + Python
  + Java
* Web/User Interface Layer
  + Amazon S3 Buckets
  + Route 53
  + Leaflet
    - HTML
    - JavaScript
    - CSS
  + GeoJSON
  + Client-side storage
  + Basemap options
    - OpenStreetMap
    - Google
    - Planet

The list covers the basic ingredients required to get the web-mapping application up and running. Based on experience with both AWS and GCP cloud computing services, the project will be built using AWS and the services available from that provider. Known services required have been listed. Additional services that could be used revolve around data backups and cost management. AWS offers a variety of services for data storage and automated backups. For security and cost optimization, the Trusted Advisor service and the Cost Explorer are useful tools once an application is production-ready. Amazon CloudWatch can monitor and observe activity of an application. This service can provide the logging and metrics needed to adjust code or architecture for reliability and improved performance. (Amazon Web Services, 2020d)

The Penn State Cloud and Server GIS class project took advantage of the Amazon Relational Database Service (Amazon RDS) without having to provision underlying VMs. The open-source relational database management system PostgreSQL was used with PostGIS extensions for storing spatial data. (A. Farley, 2019b) For this project, both vector and raster data must be managed. Amazon RDS will house the vector data and store pointers to raster data stored in Amazon S3 Buckets.

The application/geospatial layer will be the most challenging. The geospatial data will not be served to the user interface for thematic layers using any of the common map layer services such as GeoServer or ArcGIS Server. This is one of the web-mapping components undergoing the transition to a serverless architecture. As stated earlier, the stateless nature of this could be problematic and will likely require experimentation within the development process. Research suggests that Amazon Step Functions may be required with AWS Lambda and API Gateway services. (Amazon Web Services, 2020b) AWS Lambda supports several programming languages. Java and/or Python will be used due experience.

The web/user interface layer is also part of the serverless architecture. According to the documentation from AWS, the code for the user interface is stored in Amazon S3 Buckets. The interface will be written using the Leaflet framework which is “an open-source JavaScript library for mobile-friendly interactive maps.” (Agafonkin & OpenStreetMap contributors, 2019a) The same stateless challenge resides in this part of the web-mapping application too. The solution will likely require using client-side storage in coordination with the PostGIS ST\_AsGeoJSON function at the database layer and the Leaflet GeoJSON layer in the user interface. (Agafonkin & OpenStreetMap contributors, 2019b; Amazon Web Services, 2020e; Mozilla and individual contributors, 2020; PostGIS Project Steering Committee, 2020) Previous work with this library demonstrated how to add thematic layers using spatial data that has been formatted in the GeoJSON data interchange format. (A. Farley, 2019a)

The prototype will also explore various aerial imagery basemap options for the web map. While free options are available such as OpenStreetMap, the agriculture industry requires up-to-date imagery for crop locations. Farley recounted an incident where he used Google Earth Pro to map out a client field based on a verbal description of the location. The imagery was not current and he mapped out the wrong field because the free imagery he was viewing did not represent recent land clearing that had occurred in that area. (J. Farley, 2020b) This unique requirement necessitates a mechanism for managing and accommodating paid basemap service options in the user interface as opposed to just hard-coding standard, free choices. Applied Geographics publishes an “﻿Aerial Imagery Provider COMPARISON GUIDE” that features several imagery providers. They define six key attributes of each provider including resolution, refresh rate, and spectral bands. Since Near-infrared (NIR) and Red (R) spectral bands are used in the normalized difference vegetation index (NDVI) analysis to measure plant health, this guide is helpful in identifying those providers. For example, Planet offers NIR while Google currently does not. (Applied Geographics, 2020; Google, 2020b; Planet Labs, 2020)

Finally, all code and files required to reproduce this web-mapping application will be stored in a publicly-accessible GitHub repository. “GitHub is a code hosting platform for version control and collaboration.” (GitHub, 2020)

## Software Development Methodology

Development of the web-mapping application will follow an Agile software development methodology. Agile emphasizes collaboration and constant communication with application end users. Development is completed in small increments and released for user testing and feedback. This feedback is then incorporated into the next iteration. This allows changing needs and priorities to be addressed while building additional functionality. This incremental development strategy “means that each successive version of the product is usable, and each builds upon the previous version by adding user-visible functionality.” (Agile Alliance, 2020) Industry professionals Jim Farley, Bob Whitney, and their colleagues will provide the feedback for each iteration of development until the product is either deemed useful or plainly not achievable.

