Modeling and Preserving Greek Government Decisions using Semantic Web Technologies and Permissionless Blockchains

Themis Beris and Manolis Koubarakis

Dept. of Informatics and Telecommunications, National and Kapodistrian University of Athens, Greece
{tberis, koubarak}@di.uoa.gr

Abstract. We present a re-engineering of Diavgeia, the Greek government portal for open and transparent public administration. We study how decisions of Greek government institutions can be modeled using ontologies expressed in OWL and queried using SPARQL. We also discuss how to use the bitcoin blockchain, to enable government decisions to remain immutable. We provide an open source implementation, called DiavgeiaRedefined, that generates and visualizes the decisions inside a web browser, offers a SPARQL endpoint for retrieving and querying these decisions and provides citizens an automated tool for verifying correctness and detecting possible foul play by an adversary. We conclude with experimental results illustrating that our scheme is efficient and feasible.

Keywords: Linked Open Data, Blockchain, Open Government, Semantic Web, Bitcoin, Tamper-proof, Public Services

1 Introduction

Government decisions which are made by public authorities and institutions, affect significantly the daily lives of ordinary citizens. Therefore, an important dimension of open government is making these government decisions open and easily accessible to the public.

Diavgeia (`https://diavgeia.gov.gr/en`, διαύγεια means transparency in Greek) is a Greek program introduced in 2010, enforcing transparency over the government and public administrations, by requiring that all government institutions have to upload their decisions on the Diavgeia Web portal. The portal is managed by the Ministry of Administrative Reform and E-Governance. Diavgeia is now fully implemented by public authorities. The current rate of uploads in the Diavgeia portal is 16.000 decisions per working day, summing up to a total of 26 million decisions up to now. However, decisions are currently uploaded as PDF files and follow no structuring of their textual content. As a consequence, interested parties (the government, public authorities, ordinary citizens, non-government bodies, courts, the media, etc.) rely on keyword search over PDF files, in order to find decisions that might effect them in some way or verify that

uploaded decisions have been taken according to the law. Also, despite the fact that these decisions are digitally signed, there is no integrity mechanism which ensures the immutability of all decisions over time.

In this work, we aim at revolutionizing the way that decisions of the Diavgeia program are made public, by following the footsteps of other successful efforts in Europe which publish legislative documents as open linked data [1]. By applying Semantic Web techniques, we envision a new state of affairs in which ordinary citizens have advanced search capabilities at their fingertips on the content of public sector decisions. In addition, through the use of the bitcoin blockchain, we enable decisions to remain immutable, introducing unprecedented levels of transparency to the Diavgeia program and ensuring the integrity of the published decisions as open linked data.

Contributions. Towards achieving the aforementioned goals, we provide an open source implementation, named DiavgeiaRedefined[1], which aims to replace the current production implementation of Diavgeia. The DiavgeiaRedefined project consists of the following modules:

1. **Diavgeia ontology**. We follow the latest Semantic Web standards and best practices and develop an OWL ontology, called *Diavgeia ontology*, for modeling the content of decisions uploaded by the Greek public authorities to the Diavgeia website. Using this ontology, decisions can be encoded in RDF and be interlinked with other Greek government data (e.g., legislation in the system Nomothesia [1]), empowering interested parties to pose rich queries over these data sources. The linking of Diavgeia with Nomothesia has the benefit of making sure that the references of public sector decisions refer to valid legislative documents (laws). We also interlink Diavgeia with a dataset encoding the administrative geography of Greece.
2. **Web editor and Visualizer**. DiavgeiaRedefined provides web applications to prove that Semantic Web technologies can be used by the Greek government in a user-friendly manner. We develop a Web editor that can be used by public authorities, for authoring their decisions. The result of this procedure is the creation of decisions expressed in RDF, compatible with the Diavgeia ontology. We also develop a Web tool that visualizes the aforementioned decisions.
3. **Blockchain tools**. By organizing and aggregating decisions into Merkle trees [2], we provide a way to store decisions expressed in RDF on the bitcoin blockchain with very low cost. We also develop a blockchain verifier that can be used by interested parties to verify the correctness and detect possible foul play by a participant in the process.
4. **SPARQL Endpoint**. By employing Fuseki server, we empower interested parties to browse, search and pose interesting SPARQL queries to public sector decisions.

[1]The source code of the project can be found on `https://github.com/ThemisB/diavgeiaRedefined` and its landing page can be accessed on `http://pyravlos-vm5.di.uoa.gr/diavgeia/`.

