A Blockchain based Data Production Traceability System

Sandino Moeniralam

February 2018

Abstract

Satellite data is highly variable, with data sets continuously being transformed according to the needs of the user [31]. At present, no secure way of tracking down the changes made to the source data exists. The purpose of this research is to design a Blockchain based Data Production Traceability System for the Sentinel-2 satellite data, in order to keep track of, and verify each modification made to the original data set. We looked at what data provenance and data lineage exactly entails, what solutions currently exists and what these lack. Further more, the inherent safety characteristics Blockchain offers are taken into account to design a system that captures every step of the data transformation process, by recording the data sets, the production environment and the exact steps taken within this environment on the data sets, to trace back and verify every (intermediate) result.

Contents

1 Introduction

Donald Miller once said, "In the age of information, ignorance is a choice." [26] In a time where information is omnipresent, but truth is hard to come by, agreement on which data can be trusted is vital [20]. In the current information age, the need for data lineage is growing every year. Not limited to the world of science, but in any data processing setting. This need for data lineage coincides with the need for reproducibility [19], for without the ability to trace back any state, reproducibility is hard to come by and security can not be guaranteed. According to The National Physical Laboratory, data traceability in the context of satellite data is defined as the ability to verify for each step of a processing chain that the result of the current processing step is demonstrably linked to the output of the previous processing step [32]. This is almost equivalent to the definition of the International Vocabulary of Metrology, that describes data traceability as the ability to relate the results of a measurement through a documented unbroken chain [32].

The ability to trace back any modifications made to data, and verify its integrity is vital [20] for the reproducibility of scientific research. The concept of Blockchain could prove instrumental in building a system that guarantees data provenance as well as data lineage, due to the inherent characteristics that Blockchain has, such as security, verifiability, reproducibility and being distributed across all nodes. For any such system, identification of the type of data is an integral part of reproducibility. In this research, the focus is on satellite data in particular. Satellite data is processed at various levels ranging from Level 0 to Level 4 [29]. This research designs a system that solves the issues of data lineage in satellite data, by storing the different levels of data along with the production environment in which they are processed and a complete record of all the steps taken. Such a system would not only secure satellite data, it would also help in the reconstruction, analysis and verification of each step. The purpose of such a system would be to enable, for any moment in time, for any user, to verify each modification made. Given the necessary resources, each state should be reproducible. This system could be used in other fields as well, where large datasets are continuously processed by rapidly changing production environments.

As a real world example, the Earth Observation (EO) data of the Sentinel-2 Copernicus program will be taken. These missions deal with providing information concerning agriculture and help managing food security [27], emphasizing the importance of securing this type of data. At present satellite data at Airbus is stored as ordinary datasets in cloud services such as Google Cloud [12]. By storing the hashes and pointers of these datasets in a Blockchain, an extra layer of security could be offered while the responsibilities are shared among all the users, thus avoiding a single authority and single point of failure.

2 Problem statement

Ideally, any type of (satellite) data would be retrievable, reproducible and its integrity verifiable. Be that as it may, currently no system exists for storing satellite data this way. Furthermore, there is no system yet that captures the configurations of the rapidly evolving production environment in its entirety. The Copernicus program that is currently being rolled out is the largest Earth Observation (EO) program to date [27]. It will comprise of over 30 satellites doing long term earth observation, for a variety of goals. These include improving the management of the environment, understanding and mitigating the effects of climate change, and ensuring civil security [27]. All data will be freely available to anybody for research. Herein lies the reliability-issue addressed in this thesis; because of the significance of possible research based on this data, as well as the data itself, verification of such data as well as its reproducibility and trustworthiness are concerns to be vigorously addressed. With even a basic Version Control System (VCS) lacking [12], it comes to no surprise that tracking down individual changes to the data seems an impossible task at the moment. This is a problem since precious datasets are not well protected from malintent or neglect such as data degradation. Disagreement about the results can arise when the underlying datasets are not trusted by all. It is becoming a trend that scientific communities realize the benefits of sharing their data, and give access to their production environments [3]. To improve this international cooperation, trust in the data is crucial [34]. In this research we compare existing solutions to deal with data lineage, analyze what these lack, and design a system that does meet all our conditions for data lineage of the Sentinel-2 Copernicus EO data.