## Expected Challenges

Expected challenges have been stated in the preceding paragraphs. Uncertainty lies around the stateless nature of the serverless architecture. Passing geospatial data directly from the database to the user interface and keeping that data available as thematic mapping layers for the mapping application in a stateless architecture will be a trial and error process until a solution is established. Performance in this type of architecture is unknown. Imagery data retrieval from Amazon S3 Buckets for display on the user interface will require research and testing. In general, piecing together the required components will take time and experimenting.

Tied to performance is a question about the amount of geospatial data that can be passed from the database to the client browser expeditiously. At the time of writing this report, the AWS Lambda Developer Guide listed specifications for configuring functions in the AWS Lambda console that include memory with a maximum limit of 3 gigabytes. (Amazon Web Services, 2020a) Testing with various dataset sizes will be required to determine the optimal data transmission capacity. The optimal size may also influence the map view on the user interface. Map zoom levels or number or features that can be queried at one time could be impacted by the serverless capacity limits.

Experience teaches software developers to expect the unexpected and be prepared to solve problems as they arise. This project is not atypical in that sense. Some aspects of it have already been proven. Other parts of the final solution have not been solved. Also expected are changes in the technology as development progresses. The specifications for AWS Lambda functions could change requiring code rewriting or application architecture redesign.

These challenges relate to the working web-mapping application prototype itself. Other challenges exist around migrating the existing data and processes to the new cloud-based solution. Hopefully, the Agile approach employed will address some of these concerns during the development cycles. It is possible that any final solution will include a number of utility functions that work outside of the web-mapping application to prepare data or administer the application.

# Anticipated Results

Confidence that the project objective is realistic comes from previous experience and the presence of documentation for building serverless solutions. The anticipated result of the project is a prototype version of a web-mapping application built using open-source technology and deployed in a serverless fashion using AWS cloud services. The web-mapping application will overlay thematic data from a database without using common map layer services such as GeoServer or ArcGIS Server. This will be the most significant accomplishment of the project. While the performance and usability of the solution is unknown, the project will answer questions related to the feasibility of serverless web-mapping applications.

# Project Timeline

The project timeline takes into account necessary application features that must be completed and labels them as milestones. Because some of these features have no precedent, the time to complete them is unknown. Estimates are based on experience with similar technology or projects. The dates assigned to the milestones are introduced to maintain progress and forward momentum toward the final solution. At least one required feature can be developed as part of remaining coursework.

## Remaining Coursework

One class outside of project work remains. The class is GEOG 868: Spatial Database Management. The course will be taken during the 2020 summer semester. The syllabus for this course includes a final project that requires the development of a spatial database to solve a problem of the student’s choosing. The option to use PostgreSQL with PostGIS is allowed. This class project will be used to at least begin to solidify the database layer storing both vector and raster data using AWS services. The syllabus also allows students to develop a “front-end” to the database. If time allows, pieces of the serverless architecture will be attempted and the final project for this class will be hosted using AWS services. (Detwiler & Sloan, 2020)

## Project Milestones

Table 3 provides the general timeline for project feature completion, capstone deliverables and program requirements.

**Table 3. Project Timeline**

|  |  |
| --- | --- |
| **Date** | **Milestone** |
| July 22, 2020 | Spatial Database Management course ends |
| August 26, 2020 | Database layer of web-mapping application finalized |
| October 14, 2020 | Serverless web-mapping application overlaying at least one thematic layer from database for visualization functioning |
| October 28, 2020 | User-interface and serverless functions complete for drawing new vector boundaries for database storage |
| November 11, 2020 | User-interface and serverless functions for uploading and storing raster data |
| November 25, 2020 | User-interface and serverless functions for querying data spatially and by attributes |
| December 9, 2020 | User-interface and serverless functions for updating spatial and attribute data |
| December 23, 2020 | Temporal visualization features available |
| January 2021 | Submit intent to graduate |
| January – April, 2021 | Continued improvements to web-mapping application, presentation preparation, final project paper |
| May 2021 | Project presentation @ TNRIS GeoRodeo |
| \*August 2021 | Submit intent to graduate |
| \*October 2021 | Project presentation @ TNRIS GIS Forum |
| \*November 2021 | Project presentation @ GIS Day events |

\*Denotes possible later dates

## Possible Presentation Venues

Texas Natural Resources Information System (TNRIS) hosts two events each year for the Austin GIS community. The GeoRodeo targeting the geospatial developer community is usually held in early May. The annual Texas GIS Forum is the premiere statewide GIS gathering hosted in late October and spanning the course of a week with workshops and multiple days of presentations. Either of these in 2021 are options. The May GeoRodeo is part of the planned timeline.