5. **Evaluation**. We evaluate DiavgeiaRedefined in two ways: (i) by calculating the blockchain validation time for a month's regular workload, and (ii) by comparing it with the current implementation of Diavgeia in terms of disk space usage.

Organization. The rest of the paper is organized as follows. Section 2 discusses related work in legislative knowledge representation using Semantic Web technologies and presents endeavors that combine linked data with blockchain technology. Section 3 presents Diavgeia. Section 4 discusses the Diavgeia ontology, presents the Web editor and Visualizer and some interesting SPARQL queries. Section 5 provides background information on the bitcoin blockchain. Section 6 describes the two blockchain tools developed for the preservation and verification of decisions. Section 7 presents the evaluation results. Last, Section 8 summarizes our contributions and discusses future work.

2 Related Work

Democratizing access to government and legislative documents has been a primary concern of many governments across the world. Many countries have a government portal where government data is made available free of charge, in some cases as linked data (`https://data.gov.uk/`). The development of information systems archiving the content of legislative documents as linked data has been a common practice towards making legislation easily accessible to public [3]. For example, the MetaLex Document Server [4] hosts all national regulations of the Netherlands, while [5] presents a service for publishing the Finnish legislation as linked open data. Nomothesia (`http://legislation.di.uoa.gr/`) [1] is a research project in our group, which publishes Greek legislation as linked open data and it also offers SPARQL endpoint and RESTful API to interested parties for search reasons. All of the above endeavors adopt different vocabularies and ontologies to express the particularities of each country's legislation. Recently, the European Council introduced the European Legislation Identifier (ELI) [6] as a common framework that can be adopted by the national legal publishing systems in order to unify and link national legislation with European legislation. ELI is partly based on the use of Uniform Resource Identifiers (URIs), and partly on a set of structured metadata for referencing European and domestic legislation. All of the afforementioned vocabularies and ontologies, are not a one-size-fit-all model but they have to be extended to capture the particularities of national legislation systems.

The Semantic Web community has just begun to consider applications which take advantage of the distributed, undeletable and immutable nature of the blockchain. Decentralized Semantic Identity [7] examines a semantic approach for W3C WebID, in which the Namecoin blockchain is used to register the user's WebID URI and domain names. In this endeavor, the proposed authentication scheme is outside the control of any single entity. Recently, [8] proposed a linked data based method of utilizing blockchain technology to create tamper-proof audit logs that provides proof of log manipulation and non-repudiation. Finally, [9]

discusses what Semantic Web research and development can offer to blockchain research and development and vice versa.

The present work follows in the footsteps of the aforementioned efforts. DiavgeiaRedefined offers an OWL ontology which extends ELI and is linked with the Nomothesia ontology [1] and with the ontology of the administrative geography of Greece[2]. Similar to the blockchain enabled audit logs, we use bitcoin's transaction scripting language to store government data on the blockchain as metadata of a blockchain transaction.

3 Background on Diavgeia

In this section, we present Diavgeia in detail and point out the problems of the current implementation.

3.1 Greek public sector decisions and relevant laws

Public sector decisions cover a broad spectrum of activities in Greece. The Greek government has enacted 34 different decision types that may be uploaded on Diavgeia. The decision type is chosen by the government institutions according to the context of the decision. Despite the many different decision types, we observed that the majority of them follow the same pattern. A decision starts by refering to a number of different Greek laws on which is based[3] and then gives the main text of the decision. The following figure illustrates an example of an *Appointment* decision type that adheres to the aforementioned pattern.

Example of an *Appointment* decision type

Appointment of R.F. as Full Professor

In accordance with:

1. The provisions of Law 3549/2007, article 25, paragraph 1.
2. The provisions of Presidential Decree 2011/54.
3. The provisions of Law 4386/2016, article 70, paragraph 4.

We decide:

1. The appointment of R.F. as Full Professor at the X department, at the Y University, on the subject of "Semantic Web".

Despite the fact that this pattern can be used to define a common format for the different types of decisions, for the time being, public sector authorities

[2]The Greek Administrative Geography dataset and the ontology are available on: `http://linkedopendata.gr/dataset/greek-administrative-geography`

[3]The following is a fun example. A recent decision, listing the proposals that will be funded under a particular research and innovation call, starts with references to 33 (!) Greek laws. The good news is that 176 proposals will be funded; one of them for our work extending Nomothesia [1].

upload their decisions as PDF files which follow no structuring of their textual content. Futhermore, citizens have no guarantee that the legislative references of a decision exist and are valid (such as *Laws* and *Presidential Decree* of the appointment example). By using the 5-star rating model for data [10], Greek public sector decisions are marked as 1-star.