3 Related Work

Storing data in Blockchain-like databases has recently been researched at the University of Leipzig [23]. Different Blockchain-like database concepts were analyzed in regard to tamper-resistance, possible use cases, proof-of-concept and overall performance. There exists a large variety of different Blockchain implementations that offer different options. One of the first real world examples of a Blockchain based product traceability systems is Provenance, which offers a traceability system for materials and products that stores information securely, and that all parties involved can access [14]. Digital assets have distinct properties that make traceability more challenging than is the case with physical assets [19]. For storing and tracing digital assets, BigchainDB and Ethereum based solutions seem to be the most promising. BigchainDB is a scalable Blockchain database, storing actual datasets on chain [24]. An Ethereum based solution on the other hand, would only store the hashes of these datasets and the pointers to these datasets on chain. This would be a lighter implementation, and computationally cheaper, but does not address the issue of a necessary trusted third party where the actual datasets would reside.

A research that goes one step further, and not only deals with distributed storage of datasets in a Blockchain-like database, but also keeps track of the different versions of the same dataset is a recent article called "Advancing Open Science with Version Control and Blockchains" by George Mason University [11]. Here some basic requirements are defined that a Blockchain based VCS should adhere to. Though the limitation here is that mere version control is not sufficient for what we are trying to achieve. Normal revision control systems have no guarantee that the data has not been altered while being stored. Full data traceability and reproducibility requires two aspects, namely;

- the data is stored in an immutable manner, preventing any later modification,
- the production environment has to be recorded along with the production process. This would allow for the reproducibility of the data, regardless if the software used is still available later on.

The issue of data traceability and data reproducibility has been investigated in a project called *Quality Assurance for Essential Climate Variables* (QA4ECV), that was an international effort funded under theme 9 (Space) of the European Union Framework Program from 2014 until 2017 [30]. The limitation here was the same as mentioned before, namely that the datasets are not completely traceable nor reproducible based on the system alone. The production environment and production process are not recorded in such a manner for later usage. Here our research could expand upon these notions; it would enable the scientific community to store snapshots taken in an immutable ledger such as Blockchain.

4 Research question

The main research question is:

What requirements should a Blockchain based data production traceability system for satellite data adhere to?

To answer this topic, the following sub-questions are defined:

- *What does the data production process of Sentinel-2 Copernicus's Earth Observation data look like?*
- *What types of data are to be distinguished?*
- *How does one capture all the steps of the data production process?*

5 Data Production Traceability Aspects

5.1 Defining reproducibility

To design a system that allows for reproducibility of satellite datasets, it is important to define what reproducibility actually means. Reproducibility is often used interchangeably with repeatability [25] even though most scientists agree there is a slight difference between the two [13]. Repeatability indicates that the same results are acquired through a repetition of the same study, using the same location, measurement procedures, observer, measuring instrument done under the same conditions in repetition over a short period of time [8]. Reproducibility on the other hand indicates how close the results of experiments are that were conducted by different researchers, at different locations with different instruments, compare. In summary, reproducibility refers to the ability to replicate the findings of others while repeatability refers to the ability to replicate the findings of oneself.

However, according to Science Magazine, some basic terms in the world of science, such as reproducibility, replicability and repeatability are not standardized [22], and refer to the same concept. Three types of reproducibility are suggested: *methods reproducibility*, *results reproducibility* and *inferential reproducibility*. *Methods reproducibility* refers to the ability to repeat as precisely as possible all the processing steps done, with the same data and tools to arrive at the same results. *Results reproducibility* refers to arriving at the same results from the same data, using a different method. Lastly, *inferential reproducibility* refers to drawing the same conclusions, based on the same results of a similar study. This differs from results reproducibility, since not all researchers draw the same conclusions from the same results.

In this research we focus on designing a system that allows for methods reproducibility in an autonomous manner.

5.2 Data lineage and data provenance

Despite being used interchangeably, there is a distinct difference in data lineage and data provenance. Data provenance simply refers to being able to verify the origins of the data, while data lineage refers to the overall data life cycle that includes the origins and all the steps taken to arrive at the output. Thus, data provenance is a vital part of data lineage, which in itself is a complete record of the entire data production process [28]. Both make data traceability and reassuring data quality possible.

