**STORAGE AND ORGANIZATION OF LARGE GEOSPATIAL DATASETS ON A DISTRIBUTED BLOCKCHAIN**

***A Report submitted.***

***in partial fulfillment for the Degree of***

**B. Tech**

***in***

**Computer Science and Engineering**

***by***

**SHIVANSHU SINGH 12000120131**

**AYUSH TAH 12000119129**

*Mentored By*

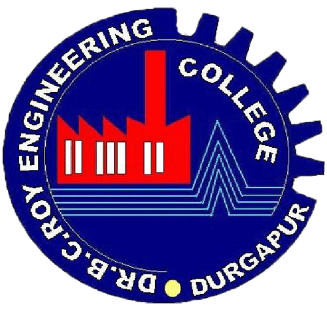
**PROF. DEBKANTA CHAKRABORTY**

pursued in

Department of Computer Science and Engineering

**DR. B. C. ROY ENGINEERING COLLEGE, DURGAPUR**

To



**DR. B. C. ROY ENGINEERING COLLEGE**

**DURGAPUR**

**February 2023**

**INTRODUCTION**

With the current rapid expansion and development of global navigation systems, information and communication technology, and sensor technology and sheer number of geo satellites being launched an unprecedented amount of geospatial data is being generated every smallest of intervals. The geospatial data we are talking about consists of but not limited to individual trajectories, location-based services, remote sensing images and so on, thus there has been a persisting need for the ability of sharing and use of spatial information reliably. As any data of such magnitude and importance is concerned data privacy, integrity and security play a crucial role both in sharing and storage of such data, however, it is quite a challenge to guarantee data privacy and security in geospatial data when it is so extensively used in a myriad of technologies. In addition, geospatial data is stored in many different formats and calibrated by different standards. Any effort to compare, combine or map data first requires a significant amount of data scrubbing and reformatting thus making a claim to the integrity of data becomes difficult. Data integrity depicts the data consistency and accuracy while sharing or storing the data, which quantifies the validity and fidelity of data. To preserve privacy and integrity of geospatial data, geospatial data should be validated to satisfy the security requirements and be protected from unauthorized modifications.

Recently Blockchain or rather IPFS which is based or derives many principles from blockchain has been realized as a possible solution to the problems we face with geospatial data because of the tamper-proof, traceable, trust-free, transparent, and decentralized characteristics it possesses. IPFS achieves these de-centralized and security characteristics by integrating consensus mechanisms, asymmetric cryptographic algorithms, and distributed data storage to name a few. Hence, IPFS presents itself as a major player in finding a solution for the problem we are trying to tackle.

**PROBLEM DEFINITION**

Ever-growing geospatial data needs secure and scalable storage for archiving data. Blockchain-based technologies can offer strong byzantine fault tolerance and distributed data storage and access. However, currently, there are no mature options for storing and organising large geospatial datasets on a distributed blockchain. Our aim is to create a software toolset for storage browsing and permission-based access to geospatial data using blockchain.

**RESEARCH QUESTION**

Can geospatial data storage using blockchain which provides distributed and decentralised data storage replace traditional centralised data storage?

**LITERATURE SURVEY, CURRENT STATUS AND GAPS**

There are several studies that have been conducted in this domain, though most are generally geared towards validation of data rather than the simple task of uploading and retention of data in IPFS. One study by J. R. Cedeno Jimenez, A. Folini, and M. A. Brovelli, pushed the idea of using IPFS to validate user contributed geospatial data in the IPFS network. They use the Velas blockchain infrastructure and implement a combination of a Discrete Global Grid System (DGCS) with smart contracts. Two kinds of smart contracts got produced: cell and registry smart contracts. Each DGGS partition has an individual cell smart contract that holds a list of observations present in that area. On the other hand, registry smart contracts monitor all DGGS cells that got added to the system. Presently, public authorities validate SIMILE(System for the Integrated Monitoring of Insubric Lakes and their Ecosystems) observations, which consumes time and may not be readily available. Hence, a fully functional prototype got developed to address this concern. Users can now add, oversee, and verify personal observations as well as those belonging to other users. This effort confirms the potential of creating decentralized apps for validating geographical data as a citizen science solution.