In this work, we improve the current way of publishing Greek public sector decisions on the Web, by expressing them as 5-star open linked data. We take advantage of the aforementioned pattern and develop the Diavgeia ontology based on it. Technically, we view decisions as a collection of legal RDF documents with this standard structure.

We also employ the Nomothesia in order to ensure that the references to Greek legislation exist. Nomothesia has so far published 5 primary types of Greek legislation (*Constitution*, *Presidential Decrees*, *Laws*, *Acts of Ministerial Cabinet*, and *Ministerial Decisions*), as well as, 2 secondary ones (*Legislative Acts* and *Regulatory Provisions*). Nomothesia structures all legal documents, by using persistent URIs according to the template proposed by ELI `http://www.legislation.di.uoa.gr/eli/{type}/{year}/{id}`. For instance, for the first provision of the appointment example, a linking of Diavgeia with the Nomothesia URI `http://legislation.di.uoa.gr/eli/law/2007/3549/article/25/paragraph/1` can be made. By integrating Nomothesia into Diavgeia, we also give citizens the ability to simply click to the legislative references of public sector decisions and see instantly the relevant passage of Greek legislation.

3.2 Metadata of Decisions

In addition to the uploading of the PDF file, public sector authorities also have to fill metadata information which describe the decision. The metadata used vary according to the type of decision. For instance, the metadata of the *ExpenditureApproval* decision type, holds important information about government's expenditure (such as the sender and receiver VAT registration numbers, the expense amount, etc). Diavgeia offers an OpenDataAPI (`https://diavgeia.gov.gr/api/help`) that can be used as an endpoint to query over the metadata. Despite the fact that OpenDataAPI is a step towards promoting transparency, inconsistency between decision text and metadata information is possible[4]. In our work, we embed metadata information into the RDF document, eliminating the possibility of inconsistency.

3.3 Identifiers and Modifications of Decisions

Each decision is assigned a unique Internet Uploading Number (IUN), certifying that the decision has been uploaded on Diavgeia. IUN is of significant importance, since citizens and other public authorities can use decisions, by solely

[4]An article about inconsistent metadata on Diavgeia website: `https://eellak.ellak.gr/2016/07/06/veltionontas-tin-piotita-dedomenon-stin-diavgia/`

referring to their unique number. In addition to IUN, each decision is also assigned a unique version token. Government institutions can upload a new version of a decision by claiming a new version token, but maintaining the same IUN. Diavgeia functions in an **append-only** manner, as it maintains the original decision with all its subsequent modifications. This is exactly why Diavgeia is amenable to blockchain technologies as we discuss in Section 6.

4 Modeling Decisions using Semantic Web Technologies

In this section, we develop an OWL ontology for modeling decisions of Diavgeia. We call our ontology *Diavgeia ontology* and we discuss its current version that adopts the ELI framework and the Nomothesia ontology. We present the Web editor and Visualizer components of DiavgeiaRedefined that generate and visualize the decisions expressed in RDF, respectively. Futhermore, we describe the linking of decisions with other datasets and we pose interesting SPARQL queries which take advantage of this interlinking.

4.1 The Diavgeia ontology

The ontology of Diavgeia is based on the pattern followed by public sector decisions, as discussed in Section 3.1. It imports and uses properties that are defined in the ELI ontology and the Nomothesia ontology. The core[5] of Diavgeia ontology is shown on Figure 1. The 34 different decision types can be viewed as legal documents (class `LegalResource` of the ELI ontology). A decision (`LegalResource`) `changes` itself, by generating a new `version` and maintaining its unique `iun`. A `LegalResource` is composed of multiple `Considerations` and `Conclusions`. The `Consideration` class models the passages that are used to prove the validity of the decision (e.g. the three provisions of the appointment example), while `Conclusion` models the passages that are used as conclusions of the decision (that is the final passage of the appointment example). Both `Consideration` and `Conclusion` classes, use the `cites` property of the ELI ontology, to make a reference either to Greek Legislation (Nomothesia) or to another decision of Diavgeia. The `has_text` property is used to describe the text body of either `Consideration` or `Conclusion`.

The Diavgeia ontology offers 121 properties to cover all the particularities of different decision types. In addition to `Consideration` and `Conclusion`, the ontology provides classes which describe important public sector activities. For instance the class `Expenses` links an expense of a public authority to an individual or business. For the time being, this crucial information is expressed as metadata of the PDF decision, underlying the possibility of metadata inconsistency as described in Section 3.2. By merging metadata and decision text in a single RDF file, this possibility is eliminated.