One key aspect of data lineage is the way the process is visualized. Due to the vast amount of meta-data that the entire chain can hold, the visualization is usually limited to a particular part of the chain, hereby omitting certain details. There usually exists several layers of abstraction, with the highest layer

giving a basic overview of the most important parts of the chain. These include the input and output dataset, and the systems the dataset interacts with. When zooming in, more details become available to the user [6].
How well documented, and how much meta-data is stored about the production process is determined by the data management requirements of a particular organization. These in turn depend on the regulation to which the organization must abide by. The more detailed a production process is recorded, the easier it is to reproduce, however, the more complex the recording process becomes.

Technically, one can distinguish two approaches for recording lineage in the context of digital data. Namely, *workflow* or *coarse-grain* lineage, and *dataflow* or *fine-grain* lineage [9]. *Workflow lineage* describes how derived data has been calculated from the original dataset, while dataflow lineage describes how data has moved through the processing chain. In other words, workflow describes the logical steps taken: what consequence a certain action has, for instance what step should be taken after a (partial) failure, whereas *dataflow lineage* manages the data itself and is a more complex than workflow lineage. The data can, for instance, be split, merged, imported or exported [15]. To acquire complete data traceability through a detailed recorded data lineage, both have to be incorporated [21].
At present, several open source lineage capture applications exist that do exactly that. CamFlow [33] and SPADE [2] are tools that provide OS lineage for the Linux kernel. Other applications exist for specific programming and scripting languages, such as NoFLow for Python scripts [10] and RDataTracker for R [7]. These applications differ from Version Control Systems in the sense that they focus on traceability and not on the recoverability of older versions of the same dataset.

What types of data are to be distinguished?
This research categorizes the following types of data necessary for complete reproducibility and traceability:

- The datasets
- The production environment, that can include the entire OS in which it runs
- The production process, which is a complete list of the processing steps and comments explaining the reasons and assumptions for making certain modifications

5.3 Sentinel-2 Copernicus EO data

Due to the costs of the Copernicus program, and it being the world's single largest observation program [], quality assurance is a vital aspect. Instead of scientific measurements that takes months of even years, Copernicus is designed to give uninterrupted data for decades on end. Thus, the "Quality Assurance

For Essential Climate Variables" (QA4ECV) was created. The goal of this project, that ran from January 2014 until December 2017, was to develop an internationally acceptable Quality Assurance. The reasoning behind this project was that the potential of satellite data to benefit climate change and air quality services is too great to be ignored.
The Provenance Traceability Chains that the QA4ECV designed, allow for the storage of input and output details, as well as the processing step. In this way, later on, datasets can be processed in the same manner and the output should be the same. What this system lacks though, is a way to actually store the data and the production environment in a secure way. It still makes use of a centralized architecture and does not hash the input for later verification.

What does the data production process of Sentinel-2 Copernicus's Earth Observation data look like?
We specifically focus on the Sentinel-2 missions, as Airbus actually built the two main satellites and actively uses data this mission produces. The Sentinel-2 mission provides optical imaging for land services, that can be used for emergency services. The data consists of multi-spectral data with 13 bands in the visible, near infrared, and short wave infrared part of the spectrum [31].
Satellite data is processed before it is released to the public. The data process consists of level 0 to level 4 [29][17], however, some organizations have slight adjustments to this model and differentiate within levels themselves, e.g. Level 1A-1C, 2A-2B, 3A-3B. For the Sentinel-2 mission, datasets are released to the public starting at level 1C. It is this data production process that we hope to improve in this research.

- Level 0 is raw, unprocessed instrument data
- Level 1A is unprocessed instrument data together with ancillary information
- Level 1B is data processed to sensor units, e.g. brightness temperatures [29]
- Level 1C is level 1B data that is resampled, conversed to reflectances and stripped of defective pixels
- Level 2A is a Map-Image product of level 1B data. Level 2A data is frequently labelled as "geo-referenced" [17]
- Level 2B are derived geophysical variables, for accurate spatial positioning on remotely sensed imagery
- Level 3A the data is accounted for relief displacement in order to obtain consistently high position accuracies.
- Level 3B covers a larger area than level 3A data, as it is in principle level 3A data combined

- Level 4 is modeled output or variables derived from multiple measurements

Identification of the type of data is vital for reproducibility [20]. For a truly secure traceability system that allows for the reproduction of any data set, the datasets themselves, along with the production environment and the documentation (log and/or configuration files) that includes the description of every step taken, must be stored. At present, the way QA4ECV proposed tracking data is visualized in Figure 1 and Figure 2.