Another review paper by Jesus Rodrigo Cedeno Jimenez, Pengxiang Zhao, Ali Mansourian, and Maria Antonia Brovelli focuses more on the crowdsourcing of geospatial data and two main platforms that were developed for it, namely FOAM and D-GIS. But they found out the platforms were just conceptualized not realized/ deployed.

**OBJECTIVES OF THE THESIS**

The objective of the project is to build a web app that uses IPFS to store data locally or on the IPFS network. The features we are striving towards are:

* To build a better understanding of file storage, especially geospatial data storage on a blockchain.
* To build a web app to demonstrate our project with a robust and efficiently coded backend and simple looking and easy to use GUI on the frontend.
* To develop a more secure file storage system than the existing traditional centralised data storage.

**WORK DONE TILL NOW**

Currently, we are building the web app/web page for the project. We finished coding the uploading of files to our IPFS network. The GUI is currently under development. We are using JavaScript or more precisely Node.JS for the back end and HTML, CSS, and JavaScript for the front end. We’re working on a lot more features that we want to add, so we work even harder towards adding new features and mechanisms later down the road.

Till the current date the back end is operational but right now only accepts 1 file at a time which we are working to be able to do as many as the user wants. There’s also a basic front-end with renaming, uploading and downloading features but this will also be totally reworked to be more robust and fluent.

At present the backend is operational but only accepts one file at a time which we are working to be able to do as many as the user wants. There’s also a basic front-end with renaming, uploading and downloading features but this will also be totally reworked to be more robust and with fluent ui.

The app can be made on any operating system, but we used ubuntu because of its reliability and general ease of installation of the various packages that we are going to use.

We are using WSL2 (Windows Subsystem for Linux) which is available for systems running Windows 10 ver. 1903 or above, because with its recent milestones, it is very reliable and easy to work with and as we will do most of our work in CLI we don’t need to do a standalone installation of ubuntu.

Our backbone of the project runs on IPFS (InterPlanetary File System). IPFS is a protocol, hypermedia, and file-sharing peer-to-peer network for storing and sharing data in a distributed file system. It uses content-based addressing to uniquely identify each file in a global namespace connecting IPFS hosts. IPFS can replace the location-based hypermedia server protocols HTTP and HTTPS to distribute the World Wide Web.

Every file added to IPFS is given a unique address derived from a hash of the file’s content. This address is called a Content Identifier (CID) and it combines the hash of the file and a unique identifier for the hash algorithm used into a single string. IPFS currently uses SHA-256 by default, which produces a 256-bit (32 byte) output. SHA-256 is also used by bitcoin.

Once a file is uploaded to a node, the file is broken to pieces. Based on the contents of each chunk of the file, each of them is assigned a CID, and the file is converted into a Merkle DAG (Directed Acyclic Graph). Each node of the Merkle DAG consist of chunk of the file.

So, if any chunk of the file is mutated, that will make the whole file invalid, as the hash and the DAG depends upon the content of the file. The CID of the root node of the Merkle DAG is given to the user who uploaded the file, which can be used to access and to retrieve the file anytime.

**PLAN**

Currently we are building the webapp/web page for the project. We finished coding the uploading of files to our IPFS network. The GUI is currently under development. We are using JavaScript or more precisely NodeJS for the back end and HTML, CSS, and JavaScript for the frontend. We’re working on a lot more features that we want to add, so we work even harder towards adding new features and mechanisms later down the road. Following is the projected timeline we are to follow:

1. By this months end the front-end should be a lot more robust and the back-end will be modified to work with multiple files as well as world-wide sharing instead of a local network.
2. By the end of April the User Authentication/Login functions should be somewhat complete and the necessary visual changes would be complete.
3. After the deployment of the previous step, we would work on polishing the app even more and ironing out the wrinkles.