Later presentation options aside from the 2021 Texas GIS Forum are November GIS Day events in Austin or at Pennsylvania State University.

The conference dates for the “Free and Open Source Software for Geospatial” (FOSS4G) events are not posted beyond August 2020. North America events in 2021 could be options.

Conferences specific to agriculture technology that have dates scheduled:

* World Agri-Tech Innovation Summit – San Francisco, March 9-10, 2021
* InfoAg – St. Louis, July 2021

The ideal presentation venue will be held in between March and early May of 2021 allowing for a Spring or Summer 2021 graduation.

# References

Agafonkin, V., & OpenStreetMap contributors. (2019a). *Leaflet*. https://leafletjs.com/

Agafonkin, V., & OpenStreetMap contributors. (2019b). *Using GeoJSON with Leaflet*. Leaflet. https://leafletjs.com/examples/geojson/

Agile Alliance. (2020). *What is Agile?* https://www.agilealliance.org/agile101/

AgriXP. (2020). *AgriXP*. https://agrixp.com/compare?lang=en

Amazon Web Services. (2020a). *AWS Lambda Developer Guide*. AWS Documentation. https://docs.aws.amazon.com/lambda/latest/dg/welcome.html

Amazon Web Services. (2020b). *AWS Step Functions*. https://aws.amazon.com/step-functions/

Amazon Web Services. (2020c). *Build a Serverless Web Application*. Projects on AWS. https://aws.amazon.com/getting-started/projects/build-serverless-web-app-lambda-apigateway-s3-dynamodb-cognito

Amazon Web Services. (2020d). *Cloud Products*.

Amazon Web Services. (2020e). *Serverless Build and run applications without thinking about servers*. Serverless Computing. https://aws.amazon.com/serverless/

Amazon Web Services. (2020f). *What is cloud computing?* https://aws.amazon.com/what-is-cloud-computing/

Applied Geographics. (2020). *2020 EDITION Aerial Imagery Provider Comparison Guide*. https://www.mapgeo.io/aerial-imagery-comparison-guide

Asay, M. (2020). *How open source might prove helpful during the coronavirus pandemic*. TechRepublic. https://www.techrepublic.com/article/how-open-source-might-prove-helpful-during-the-coronavirus-pandemic/

Axis Maps. (2017). What is a web map? *Cartography Guide*. https://www.axismaps.com/guide/web/what-is-a-web-map/

Binh, T. T., Tien, P. D., Long, D. T., Truong, H. L., & Thanh, N. N. P. (2019). Building agricultural information system in An Giang province. *An Giang University Journal of Science*, *6*, 89–99.

Bordogna, G., Kliment, T., Frigerio, L., Brivio, P. A., Crema, A., Stroppiana, D., Boschetti, M., & Sterlacchini, S. (2016). A spatial data infrastructure integrating multisource heterogeneous geospatial data and time series: A study case in agriculture. *ISPRS International Journal of Geo-Information*, *5*(5), 1–27. https://doi.org/10.3390/ijgi5050073

Buytaert, D. (2020). *Is Open Source recession-proof?* https://dri.es/is-open-source-recession-proof

Carey, S., & Macaulay, T. (2019). *What are the advantages of open source software in the enterprise?* COMPUTERWORLD. https://www.computerworld.com/article/3412269/what-are-the-advantages-of-open-source-software-in-the-enterprise-.html

Castro, P., Ishakian, V., Muthusamy, V., & Slominski, A. (2019). The rise of serverless computing. *Communications of the ACM*, *62*(12), 44–54. https://doi.org/10.1145/3368454

Cloudflare. (2020a). *What Is Serverless Computing? | Serverless Definition*. https://www.cloudflare.com/learning/serverless/what-is-serverless/

Cloudflare. (2020b). *What Is Vendor Lock-In? | Vendor Lock-In and Cloud Computing*. https://www.cloudflare.com/learning/cloud/what-is-vendor-lock-in/

Croptracker. (2020). *Croptracker Farm Management Software*.