Figure 1: The QA4ECV traceability chain [30]

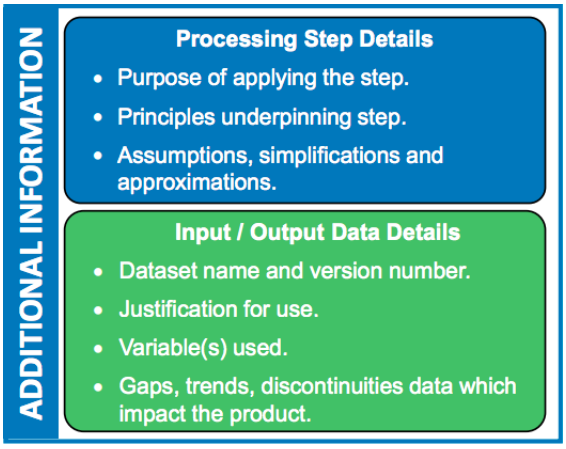

Figure 2: The meta-data stored on the traceability chain [30]

This design lacks verification of the data's integrity, as well as data provenance. What this model does provide is traceability to a certain extend, depending on how well documented the processing steps are.

The Sentinel-2 data is currently stored solely in the Google Cloud service. Should another cloud service be desirable for whatever reason, the data would have to be moved, and the pointers pointing to this data altered. One solution would be to allow for new pointers that point to the new locations of the same data be put in new blocks. In this way, moving data from one location to another would not be an issue.

6 Design

Having identified which information must be stored to allow for complete traceability and reproducibility, we now focus on the technology that can be used to store this information. Blockchain is an immutable, secure, interoperable and reproducible way of storing (meta)data. Because of these inherent characteristics Blockchain has, it makes sense to use this technology for our system. Ideally, we intend to design a system that has two basic views of the production process. One for humans, and one for machines. The human view would visualize the production process in such a way that any user with a reasonable amount of knowledge could understand and reproduce (parts) of the production process. On the other hand, the machine view would allow for machine to automatically replay (parts) of the production process.

6.1 Blockchain

Blockchain is a distributed ledger that exists of a list of blocks. Each block contains a hash of the previous block, a timestamp, a proof-of-work or proof-of-stake and the data that it needs to hold. The hashes that point to the previous blocks are what give Blockchain its security. Data that are stored on the Blockchain is practically immutable, since any modification made in a block will result in a different hash, and hence a brake up of the chain. Blockchain works on the basis of decentralized consensus, which means that at least 50% [23] of the nodes in a given network have to agree on the contents of their Blockchain. This is also what makes a Blockchain based system so hard to hack, after all, an attacker would have to successfully attack at least 50% of the nodes in a network.
As with any technology, Blockchain has its drawbacks. Scalability issues concerning the amount of data stored and how many nodes can exist in a network, while every transaction is still shared in a relatively fast manner, remain unresolved. Another disadvantage is the collective cost of operating a Blockchain network. Every block has to be verified by all, requiring enormous amounts of energy. The proof-of-work that gives Blockchain its security and immutability also has a drawback [16]. Covering the costs of operating a Blockchain network can prove difficult for a non-profit setup. The speed at which modifications are transferred through the network is another issue. The more nodes there are in a network, the longer it takes for a transaction to reach every individual node. Lastly, the immutability also means that any human errors that were added to the system are there to stay.

For all intends and purposes, Bitcoin is the perfect example of a provenance system since every Bitcoin ever mined is accounted for [14]. Every node in a network has access to a full copy of every transaction ever made in the system. This is exactly what is so appealing about the Blockchain technology in the context of data traceability by data lineage. On a more abstract level, a Blockchain can be seen as a model of state-machine replication, a service that maintains the state of an asset, where clients invoke operations to transform this asset,

that the Blockchain can emulate, and verify at every node [5].
For this research, two Blockchain applications are compared. Firstly, BigchainDB is a Blockchain database where data is stored at several nodes, while the metadata is stored at every node. The main drawback here is a lack of scalability for large volumes of data [24].
An Ethereum based approach on the other hand, would leave the issue of where the actual data is stored up to the users of the system [4]. In this case, the Ethereum network would only store hashes of and pointers to the data. The data itself would reside on local servers or on Cloud based services such as Google Cloud. The issue here though would be that any new user would have to be permissioned to gain access to the actual data, after which the researcher could verify the validity of the data using the Blockchain.