[5]The full Diavgeia ontology is available on: `https://github.com/ThemisB/diavgeiaRedefined/blob/master/rdf/diavgeia.owl`

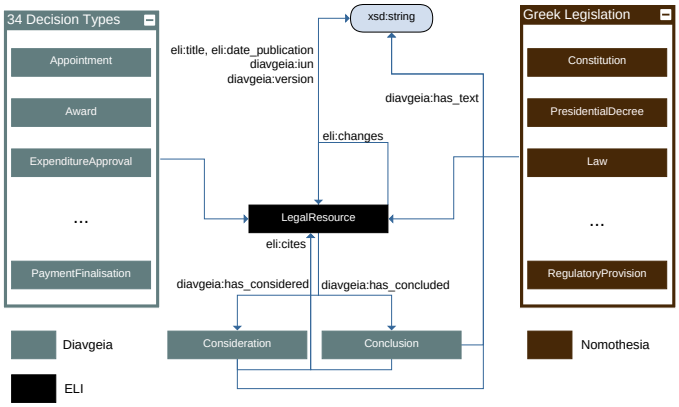


Fig. 1. The core of Diavgeia ontology

In order to identify legal resources, we also need appropriate URIs. Persistent URIs is a strongly recommended best practice [11], according to ELI. It is very important to have reliable means to identify the public sector decisions. Based on what is stated in Section 3.3, we can structure the persistent URIs of decisions according to the template `http://www.diavgeia.gov.gr/eli/{iun}/{version}`. Modifications of a decision result to the generation of a new URI which has the same `iun` and a new `version` number. Thus, the version of an enacted decision can be seen as the decision which has the most recent `date_publication` for a specific `iun`.

4.2 Web editor & Visualizer

DiavgeiaRedefined offers two main Web components in order to transparently adopt Semantic Web technologies to the production implementation of Diavgeia. The first one is a web editor for decisions, used exclusively by public sector authorities. The Web editor is a well-structured HTML form that government institutions can use in order to write their decisions. The HTML elements of the form are associated with the properties and classes of Diavgeia ontology. By submitting the form, the decision is stored both as a compressed Notation3 file in the filesystem of Diavgeia and in Jena Apache's triple store.

The Visualizer is another component of DiavgeiaRedefined which can be used both by public authorities and citizens. Its purpose is to provide a visualization of the decisions expressed in RDF inside a Web browser. Users provide the persistent URI of the decision they want to visualize and the decision is displayed in the browser in a user-friendly manner.

4.3 Linking decisions with other public sector data

The linking of decisions with other public sector data, can be done by public authorities, using the Web editor component of DiavgeiaRedefined. Firstly, the

`Consideration` or `Conclusion` classes of a decision, may make reference to the Greek Legislation of Nomothesia, as mentioned in Section 4.1. Linking Diavgeia with Nomothesia is easy, since the latter provides persisent URIs according to template `http://www.legislation.di.uoa.gr/eli/{type}/{year}/{id}`.

Public authorities have also to link `SpatialPlanningDecisions` decision type, with the dataset of administrative geography of Greece. Linking decisions with it is also easy and it is achieved through the construction of constant mappings.

4.4 Querying the Resulting RDF Data using SPARQL

By employing the Fuseki Server, we enable the formulation of complex queries over decisions of Diavgeia. This provides interested parties a mechanism to monitor the decisions of public sector organizations. We present some interesting queries one can pose.

```
SELECT ?decision WHERE {
  ?decision diavgeia:has_expense ?expense;
            eli:date_publication ?date.
  ?expense diavgeia:expense_amount ?amount.
  FILTER (?date >= "2017-01-01"^^xsd:date &&
  ?date <= "2017-12-31"^^xsd:date)
} ORDER BY DESC(?amount) LIMIT 5
```

Retrieve the decisions with the 5 highest government expenses of 2017.

```
SELECT DISTINCT ?decision WHERE {
  ?nomothesiaLegislation eli:passed_by ?signatory;
                         eli:date_publication ?date.
  ?signatory foaf:name "I.Stournaras".
  ?reference eli:cites ?nomothesiaLegislation.
  {?decision diavgeia:has_concluded ?reference} UNION
  {?decision diavgeia:has_considered ?reference}
}
```

Find decisions which make a reference to greek legislative documents that have been signed by the ex-Greek Minister of Finance I. Stournaras.

5 Background on Bitcoin Blockchain

Bitcoin [12] is the first decentralized digital currency based on a distributed, peer-to-peer consensus network. Transactions propagate through the network in order to be verified and stored in a blockchain. Blockchain is the immutable public distributed ledger which records all bitcoin transactions, forming a chain of blocks. Each block in the blockchain is composed of the previous block in the chain and a payload of transactions.

