Implementing a GIS Laboratory in a Cloud Based Environment
John Maloney, Senior Lead Software Developer, S & P Global, Billings, MT

ABSTRACT TEXT: Scalability and Performance have always been the challenging factors for Geographical Information Systems, and with the distributed nature of the Cloud there needs to be a solid analysis of how a GIS fits into that infrastructure. The differentiating factors of a GIS when compared to a typical application are the rendering engine and the geospatial analysis engine. These two engines are center-pieces of my project, delivering them from a distributed computing infrastructure is the goal.

Typically, the architecture of a system was defined as vertically stacked layers that communicated up and down through the layers by having a reference to the layer above and below it. The machine infrastructure for an application was made up of a handful of instances that had hardware capable of executing at the peak load needed (estimated) for the application. This allowed for the application infrastructure to respond efficiently when the traffic (load) was highest. This required a hefty up-front investment into the infrastructure for a machine that may only hit peak performance for a small percentage of its active time.

The purpose of this project is an academic exercise into the infrastructure requirements and the architectural concepts that make up the foundation of a Geographical Information System. With Cloud computing the nature of applications infrastructure and architecture has dramatically changed. Cloud-based offerings have made their way into the main providers of GIS (e.g. ESRI, MapBox, Carto, etc) and it is worthwhile from a software engineering and development perspective to replicate the nature of these offering so as to understand the benefits and the obstacles to providing custom GIS software to the market. While a case can be made that this constitutes the “reinventing of the wheel”, it is also well understood that custom GIS is a valid business model and relying on large third party providers for software is not always the best approach. One can simply look into the open-source community to see that GIS has its foundation in freedom to raise systems out of open standards rather than commercial software.

Learning Objective1: The overall learning objective of my proposal is to allow the attendee to gain insight into the thought processes and architectural decisions that were made to develop a custom GIS based in a disconnected and scalable environment (i.e. the Cloud). These insights would include:

1. How is the rendering engine hosted in the Cloud, how does it get constructed during peak traffic and deconstructed during non-peak traffic times.
2. How does the rendering engine receive requests and distribute data (e.g. Tiles) in a volatile environment.
3. How are the underlying data caches built and used by the rendering engine(s)?
4. How do different clients interact with the rendering engine in a disconnected environment?
5. How is the overall health of the system reported on and how can the development operations managers use this information.

Learning Objective2: The Second objective is to allow the user to have access to the system through GitHub. I will show them how to interact with the underlying code that makes up the entirety of the GIS developed for this project. The end goal would be the GIS developers could use the open source code to learn how to generate their own custom GIS system that was designed to live in the Cloud.

Case Studies: The case study consists of the project that is delivered through this endeavor. This project lays the groundwork for a basic GIS that was designed with the benefits and pitfalls of the Cloud in mind. The goal of this project is not only to define the base framework but to also allow for the growth of that framework by making clear the opportunities to add modular functionality to the system.

Skills: Given the audience would be GIS developers and some decision making folks the skills would be a deeper understanding of where to begin with the development of a custom GIS and how to make design decisions around scalability and modularity. The end take away would be a stronger ability to address the development of a custom GIS which can exist in the Cloud.

ArcGIS Server Architectures in the Cloud (AWS)
Mark Korver, MCP/DA, Geospatial Lead - Solution Architecture Specialist Team, Amazon Web Services, Seattle, WA

ABSTRACT TEXT: You can use ArcGIS Enterprise Amazon Machine Images (AMIs), which are virtual machine images prepared, by ESRI, to launch several different architectures on the Amazon Cloud. ESRI makes this easier by providing AWS Cloud Formation templates, automation scripts, that take the heavy lifting out of deploying ArcGIS, including the creation of VPCs, ELBs and licensing.

This talk will present deploying ArcGIS from the perspective of an AWS Solution Architect, incorporating new AWS services such as Application Load Balancers and Amazon FSx storage and Aurora PostGreSQL.