How does one capture all the steps of the data production process?
Practically, any data that is put into the Blockchain has to be transferred to the other nodes. In the case of this research, this data consists of the datasets on which the modifications have been done, the entire production environment, and lastly all the processing steps. These last can be split up into a list of steps for humans, and one for machines to automatically repeat every modification. Due to the size of the production environment alone, it would be impractical to transfer this entire setup to all nodes every time a modification is made. Instead, once every node has a complete setup, sending only the differences in the new production environment would speed up the entire process. After every step of the production process, every node would cryptographically verify the contents of every block, while for each step, a random node would actually replay the step to verify the work and dataflow.
Instead of merely storing transactions, the Ethereum Blockchain allows for the storage and execution of so called "smart contracts". Due to the decentralized nature of the Blockchain, and every node having their own full copy of all these smart contracts, Ethereum allows for the Ethereum Virtual Machine, in which the complete computational capacity of all the nodes in the network can be used by all. Smart contracts have created a level of automation, previously unknown. Once a smart contract is created and distributed, no single individual, including the creator, can change the contents of this. A contract could state that an amount X should be transferred to wallet Y when Z happens. Automatically, without the need for a trusted third party. The downside of this automation and the immutability of the contents of the Blockchain, is that errors can stay on the Blockchain indefinitely.

6.2 Alternatives to Blockchain

Blockchain and cryptographically signed linked lists share commonalities. Both can be used to verify data stored elsewhere, however, in Blockchain, the blocks are linked via hashes of the previous block and not as pointers as is the case in a linked list [1]. Secondly, data structure within a block is more complex than in a linked list. Merkle trees are used for storing the data.

Another key difference between Blockchain and a signed linked list is that a linked list is a data structure, whereas Blockchain is a protocol to come to a distributed consensus [1]. Every node in a network stores its own copy of the Blockchain for this reason. In a Blockchain blocks cannot be altered or removed later on, as is the case in a linked list. Using a design based on a signed linked list would make it too easy to maliciously alter data.

The Blockchain technology requires every block that is appended to the chain to be verified by all. Bitcoin's process, called mining, is highly expensive due to hardware and electricity costs. One implementation of Ethereum uses proof-of-stake instead of the above mentioned proof-of-work. With proof-of-stake a specific node is chosen to verify a block, based on the assets in that block, hereby circumventing the need for every node to try to verify a new block.

One major drawback of the Blockchain technology is that errors are hard to correct later on. Any information put on the Blockchain is in principle immutable. For this design to work, the assumption must be made that any data that is hashed, and whose pointer is put in the Blockchain, is correct. However, pointers to new data locations should be added in consecutive blocks, hereby overiding previous pointers.

6.3 Proposed design

By incorporating the Blockchain from the ground up, but not storing the actual data on there, we believe a light version of a data production traceability system could be implemented without changing the current production process. Each block would consists of:

Cryptographic hash of the previous block
The cryptographic hash is what gives Blockchain its strength. By chaining the blocks cryptographically together, the data becomes immutable without recalculating the entire chain.

Timestamp
Incorporating a timestamp gives users later on a good overview of when what exactly happened.

Proof-of-stake
The proof-of-stake makes it easy to verify a block, but hard to change the contents of a block. The creator of the block is chosen at random, based on the amount of stake he or she has within the block. For instance, if a person has a 25% stake of the contents of the block, he or she has a 25% chance of being chosen at random to create the block. This prevents the process of mining, circumventing unnecessary costs in computational power and electricity. If an attacker would want to alter the contents of a block once it has been created,

he or she would have to control more than 50% of the assets maintained in the network, instead of controlling more than 50% of the nodes in a network as is the case with proof-of-work. Overall, proof-of-stake is a safer and cheaper alternative to proof-of-work [18].

Hash(dataset)
The dataset is hashed, allowing users to verify that the datasets have not been altered.

Pointer to dataset
This can be one or several pointers, pointing to different locations where the dataset is stored. By using this structure, the actual datasets can stay stored as they are now, using the Google Cloud in the case of Airbus. It would be recommendable to use different locations, and different cloud services to guarantee access later on.