Bitcoin uses a stack-based scripting system for modeling transactions, called Script [13]. Transactions consist of multiple inputs and multiple outputs. Bitcoins are transferred on a transaction input and output, where the input defines where bitcoins are coming from and the output defines the destination. OP_RETURN opcode is a special instruction of Script which allows to save metadata up to 80

bytes on a transaction output [14]. *Miners* are specialized nodes on the network that keep the blockchain consistent, complete, and unalterable. By solving a hard cryptographic problem, miners generate and add new blocks to blockchain. The rest nodes of the network can easily verify and mutually agree that the solution given by the miner is correct and accept the new block.

The consensus algorithm of bitcoin guarantees that, for an attacker to be able to alter an existing block, he must control the majority of the computational resources of the network [15]. As more transactions and blocks are generated, the difficulty of the cryptographic problem increases significantly, which makes the tampering of data written in the blocks very difficult. This security property is often rephrased by saying that the bitcoin blockchain can been seen as an immutable, permissionless data structure. Thus, even if the main goal of bitcoin is to transfer digital currency, there are certifying services which take advantage of the tamper-proof nature of blockchain, by providing users a way to certify existence or ownership of documents (such as *Proof of Existence*[6], *OpenTimestamps*[7] and *Stampery* [16]). In our work, we use the OP_RETURN opcode to embed government data as metadata of a bitcoin transaction, similar to the aforementioned certifying services.

6 Preserving Decisions using Bitcoin Blockchain

In this section, we describe the use of the bitcoin blockchain on Diavgeia. We present in detail the two blockchain tools that DiavgeiaRedefined offers, called Stamper and Consistency Verifier.

6.1 Stamper

Stamper is the tool which should be used by the administrators of Diavgeia in order to store public sector decisions on the bitcoin blockchain. The stamping procedure is described as follows:

1. Government institutions upload their decisions on Diavgeia. The backend of Diavgeia stores decisions as compressed Notation3 files in its filesystem and in the triple store.
2. The administrator of Diavgeia has to decide that the stamping procedure should take place at predefined time intervals t, ensuring the integrity of decisions. Thus, the backend of Diavgeia starts a new stamping procedure every t time units.
3. At the start of the stamping procedure, we find all the compressed Notation3 decisions which have not been stamped yet. Stamper organizes and aggregates these decisions into a Merkle Tree [2], using the hash function SHA-256. The root of the Merkle tree represents the fingerprint of the decisions which will be included in the forthcoming bitcoin transaction. By

[6]Proof of Existence: `https://poex.io/about`

[7]OpenTimestamps: `https://opentimestamps.org/`

applying the SHA-256 hash function on the Merkle tree construction, the resulting root has a constant size of 32 bytes.

4. The next step is to create a Bitcoin transaction and broadcast it to the rest of the network. DiavgeiaRedefined uses the bcoin library (`http://bcoin.io/`), offering Diavgeia an spv node[8], maintaining only a chain, a pool, and a *hierarchical deterministic (HD)* wallet [17] based on BIP44 [18].
   A stamping transaction in our model consists of two outputs and one input. The first output contains the OP_RETURN opcode followed by the Merkle root in the *scriptPubKey* output (*scriptPubKey* = OP_RETURN + Root). This output guarantees the immutability of decisions. The second output is a pay-to-pubkey-hash[9], having as *pubKey* the next derived public address of the HD wallet. The input *scriptSig* consists of Diavgeia's signature and the current *publicKey* derived from HD wallet (*scriptSig* = *signature* + *publicKey*). The size of a stamping transaction is 267 bytes. In order to have certain guarantees that our transaction will be written into the next block and confirmed nearly immediately, mining fees can cost up to 120,150 satoshi (0.00125 bitcoin), which at the time of writing roughly amounts to $16.84.

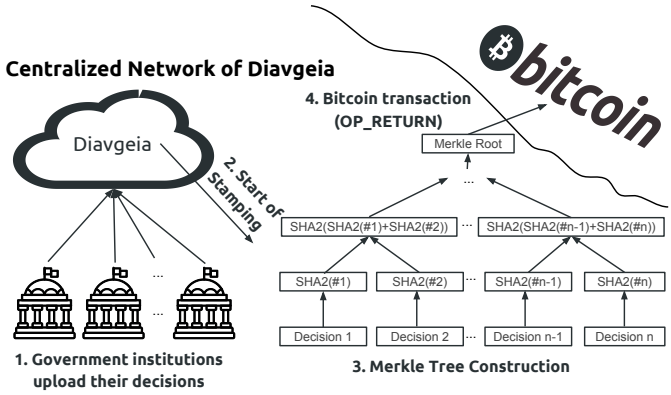


Fig. 2. The Stamping procedure

After the end of each stamping transaction, Diavgeia publishes to its website the transaction identifier (Txid) and the order of decisions, as used for the Merkle tree construction. It also publishes once, the Master Public Key of its HD wallet. By publishing Diavgeia's master public key, interested parties are able to track the sequence of public keys and stamping transactions of Diavgeia.