Cubbage, S. (2019). Farm Journal AgTech Expo. *Data Survival Guide 2.0*.

Cubbage, S. (2020). *Ag Could Have A Wild Ride In The Next Decade*. Farm Journal AgWeb. https://www.agweb.com/article/steve-cubbage-ag-could-have-wild-ride-next-decade

Detwiler, J., & Sloan, J. (2020). *Spatial Database Management*. John A. Dutton e-Education Institute, College of Earth and Mineral Sciences, The Pennsylvania State University. https://www.e-education.psu.edu/spatialdb/

Dhanapal, Y., Mateer, C., & Timmons Group. (2019). Enhancing GIS with Cloud Technology and Serverless Computing. *GeoCon 2019 Virginia GIS Conference Hosted by VAMLIS*.

Dignan, L. (2020). *Top cloud providers in 2020: AWS, Microsoft Azure, and Google Cloud, hybrid, SaaS players*. ZDNet. https://www.zdnet.com/article/the-top-cloud-providers-of-2020-aws-microsoft-azure-google-cloud-hybrid-saas/

Dongoski, R., Selck, A., & Buie, B. (2017). Digital agriculture: influences, trends, and opportunities among ag retailers. *EY*. http://www.ey.com/Publication/vwLUAssets/ey-digital-agriculture-trends-ag--retail/$FILE/ey-digital-agriculture-trends-ag-retail.pdf

DroneDeploy. (n.d.). *AI-Powered Drone Data Processing*. https://www.dronedeploy.com/product/photogrammetry/

ESRI. (n.d.). *What is GIS*.

ESRI. (2019). *A framework for deploying web GIS applications*. ArcGIS Server. https://enterprise.arcgis.com/en/server/latest/create-web-apps/windows/a-framework-for-deploying-web-gis-applications.htm

EY Global. (2017). *How digital agriculture and big data will help to feed a growing world*. EY. https://www.ey.com/en\_gl/advisory/how-digital-agriculture-and-big-data-will-help-to-feed-a-growing-world

Farley, A. (2019a). *GEOG 585 Open Web Mapping - FOSS Web Mapping Application*. The Pennsylvania State University. http://personal.psu.edu/alf33/

Farley, A. (2019b). *GEOG 865 Cloud and Server GIS - Open-Source GIS in the Cloud*. The Pennsylvania State University. http://personal.psu.edu/alf33/

Farley, J. (2020a). *Interview*.

Farley, J. (2020b). *Interview*. http://www.farleyfarmsupply.com/

Farmobile Editors. (2019). *Trip 3: Extra Mile – Tracing food … how agdata aids growers*. Farmobile. https://www.farmobile.com/trip-3-the-extra-mile-steve-cubbage-the-biggest-risk/

G2. (2020a). *Best Crop Management Software*. Best Agriculture Software. https://www.g2.com/categories/crop-management

G2. (2020b). *Best GIS Software*. https://www.g2.com/categories/gis

GISGeography. (2020). *30 GIS Software Applications [2020 Rankings]*. GISGeography.Com. https://gisgeography.com/mapping-out-gis-software-landscape/

GitHub. (2020). *GitHub*. https://github.com/

Google. (2019). *Google Earth Engine*. https://earthengine.google.com/

Google. (2020a). *Google Cloud for AWS Professionals*. https://cloud.google.com/docs/compare/aws

Google. (2020b). *Google Imagery Sales FAQ*. https://support.google.com/mapsdata/topic/6250082?hl=en&ref\_topic=6246343

Google. (2020c). *Serverless computing*. https://cloud.google.com/serverless

Granular. (2020). *Granular*.

Griffin, T. (2019). Farm Journal AgTech Expo. *Blockchain: The Big Picture and the Nitty Gritty*.