Hash(production environment)
The production environment (PE) contains a Virtual Machine, with a complete Operating System and all the necessary applications to replay the data. A snapshot in time is taken, hereby avoiding missing libraries, software versions not matching, or not being able to reproduce the data due to legacy.

Pointer to production environment
The same as with the datasets mentioned above.

Hash(production process)
Storing the production process (PP) into its most minute details is where this research differs from previous ones. We suggest splitting this process into two files, one for humans to read and one for machines to reproduce all the steps in an autonomous fashion. The file meant for humans could include the information also suggested by the QA4ECV, such as the purpose of applying the step, the principles underpinning the step, the assumptions, simplifications, and approximations at the time. Other valuable information that should also be included is auxiliary meta-data not stored in the datasets themselves, the variables used, and other aspects that might have had an effect on the production process. The file used by machine to reproduce the data should be written in Solidity, which is the contract-oriented, high-level language for implementing smart contracts [4], or in another language that would allow for the manipulation of entire VM's. This is the key difference in which this research differs from previous ones. By giving the ability for machines to reproduce any step of the production process, the errors humans often make are omitted.

Pointer to the production process
The same as with the datasets and production environment mentioned above.

Table 1: A schematic sketch

Block 0	Block 1	Block 2
hash(0) timestamp proof-of-stake	hash(Block 0) timestamp proof-of-stake	hash(Block 1) timestamp proof-of-stake
hash(dataset V1) pointer to dataset level 0 data hash(PE #1) pointer to PE #1 hash(PP #1) pointer to the PP #1	hash(dataset V2) pointer to dataset V2 hash(PE #2) pointer to PE #2 hash(PP #2) pointer to the PP #2	hash(dataset V3) pointer to dataset V3 hash(PE #3) pointer to PE #3 hash(PP #3) pointer to the PP #3

The entire Blockchain would look as seen in Table 1. This Blockchain would be stored at every node in the network that has been granted access to the data before final publication. Every step of the production process, from raw level 0 data until level 1C or level 2A, would be stored, verified and send across the network. Hereby giving all parties, both inside the network, as later on, the ability to completely reproduce the data.

Instead of designing a Blockchain from the ground up, Ethereum was chosen due to its size and implementation. It is a lot harder to spoof data on the Ethereum Blockchain than it would be with a smaller, self designed one.

7 Discussion

In this research, the requirements that a Blockchain based production traceability system for satellite data should adhere to were laid out. The volatile nature of digital data requires a different approach to allow for verifiable traceability than is the case with physical assets. Digital data can be copied, altered, and hard to reproduce. The production environment in which datasets are edited is extremely changeful. Software gets updated almost every day, libraries change, and reproducing certain outcomes may prove impossible without taking a snapshot of an entire production environment. Doing so, however, is not enough. The production process should be stored alongside the production environment, to allow for reproducibility of the outcome data, and traceability of the original data. By storing the complete production environment, that consists of a Virtual Machine with all the required application, the datasets can be reproduced long after these software versions become obsolete.

The concept of Blockchain was chosen due to the inherent characteristics, such as the immutability of the contents it holds, the decentralized nature and the ability for every node to retrace any step ever taken. By using the Ethereum Blockchain, smart contracts could be included that reproduce the data in an autonomous fashion.

This research has not addressed the way in which the datasets, production environment and production process are stored. Storing it on the Blockchain itself would not make it scalable since everything would be stored everywhere.

8 Conclusion

This research has shown that using a Blockchain based solution is possible and solves the issues of data traceability and data reproducibility but does not solve the issue of data storage and data degradation.
What requirements should a Blockchain based data production traceability system for satellite data adhere to? Every block should include hashes of, and pointers to, the datasets, production environment and the production process for humans and machines. This Blockchain should be used by all parties that modify the datasets before it is published. Using an established Blockchain such as Ethereum that also offers automated, distributed computing, every state of a particular data set could be traced in a secure manner.

9 Future work

Actually implementing this design would require addressing certain issues. To build a proof-of-concept, a more technical analysis is required that takes into account the different production environments that are currently used. Another factor that comes into play is to what degree the data reproducibility could be automated. What are the limitations of the Ethereum Virtual Machine, and can any hardware configuration be virtualized within its environment? Research should be done to analyze whether Ethereum smart contracts allow for the replication of every step of the production process, or whether these require a different programming language. The provenance systems for Linux and particular programming languages mentioned in this research could serve as a good starting point.
Scalability is an issue that plagues any Blockchain technology and is currently under investigation. Avoiding storing too much of the same information is key here.