[8] A method for verifying if particular transactions are included in a block without downloading the entire block (`https://bitcoin.org/en/developer-guide#simplified-payment-verification-spv`).

[9] `https://en.bitcoin.it/wiki/Script#Standard_Transaction_to_Bitcoin_address_.28pay-to-pubkey-hash.29`

These publications are necessary to be made for the proper functionality of the Consistency Verifier (see Section 6.3).

6.2 Guarantees of Stamper

The Stamper tool provides high levels of immutability guarantees, especially when t value is configured to be small. Generally, the threat of a decision's modification or deletion appears on the time gap between two consecutive stampings. Small t values imply more stamping invocations and as a result Diavgeia creates more stamping transactions, but this comes at a higher cost. We consider a t value ranging from 3 hours to 1 day, to be an affordable solution for the government, since the daily cost of the usage of the blockchain will range from 0.00125 to 0.005 bitcoin ($16.84 - $134.72). The threshold for a decision's modification is also small, since an adversary (the administrators of Diavgeia, the government or other public authorities) are able to modify the decision in the next 3 hours to 1 day after its publishment.

As mentioned in Section 6.1, Stamper uses the open source bitcoin library (bcoin) in order to create the stamping transactions and relay them to the network. DiavgeiaRedefined does not use existing blockchain timestamping services (such as Stampery or OpenTimeStamps) because, in case of a foul play by an adversary, these third-party services might be accused of having modified the Merkle root in the first place.

6.3 Consistency Verifier

Consistency Verifier is the tool which can be used by the interested parties in order to verify that decisions have remained immutable over time. Algorithm 1 formalizes the steps Consistency Verifier takes to verify the integrity of decisions.

Data: Decisions included in stamping transaction i: d_i, Master Public Key: *mpk*
Result: Boolean result of verification.

```
1  foreach usedPublicAddress of mpk do
2  |  transaction ← getTransactionBySigScript(usedPublicAddress);
3  |  if transaction has OP_RETURN in the scriptPubKey output then
4  |  |  merkleTree ← constructMerkleTree(d_i);
5  |  |  if merkleTree→merkleRoot != transaction→merkleRoot then
6  |  |  |  return false;
7  |  |  end
8  |  end
9  end
10 return true;
```

Algorithm 1: Consistency Verification procedure

The first step is to download the compressed Notation3 decisions which have been included in stamping transactions. Afterwards, the verifier downloads in ascending time order all bitcoin transactions (using the *chain.so* bitcoin block reader, available at `https://chain.so/`), related to the used public addresses

derived from Diavgeia's master public key. In case of a stamping transaction, the verifier constructs the Merkle tree using the decisions of the first step. If the computed Merkle root is equal to the Merkle root found on the stamping transaction, decisions have remained unmodified.

7 Experimental Evaluation

This section presents a scalability evaluation of the Consistency Verifier tool and discusses the disk space reduction that gzip compression of Notation3 files offer. Section 7.1. describes the synthetic dataset used in the Consistency Verifier experiment. In Section 7.2, we illustrate the details of the test environment and in Section 7.3. we discuss the results of the experiment. The disk space reduction is presented in Section 7.4.

7.1 Dataset

To simulate the consistency verification process, we generated synthetic gzip Notation3 decisions[10], according to the *Diavgeia ontology*. Firstly, synthetic decisions have 7-17 `Consideration` and `Conclusion` class entities, each one of them has 150-350 random bytes as text part. Moreover, we have included several common-used properties, such as protocol number and thematic categories of a decision, as well as, information related to the departments of government institutions which upload them (phone number, address, etc).

We examine the time it takes an interested party to verify the consistency of Diavgeia in a month's common workload. We consider the scenario in which Diavgeia stores decisions on bitcoin blockchain, once a day. According to the Webpage of Diavgeia (`https://diavgeia.gov.gr/en`), the current rate of uploads is 16000 decisions per working day and assuming a month has 22 working days, we make 22 bitcoin stamping transactions. To examine the scalability of the verifier, we provide 3 different datasets, containing 8000, 16000 and 24000 decisions per day, summing up to 176000, 352000, 528000 compressed gzip N3 decisions, respectively.

