Data-intensive spatial data platform

Abstract:
The urge for data-intensive spatial data platforms came from the large growth of spatial data and the high computational complexity of spatial data queries. For instance, trajectory data of a fleet of one thousand cars, where each data point comprises time, latitude and longitude, will require 2 GB of storage space each day. Performing complex geometric queries to these large scale data can be very expensive to process. Taking a typical shortest path query in ride-hailing applications (e.g: Uber) as an example, when a ride request appears, the system has to compute the shortest path of all drivers from the passenger’s location. This requires large computing memory and storage space. These two problems require robust architecture that can query spatial data on a large scale. If we take budget constraint as a consideration, an additional requirement would be to support queries on a cost effective architecture such as cloud environments. In this short tutorial, we will explain spatial data platforms in three different stacks: laptop, cloud computing, and supercomputing stack, along with the layers in each stack.

Keywords:
Spatial Data Platform, Geographic Information System (GIS) Software, Data Platform Stacks, ESRI1 GIS tools for Hadoop, CyberGIS, Hadoop, Spark, Cloud Computing, Distributed Systems

Basic Concepts:

1. Data platform stacks
The three main stacks are Laptop stack, Cloud Computing stack and Supercomputing stack.

Laptop stack is the stack we use in our everyday life. It consists of laptops and Desktop PCs.

Cloud Computing stack is usually utilized when you have limited funding and are usually used for performing tasks that do not require extremely high performance. This ecosystem can be scaled easily by adding more nodes (computer used as a server) to the cluster. These nodes can be rented from on-demand cloud computing platforms like Amazon Web Services (AWS) and Google Cloud Platform (GCP).

Supercomputing stack is utilized when you have a large funding and require high performance. They are usually used for performing tasks with high computational requirements like matrix

1: ESRI: Environmental Systems Research Institute
multiplication. It is not as easy to scale in this ecosystem as compared to the cloud computing stack.

The Cloud Computing and Supercomputing stacks are usually used for high memory and high throughput requirements.

In the following sections, each stack will be explained in detail. The taxonomies mentioned in the below figure are explained in Section 2 by taking Laptop Stack as an example.

2. Laptop Stack

2.1. Hardware taxonomy
Cluster hardware refers to the physical components. The most common example is laptop and personal computer (PC).

2.2. System Software taxonomy
System software is a software that runs on top of the hardware and provides a platform for other softwares to run on it. The most common example of system software in laptop stack is Operating systems like Linux, Windows and macOS.
2.3. Middleware and management taxonomy
Middleware is a software that provides services to software applications beyond those provided by the operating system, like managing network based requests. Middleware allows the building of interoperable, reusable software components (like services) and the integration of software components. Examples in Laptop Stack include Representational state transfer (REST API).

2.4. Application level taxonomy
Spatial data analysis is conducted in the application level. In the Laptop Stack, R libraries and ArcGis are commonly used.

2.4.1. ArcGIS
ArcGIS is a platform provided by ESRI (Environmental Systems Research Institute) to create, manage, analyse and share spatial data. It can be deployed on-premise (Desktop version) or in the cloud (AWS, Microsoft Azure or ESRI managed cloud services).

Figure 2: A snapshot of ArcGIS Pro [6]

3. Cloud Computing Stack

3.1. Hardware taxonomy
Cloud Computing stack is based on hardwares with cost and capacity optimization like Commodity Racks which are cheap and failure-prone.

3.2. System Software taxonomy
The system softwares used is the same as the Laptop stack which is Operating systems like Linux, Windows and macOS.
3.3. Middleware and management taxonomy
Some of the important middleware for the Cloud Computing stack are explained below:

3.3.1. Hadoop:
Hadoop is a collection of open-source software utilities that facilitate processing data in parallel across clusters containing a large number of nodes. It develops software for reliable, scalable and distributed computing. Rather than relying on hardware to be fault-tolerant, hadoop is designed to detect and handle failures in the application level. Thus, delivering a fault tolerant service on top of a cluster of computers which are error-prone. The two most important components of Hadoop are:

(i) Hadoop Distributed File System (HDFS)
HDFS is a distributed file system designed to run on commodity hardware (error-prone and inexpensive). It is used to manage a large number of large files distributed across the nodes of a cluster with replication (copying the same data on different nodes). It is immutable i.e, it follows the Write Once Read Many model.