References

[1] Patrick Reza Schnurbusch Alex Mizrahi, Abhishek Singh. Is a blockchain essentially a linked list? 2015.

[2] Dawood Tariq Ashish Gehani. Spade: Support for provenance auditing in distributed environments. 2012.

[3] Chad Berkley Dan Higgins Efrat Jaeger Matthew Jones Edward A. Lee Jing Tao Yang Zhao Bertram Ludäscher, Ilkay Altintas. Scientific workflow management and the kepler system.

[4] Vitalik Buterin. A next generation smart contract decentralized application platform. 2015.

[5] Christian Cachin. Architecture of the hyperledger blockchain fabric. 2016.

[6] Juliana Freire Claudio T. Silva and Steven P. Callahan. Provenance for visualizations. 2007.

[7] Elizabeth Fong Matthew Lau Barbara Lerner Thomas Pasquier Margo Seltzer Emery Boose, Aaron Ellison. Scientific data provenance in r: Rdatatracker and ddg explorer. 2014.

[8] Inc. Engineered Software. Repeatability and reproducibility. 1999.

[9] Sidra Islam. Provenance, lineage, and workflows. 2010.

[10] Vanessa Braganholo Juliana Freire Joao Felipe Pimentel, Leonardo Murta. noworkflow: a tool for collecting, analyzing, and managing provenance from python scripts. 2017.

[11] Foteini Baldmitsi Angelos Stavrou Jonathan Bell, Thomas D. LaToza. Advancing open science with version control and blockchains. 2017.

[12] Sjaak Koot. Commonsense. 2017.

[13] Labmate. What is the difference between repeatability and reproducibility? 2014.

[14] Project Provenance Ltd. Blockchain: the solution for transparency in product supply chains. 2015.

[15] PC Teach me. Ssis: Workflow vs. dataflow. 2009.

[16] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[17] Joseph M. Piwowar. Getting your imagery at the right level. 2001.

[18] Ameer Rosic. Proof of work vs proof of stake: Basic mining guide. 2017.

[19] P. Krause V. Curcin M. Tristan Vicente G. Michalakidis L. Agreus P. Leysen N. Shaw K. Mendis S. de Lusignan, S.-T. Liaw. Key concepts to assess the readiness of data for international research: Data quality, lineage and provenance, extraction and processing errors, traceability, and curation. 2011.

[20] Jiten Bhagat Iain Buchan Philip Couch Don Cruickshank Davide De Roure Mark Delderfield Ian Dunlop Matthew Gamble Carole Goble Danius Michaelides Paolo Missier Stuart Owen David Newman Shoaib Sufi Sean Bechhofer, John Ainsworth. Why linked data is not enough for scientists. 2010.

[21] Wasim Sadiq Cameron Foulger Shazia Sadiq, Maria Orlowska. Data flow and validation in workflow modelling. 2003.

[22] John P. A. Ioannidis Steven N. Goodman*, Daniele Fanelli. What does research reproducibility mean? 2016.

[23] Martin Stoffers. Trustworthy provenance recording using a blockchain-like database. 2017.

[24] Andreas Müller Dimitri De Jonghe Troy McConaghy Greg McMullen Ryan Henderson Sylvain Bellemare Alberto Granzotto Trent McConaghy, Rodolphe Marques. Bigchaindb: A scalable blockchain database. 2016.

[25] Unknown. Accuracy, precision, reproducibility, repeatability resolution. what do they mean?

[26] Unknown. Az quotes. 2013.

[27] Unknown. Copernicus overview. 2017.

[28] Unknown. Data lineage. 2017.

[29] Unknown. Data processing levels. 2017.

[30] Unknown. Quality assurance for essential climate variables. 2017.

[31] Unknown. Sentinel-2 data products. 2017.

[32] Unknown. Traceability of eo data. 2017.

[33] Unknown. Practical linux provenance. 2018.

[34] Dennis Gannon Yogesh L. Simmhan, Beth Plale. A framework for collecting provenance in data-centric scientific workflows. 2006.