7.2 Test Environment

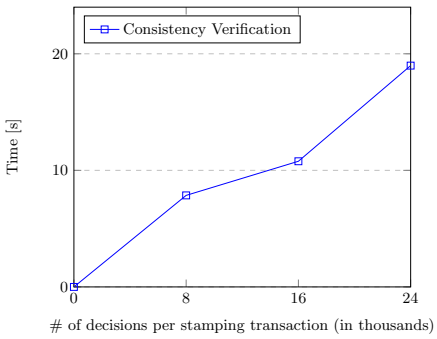
The verification experiment was run on a MacBook Pro with a 2.9 GHz Intel Core i5 processor and 8GB of memory, since this process may be executed by interested parties with a standard modern computer. The Javascript methods used to measure the elapsed verification time is *console.time* - *console.timeEnd*. The execution time measures the time needed to create the 22 Merkle trees and compare the computed roots with the roots extracted from the stamping transactions. The recorded time does not take into account any network time; the time needed to download synthetic decisions from our Web server or the

[10]Datasets are available in: `https://doi.org/10.6084/m9.figshare.5729292.v1`

time needed to gather bitcoin transactions, by making requests to *chain.so*. To account for variability in the testing environment, each reported elapsed time is the average of five independent executions.

7.3 Experimental Results

The experiment consisted of retrieving all synthetic decisions from our Web server and bitcoin transactions from *chain.so* and then compare the corresponding Merkle roots for validity, as presented in Algorithm 1. We use the 3 different datasets described in Section 7.1. The elapsed verification time is plotted in Figure 3.



#of decisions per stamping tx	Verification Time (s)
8000	7.853
16000	10.780
24000	18.986

Fig. 3. Evaluation of the Consistency Verifier

This experiment validates the linear time growth of the Consistency Verifier. The integrity check for a month's regular workload, consisting of 16000 decisions per day, takes about 11 seconds. Even for the extreme case of 24000 decisions per day, the verifier takes approximately 19 seconds to perform the integrity check. These results validate the scalability of our blockchain solution and demonstrate that interested parties can efficiently perform integrity checks over the data of Diavgeia.

7.4 Disk Space Reduction

Diavgeia currently hosts over 26 million decisions, leading to disk space limitations. The average size of a PDF-decision is about 2.5MB, summing up to a total of 65TB. We have created a sample, consisting of equivalent PDF and compressed gzip Notation3 files, for each different decision type of Diavgeia ontology[11]. For the aforementioned sample, we notice that compressed Notation3 decisions are about 86 times smaller, compared to the equivalent PDF files. Hence, encoding decisions in RDF not only allows for sophisticated SPARQL querying, but it also saves space.

[11]The sample is available on: `https://github.com/ThemisB/diavgeiaRedefined/tree/master/rdf/samples`

8 Conclusions and Future Work

In this paper we presented how Greek public sector decisions can be published as linked open data using Semantic Web technologies. We also discussed how to use the bitcoin blockchain to guarantee decision immutability over time. The Diavgeia ontology we employed, is based on the latest European standards and captures the particularities of Greek public sector decisions. We highlighted the importance of intelinking Diavgeia with other publicly available open data, by posing interesting SPARQL queries. We implemented the Web editor and Visualizer components in order to transparently adopt Semantic Web technologies in a user-friendly manner. We also introduced two blockchain tools; Stamper is responsible for storing government data on the bitcoin blockchain and Consistency Verifier provides citizens an automated way to verify the integrity of decisions. Finally, we evaluated the Consistency Verifier and measured the disk space reduction which compressed Notation3 decisions offer over the current PDF-format.

As an initial step of our future work, we would like to proceed with a usability evaluation of DiavgeiaRedefined and study possible improvements. As a part of functional improvements, we would like to implement a question answering system which will translate natural language queries to SPARQL queries. This system will offer ordinary citizens a way to examine the legality and good administration of Diavgeia, without posing SPARQL queries by themselves. Futhermore, we plan to integrate the HDT mechanism [19] to store the decisions on the Apache Jena Fuseki as compressed data and pose SPARQL queries, without the need of decompressing the decisions.

We acknowledge bitcoin's limitations in terms of cost, speed, and scalability [20]. We would like to apply Stamper and Consistency Verifier to other blockchain technologies, such as Ethereum [21]. In our future work, we will further develop Consistency Verifier. Firstly, we will offer a slower, but safer option of downloading the blockchain, in order to replace the requests made on *chain.so* explorer. We will also implement an inclusion mechanism, to verify that a given decision has remained unchanged. Moreover, the verifier does not take into account the data available through the SPARQL endpoint, meaning that a modification to this data will go unnoticed. We will extend the verifier with the option to perform a full verification procedure which will ensure that data offered through SPARQL endpoint is the same with the compressed Notation3 decisions and therefore same with the stamping transactions of bitcoin.