(ii) Map Reduce
Map Reduce is a programming platform for processing a large number of data in parallel in a reliable and fault tolerant manner. A Map Reduce job splits the input data into independent chunks which are processed by map tasks that are run in parallel. The framework sorts the outputs of the maps and inputs them to reduce tasks. Reduce tasks usually use the outputs of map tasks to get the final output of the job. Map Reduce cannot be used for interactive queries as it is extremely slow. Also, it saves and reads from disk.

3.3.2. Spark:
Spark can be briefly defined as a unified computing engine and a set of libraries for parallel data processing on computer clusters. Unified implies that Spark can support a wide range of data analytics tasks, ranging from SQL queries to Machine Learning and Streaming applications. Spark is just a computing engine and doesn’t support permanent storage but it can be used with a large number of persistent storage systems (eg: cloud storage systems like Amazon S3, distributed file systems like Hadoop and NoSQL databases like Cassandra).
Spark is faster than Map Reduce as it saves data on memory but this leads to high memory needs. A few important concepts in Spark are:

(i) DataFrame
DataFrame is a distributed collection of data organised in the form of a table.

(ii) Resilient Distributed Dataset (RDD)
RDD is an immutable distributed collection of objects which is a fundamental data structure of Spark.
(iii) **SparkSQL**
SparkSQL is a module in Spark for structured data processing. It can be used to execute SQL queries.

### 3.3.3. Hive
Hive a data warehousing framework built on top of Hadoop for supporting query and analysis of data. Hive provides a SQL like interface called Hive Query Language (HiveQL) to query data stored in various databases that integrate with Hadoop.

### 3.4. Application level taxonomy
Two of the most common spatial data platforms in the cloud computing stack are Simba and ESRI GIS tools for Hadoop.

#### 3.4.1 Spatial In-Memory Big data Analytics (Simba)
Simba is a spark based system which offers scalable and efficient in-memory spatial query processing and analytics for big spatial data. It runs on commodity hardware. It extends the SparkSQL engine to support SQL and dataframe based spatial queries. For efficient processing and analytics, it constructs indexes over RDDs. Simba's query optimizer leverages these indexes and novel spatial-aware optimizations. Simba is useful for programmers as they can optimize low-level concepts according to their use case.

#### 3.4.2. ESRI GIS tools for Hadoop
ESRI GIS tools for Hadoop is an open source toolkit that leverages Hadoop's Map Reduce framework to perform big spatial data analytics. GIS tools integrates three other projects:

1. **ESRI Geometry API for Java**
   This API is the main building block for the toolkit as they enable spatial data processing in 3-rd party data processing solutions. For example, can be used for spatial data processing in the Hadoop system. The three main methods provided by it help to:
   a. create geometries directly with the API, or by importing from supported formats: JSON, WKT, and Shape
   b. perform spatial operations: union, difference, intersect, clip, cut, and buffer
   c. Test topological relationships: equals, within, contains, crosses, and touches

2. **Spatial Framework for Hadoop**
   This framework enables using Hadoop data processing systems for spatial data analysis. Its most important component is a set of user defined functions (UDFs) that extend Hive and are built upon capabilities of the Esri Geometry API.

3. **Geoprocessing tools for Hadoop**
   These tools facilitate connecting data between Hadoop and ArcGIS, submitting workflow jobs, and converting data to and from JSON. The main aim of these tools are to transfer Hadoop results to ArcGIS for visualization.
The distinguishing factor of ESRI tools for Hadoop is that it optimizes performance across the cluster by combining spatial indexing, data load balancing and data clustering.

### 3.4.2.1. Spatial Indexing
Spatial indexing is a data structure that allows for efficiently accessing spatial objects. Without indexing, any query would require a sequential scan of every record in the database. The spatial indexing in GIS tools for Hadoop leverages the Hadoop MapReduce framework and uses quad-tree (a tree data structure in which every internal node has exactly 4 children) as its index structure.