Helendi, A. (2018). *How Much Money Can You Save By Going Serverless?* Dashbird. https://dashbird.io/blog/saving-money-switching-serverless/

Heneghan, J. (2019). Farm Journal AgTech Expo. *Sort Through The Data Overload*. https://www.farmjournalagtechexpo.com/schedule

Jacoby, D., Wynden, A., Gao, B., Giannini, F. C., Costa, M., & Coady, Y. (2019). Geospatial Computing Collaborations. *2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 1–8. https://doi.org/10.1109/pacrim47961.2019.8985053

Javadnejad, F., Gillins, D. T., Higgins, C. C., & Gillins, M. N. (2017). BridgeDex: Proposed Web GIS Platform for Managing and Interrogating Multiyear and Multiscale Bridge-Inspection Images. *Journal of Computing in Civil Engineering*, *31*(6). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000710

Jonas, E., Krauth, K., Schleier-smith, J., Gonzalez, J. E., Patterson, D. A., & Berkeley, U. C. (2019). *Cloud Programming Simplified : A Berkeley View on Serverless Computing*. 1–33.

Jones, A. (2018). *Stateful vs. Stateless Architecture Overview*. Bizety.Com. https://www.bizety.com/2018/08/21/stateful-vs-stateless-architecture-overview/

Land IQ. (2018). *Land IQ Agriculture*.

Matteson, S. (2018). *How to decide if open source or proprietary software solutions are best for your business*. TechRepublic. https://www.techrepublic.com/article/how-to-decide-if-open-source-or-proprietary-software-solutions-are-best-for-your-business/

Mello, U., & Raghavan, S. (2018). *Bringing the power of Watson to farmers*. IBM Research Blog. https://www.ibm.com/blogs/research/2018/09/smarter-farms-agriculture/

Microsoft. (2019). *Serverless web application on Azure*. Microsoft Azure. https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/serverless/web-app

Microsoft. (2020a). *Azure FarmBeats*. https://www.microsoft.com/en-in/campaign/azure-farmbeats/

Microsoft. (2020b). *What is cloud computing*. Microsoft Azure. https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/

Mozilla and individual contributors. (2020). *Client-side storage*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side\_web\_APIs/Client-side\_storage

Nolle, T. (2019). *The key differences between stateless and stateful microservices*. The Enterprise Architect’s Guide to Application State Management. https://searchapparchitecture.techtarget.com/tip/The-key-differences-between-stateless-and-stateful-microservices

Orlowski, A. (2019). *Open Source Technology Could Be a Boon to Farmers*. Civil Eats. https://civileats.com/2019/12/02/open-source-technology-could-be-a-boon-to-farmers/

OSGeo. (2020). *The Open Source Geospatial Foundation*. https://www.osgeo.org/

Pascaul, M., Alves, E., Almeida, T. De, França, G. S. De, Roig, H., & Holanda, M. (2012). An Architecture for Geographic Information Systems on the Web - webGIS. *GEOProcessing 2012 : The Fourth International Conference on Advanced Geographic Information Systems, Applications, and Services*, *c*, 209–214.

Paul, S. (2018). *How GIS is enabling the agricultural sector*. Geospatial World. https://www.geospatialworld.net/blogs/gis-in-agriculture/

Penn State Cooperative Extension, Penn State Geospatial Technology Program, L. A. L., & Day, R. (2011). *PAOneStop Farm Mapping and E&S Planning System*. https://paonestop.psu.edu/

Planet Labs. (2020). *Planet Basemaps*. https://www.planet.com/products/basemap/

Planet Labs Inc. (2020). *Agriculture*. https://www.planet.com/markets/monitoring-for-precision-agriculture/

PostGIS Project Steering Committee. (2020). *ST\_AsGeoJSON*. PostGIS 3.0.2dev Manual. https://postgis.net/docs/ST\_AsGeoJSON.html

Predictive Analytics Today. (2020). *Top 9 Farm Management Software*. Farm Management Software. https://www.predictiveanalyticstoday.com/top-farm-management-software/

Quinn, S. (2019). *Open Web Mapping*. John A. Dutton e-Education Institute, College of Earth and Mineral Sciences, The Pennsylvania State University. https://www.e-education.psu.edu/geog585/

Quinn, S., Baxter, R., & O’Brien, J. (2019). *Cloud and Server GIS*. John A. Dutton e-Education Institute, College of Earth and Mineral Sciences, The Pennsylvania State University. https://www.e-education.psu.edu/geog865/