References

1. Chalkidis, I., Nikolaou, C., Soursos, P., Koubarakis, M.: Modeling and Querying Greek Legislation Using Semantic Web Technologies. In: The Semantic Web - 14th International Conference, ESWC 2017. Volume 10249 of Lecture Notes in Computer Science. (2017) 591–606
2. Cucurull, J., Puiggali, J.: Distributed Immutabilization of Secure Logs. In: Security and Trust Management - 12th International Workshop. Volume 9871 of Lecture Notes in Computer Science., Springer (2016) 122–137

3. Casanovas, P., Palmirani, M., Peroni, S., van Engers, T.M., Vitali, F.: Semantic Web for the Legal Domain: The next step. Semantic Web (3) (2016) 213–227
4. Hoekstra, R.: The MetaLex Document Server - Legal Documents as Versioned Linked Data. In: The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference. Volume 7032 of Lecture Notes in Computer Science., Springer (2011) 128–143
5. Frosterus, M., Tuominen, J., Wahlroos, M., Hyvönen, E.: The Finnish Law as a Linked Data Service. In: The Semantic Web: ESWC 2013 Satellite Events - ESWC 2013 Satellite Events. Volume 7955 of Lecture Notes in Computer Science., Springer (2013) 289–290
6. ELI Task Force: ELI implementation methodology. `http://data.europa.eu/doi/10.2830/813167`
7. Faísca, J.G., Rogado, J.Q.: Decentralized Semantic Identity. In: Proceedings of the 12th International Conference on Semantic Systems, ACM (2016) 177–180
8. Sutton, A., Samavi, R.: Blockchain Enabled Privacy Audit Logs. In: The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference. Volume 10587 of Lecture Notes in Computer Science., Springer (2017) 645–660
9. English, M., Auer, S., Domingue, J.: Block Chain Technologies & The Semantic Web: A Framework for Symbiotic Development. In: Computer Science Conference for University of Bonn Students, J. Lehmann, H. Thakkar, L. Halilaj, and R. Asmat, Eds. (2016) 47–61
10. Berners-Lee, T.: 5 star deployment scheme. `https://www.w3.org/DesignIssues/LinkedData.html`
11. Archer, P., Goedertier, S., Loutas, N.: Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC. `https://joinup.ec.europa.eu/sites/default/files/document/2013-02/D7.1.3-StudyonpersistentURIs.pdf` (2012)
12. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System (2008)
13. Bitcoin Wiki: Script Manual. `https://en.bitcoin.it/wiki/Script`
14. Bartoletti, M., Pompianu, L.: An Analysis of Bitcoin OP_RETURN Metadata. In: Financial Cryptography and Data Security - FC 2017 International Workshops. Volume 10323 of Lecture Notes in Computer Science., Springer (2017) 218–230
15. Garay, J.A., Kiayias, A., Leonardos, N.: The Bitcoin Backbone Protocol: Analysis and Applications. In Oswald, E., Fischlin, M., eds.: Advances in Cryptology - EUROCRYPT 2015. Volume 9057 of Lecture Notes in Computer Science., Springer (2015) 281–310
16. de Pedro Crespo, A.S., García, L.I.C.: Stampery Blockchain Timestamping Architecture (BTA) - Version 6. CoRR (2017)
17. Gutoski, G., Stebila, D.: Hierarchical deterministic Bitcoin wallets that tolerate key leakage. IACR Cryptology ePrint Archive (2014) 998
18. Palatinus, M., Rusnak, P.: Multi-Account Hierarchy for Deterministic Wallets. `https://github.com/bitcoin/bips/blob/master/bip-0044.mediawiki`
19. Fernández, J.D., Martínez-Prieto, M.A., Gutiérrez, C., Polleres, A., Arias, M.: Binary RDF Representation for Publication and Exchange (HDT). Web Semantics: Science, Services and Agents on the World Wide Web **19** (2013) 22–41
20. Sporny, M.: LD-DL'17 Workshop Keynote Talk: Building Better Blockchains Via Linked Data. In: Proceedings of the 26th International Conference on World Wide Web, ACM (2017) 1429
21. Buterin, V.: A Next-Generation Smart Contract and Decentralized Application Platform. `https://github.com/ethereum/wiki/wiki/White-Paper`