### 2.3.4.2. Spatial Ordering
Spatial ordering is reorganizing the data to correspond to the ordering found within the index.

### 3.4.3. Clustering
A MapReduce job is executed inorder to balance the size of data in partitions. The partitions are allowed to have jagged shapes and contain multiple disjoint pieces. A buffer region is built around each partition, which contains spatial objects within a user-specified buffer radius. This helps optimize proximity queries (like range and k-NN queries) near partition boundaries as the buffer region allows us to avoid reading from multiple partitions.

### 3.4.3. Simba versus ESRI:
Simba is faster than ESRI GIS tools for Hadoop as Simba stores data in memory unlike ESRI tools for Hadoop which uses disk as it is based on MapReduce.
4. Supercomputing Stack: CyberGIS

CyberGIS is a new generation of GIS. Most conventional GIS tools rely on single-CPU architecture, while cyberGIS utilizes cyberinfrastructure (CI), which allow the access to parallel supercomputing architecture, hence enabling stronger computational power and more spatial analysis capabilities.

4.1. Hardware taxonomy

CyberGIS utilizes advanced CI. The term was first coined by the US National Science Foundation (NSF) which refers to comprehensive information technology infrastructure that provides integrative access to interrelated computational components, including high-performance computers, data and information resources, along with an interoperable suite of software [1].

CI architecture offers three integrated computing modalities: high performance computers equipped with GPUs; data intensive computing with Hadoop and Spark; and cloud computing. The two CI environments commonly used in CyberGIS softwares are: ROGER and the Extreme Science and Engineering Discovery Environment (XSEDE) [10] supercomputers.

4.2. Middleware and management taxonomy

Spatial Middleware hides the complexity of accessing and integrating with CI resources from users. It handles integration of different GIS applications to CI resources; manages the required computational resource and data requirements; and also acts as the key enabler to CyberGIS Gateway applications.

Unlike the generic CI middleware that mostly focuses on management of CI resources, Spatial middleware is designed to be spatially aware. For instance, it models the computational intensity required to perform the spatial analysis tasks in order to allocate and use CI resources more effectively. Spatial middleware also harnesses different types of CI resources and services (e.g: high performance, high-throughput, visualization, data-intensive computing).

An example of spatial middleware, GISolve provides a suite of open service APIs that defines a set of REST Web Service interfaces, that encompass the following functionalities:

1. Application Integration APIs provide integration mechanisms with cyberGIS software environments (i.e: cyberGIS gateway and toolkit). It handles the deployment of these softwares on CI resources, and publishes them into the cyberGIS environment.
2. Security APIs provide token-based authentication for software integration.
3. Computation APIs manage the computation, data and visualization resources required from the CI resources to support cyberGIS analytics capabilities.
4.3. Application Level taxonomy

There are two types of CyberGIS software, which can be an independent entity or complementary modalities.

1. Gateway: User Interface online component
2. Toolkit: Open source loosely coupled geospatial software modules available for the community to collectively improve and share.

4.3.1. CyberGIS Gateway

Gateway [18] provides easy access to online cyberGIS services (i.e: spatial analysis and visualization tools) through interactive User Interface (UI), and hides the complexity of accessing the CI resources to users.

4.3.1.1. Gateway design and components

Gateway is built using rich-client web technologies (HTML5 and ExtJS). It’s access to the underlying CI infrastructure is handled by the spatial middleware, hence making it accessible to vast science communities (e.g: users from geography or public health background). Gateway’s UI is designed using the Model View Controller (MVC) architecture and each application is built as a standalone MVC application [2].

Gateway can interact with external web services hosted by other organizations. They provide open Application Programming Interfaces (APIs) to allow developers and communities to interact with their backend service implementations. The APIs are implemented as REST Services and currently support PHP, Perl, Python and Java.

4.3.1.2. Accessing spatial data and analysis tools through CyberGIS Gateway

CyberGIS provides tools for users to do spatial analysis and access spatial data. Applications like Flumapper and TopoLens provide spatial analysis tools for different domains (e.g: public health) and access to spatial data. Due to space limitation, this tutorial will briefly discuss some of the important applications. The full list of gateway applications and their details that can be found in the link provided in the beginning of section 5.1.