Rajan, R. A. P. (2018). Serverless Architecture - A Revolution in Cloud Computing. *2018 10th International Conference on Advanced Computing, ICoAC 2018*, 88–93. https://doi.org/10.1109/ICoAC44903.2018.8939081

Red Hat. (2020a). *The Open Farmer*. Open Source Stories. https://www.redhat.com/en/open-source-stories/farming-for-the-future/open-farmer

Red Hat. (2020b). *What is open source?* https://www.redhat.com/en/topics/open-source/what-is-open-source

Reynolds, D., Ball, J., Bauer, A., Davey, R., Griffiths, S., & Zhou, J. (2019). *CropSight: a scalable and open-source information management system for distributed plant phenotyping and IoT-based crop management | GigaScience | Oxford Academic*. GigaScience. https://doi.org/10.1093/gigascience/giz009

Rosenberg, J., & Mateos, A. (2011). *The Cloud at Your Service*. Manning Publications Co.

Rouse, K. L. (2020). *Meat Inspection Act of 1906*. Encyclopedia Britannica. https://www.britannica.com/topic/Meat-Inspection-Act

Ruppert, L. (2019). Farm Journal AgTech Expo. *The Retailers’ Role in Adopting AgTech*. https://www.farmjournalagtechexpo.com/schedule

Samdan, E. (2019). *Containers vs. Serverless*. DZone. https://dzone.com/articles/containers-vs-serverless

Sandoval, K. (2017). *Defining Stateful vs Stateless Web Services*. Nordic APIs. https://nordicapis.com/defining-stateful-vs-stateless-web-services/

Santos, E. (2020). *Considerations on Deploying Geoprocessing Workflows as Serverless Functions*.

Sarjeel, Y. (2019). *Getting it right between EC2, Fargate and Lambda*. Medium. https://medium.com/thundra/getting-it-right-between-ec2-fargate-and-lambda-bb42220b8c79

Skernivitz, T. (2019). *How Ag Retailers Are Adding Value to Real-Time Data*. CropLife. https://www.croplife.com/precision/how-ag-retailers-are-adding-value-to-real-time-data/

Software Connect. (2020). *Farm Management Software*. https://softwareconnect.com/farm-management/#products

Stenta, M. (2020). *farmOS*. https://farmos.org/

Taylor, T. (2019). *3 serverless development strategies for stateful applications*. The Enterprise Architect’s Guide to Application State Management. https://searchapparchitecture.techtarget.com/tip/3-serverless-development-strategies-for-stateful-applications

Technopedia. (2020). *Web Mapping*. https://www.techopedia.com/definition/15584/web-mapping

The Climate Corporation. (2020). *Climate FieldView*. https://climate.com/

Tierra Plan. (2019). *FIELD ANALYST*. https://tierraplan.com/projects/field-analyst/

Trimble. (2020). *What Is Precision Ag?* https://agriculture.trimble.com/blog/what-is-precision-ag/

Tutorials Point. (2020). *SDLC - Software Prototype Model*. SDLC Tutorial. https://www.tutorialspoint.com/sdlc/sdlc\_software\_prototyping.htm

VMWare Tanzu. (2020). *Cloud native applications: Ship faster, reduce risk, and grow your business*. https://tanzu.vmware.com/cloud-native

Vogt, W. (2020). *Changing customer impacts ag retail*. https://www.farmprogress.com/technology/changing-customer-impacts-ag-retail

Watts, S., & Raza, M. (2019). *SaaS vs PaaS vs IaaS: What’s The Difference and How To Choose*. Bmc Blogs. https://www.bmc.com/blogs/saas-vs-paas-vs-iaas-whats-the-difference-and-how-to-choose/

Whitney, B. (2019). *Interview*. https://www.whitneyconsultinggroup.com/

Whole Foods Market. (2020). *Whole Foods Quality Standards*. https://www.wholefoodsmarket.com/quality-standards

Xia, D., Xie, X., & Xu, Y. (2009). Web GIS server solutions using open-source software. *OSSC-2009 - Proceedings of 2009 IEEE International Workshop on Open-Source Software for Scientific Computation*, 135–138. https://doi.org/10.1109/OSSC.2009.5416738