(i) Flumapper

This application provides an exploratory tool mainly for public health researchers, to capture the movement of Influenza Like illnesses (ILI) activities in North America [3]. It uses Twitter streaming data to compute the spatial pattern and flow mapping of the illness.
(ii) TopoLens
TopoLens is a tool to access and process high resolution topographic (i.e: raster elevation) data and allows support for community data services, like data customization and sharing. It provides pre-computation of high resolution elevation datasets at State and Hydrologic Unit Code (HUC) Regional levels, hence allowing the data to be effectively consumed by users. It also allows on-demand data and map services powered by CI infrastructures to efficiently produce datasets that are customized based on user’s request [4].

(iii) CyberGIS Jupyter
This application enhanced the standard Jupyter notebook to include CyberGIS analytics capabilities, such as computation management and raster data access [5]. This allows users to better customize their applications with the capabilities provided by CyberGIS. Users can also choose to run their analytics in the Notebook within a container or through external CI’s High Performance Computing (HPC) resources.

4.3.2. CyberGIS Toolkit
CyberGIS toolkit is a set of scalable loosely coupled geospatial software components. It serves the same purpose as the gateway, with an extended capability of being able to be scaled by users. To some extent it is similar to GitHub libraries. Both are open source, ready to be used pieces of software, and the users have flexibility to customize and develop it further.
to suit their needs. CyberGIS Toolkit can be both programming libraries or use-ready applications.

CyberGIS toolkit has a continuous integration process that has to be followed through each integration of an individual geospatial code. During evaluation and testing, code developers are provided with cloud-based development virtual machines along with the customized library support for their code.

When the code is ready for integration, it has to go through both portability and scalability tests. In portability tests, the code is executed in different combinations of operating systems, architecture, and software library versions. The scalability test aims to assess the code's computational requirements and to identify possible performance bottlenecks (i.e: memory requirement, network), to make sure it is compatible with HPC resources and scalable for large scale input.

CyberGIS toolkits can be found on the following:
1. The Extreme Science and Engineering Discovery Environment (XSEDE) [7]
2. Blue Waters [8]
3. ROGER [9]

Learning Objectives

1. Explain spatial data platforms in different data platform stacks (i.e: laptop, cloud computing and supercomputing stack).
2. Explain the layers of each stack:
   a. Hardware taxonomy
   b. System software taxonomy (for laptop and cloud computing)
   c. Middleware and management taxonomy
   d. Application layer taxonomy
3. Briefly discuss a few examples of spatial applications in the Cloud Computing and Supercomputing Stacks, namely Simba, ESRI tools for Hadoop and CyberGIS.

Instructional Assessment Questions

1. Which of the following is not middleware software?
   a. Hadoop
   b. Windows
   c. Map Reduce
   d. Spark
2. Which of the following is/are features of commodity racks?
   a. Cost effective
   b. Failure-prone
3. Can Hadoop files be modified after writing?
   a. Yes
   b. No

4. Which is faster?
   a. ESRI GIS tools for Hadoop
   b. Simba

5. What is the difference between CyberGIS Gateway and CyberGIS Toolkit?
   a. Gateway manages integration with CI resources; GIS toolkit offers loosely coupled GIS software modules that can be extended.
   b. Gateway offers GIS software as Web user interfaces; GIS toolkit offers loosely coupled GIS software modules that can be extended.
   c. Gateway offers loosely coupled GIS software modules that can be extended; GIS toolkit manages integration with CI resources.
   d. Gateway offers loosely coupled GIS software modules that can be extended; GIS toolkit GIS software as Web user interfaces while toolkit offers.

6. What is the equivalent for CyberGIS’ ROGER supercomputers in the Cloud computing stack?
   a. Hadoop
   b. Spark
   c. Hive
   d. Commodity Racks

(Answers: 1-b, 2-d, 3-b, 4-b, 5-b, 6-d)

Additional Resources


References

5. Dong Xie, Feifei Li, Bin Yao, Gefei Li, Liang Zhou, and Minyi Guo, “Simba: Efficient In-Memory Spatial Analytics”
11. https://www.xsede.org/
12. https://wiki.ncsa.illinois.edu/display/ROGER