dARK: *A decentralized blockchain implementation of ARK Persistent Identifiers*

Washington Segundo[1*] Lautaro Matas[2†], Thiago Nóbrega[3*], J. Edilson S. Filho[4*], Jesús Mena-Chalco[5‡]

Abstract

Persistent identifiers (PIDs) have become an essential part of the scholarly publishing ecosystem especially for assuring Open Science good practices. Over the years several persistent identifier systems have been adopted with different levels of coverage and sustainability. Most of them are based on centralized models, depending on a few agencies that support the services infrastructure. The ARK identifier, one of the known PIDs, emerged as a viable low-cost solution due to the possibility of implementing in-house providers to the global resolver. Nevertheless, known ARK implementations are mostly centralized, isolated, *in-house* in single or small groups of institutions. In this work, we present a technology that allows a low-cost and decentralized procedure for assigning ARK persistent identifiers, providing an open/non-centralized *permissioned* public identifier factory and resolver service. dARK is an ARK implementation built on top of a decentralized network based on institutional *blockchain* nodes. In this way, the data is owned, stored, and controlled by no single organization but by all the participants of the network. The dARK first minimum viable product implementation shows the feasibility of the concept by assigning and resolving identifiers of scientific production artifacts. This system runs on a community-supported infrastructure without depending on a centralized organization. In particular, decentralization was tested with data from digital library sources, identifying objects that represent scientific articles, demonstrating the feasibility of the concept. The proposed system has a flexible metadata schema, allowing the attribution of PIDs for, potentially, any type of digital/physical object.

Index Terms

Archival resource key (ARK), Persistent Identifier, Metadata, Blockchain.

[1] washingtonsegundo@ibict.br; [2] lautaro.matas@lareferencia.redclara.net ; [3] thiagonobrega@ibict.br;
[4] josefilho@ibict.br; [5] jesus.mena@ufabc.edu.br
[*] Instituto Brasileiro de Informação em Ciência e Tecnologia (IBICT)
[†] Red de Repositorios de Acceso Abierto a la Ciencia (LA Referencia / RedCLARA)
[‡] Universidade Federal do ABC (UFABC)

Contents

I. Introduction

IDentifying objects across digital and physical media became essential for scholarly communication, in plenty of knowledge domains [1]–[5]. One method to identify entities over multiple sources is to assign unique object identifiers and later use them to refer to these entities across different systems. These identifiers are called Persistent Identifiers (PID). In the scientific context, PIDs are essential for several aspects, for instance: citation; credit and authorship; patent applications; knowledge provenance; and validation [6]–[8]. In the current research ecosystem PIDs (ie: DOIs) are the key to identifying articles, books, authors, documents, files, databases, samples, art objects, and pages, among other scientific digital objects, across the national, regional and global open science ecosystems [6].

Current PID's services rely mainly on non-profit community-driven organizations. The data is stored and controlled by a small number of agencies, they provide the PIDs attribution and resolution, and also implement services for research organizations. Research organizations usually pay a contribution to sustain services and the needed infrastructure. The operational and maintenance costs are covered using different models, such as annual fees, memberships, PID creation charges, among others. Sometimes these costs are a barrier for small institutions, especially in the global south. As a result, there is a lack of PID coverage in some regions of the globe.

One alternative for assigning and resolving PIDs are ARK identifiers. ARK model will be covered in a separate section, but shortly the core concept behind is that the identifier attribution process is managed in a self-sufficient manner. It does not need a central authority to generate a new identifier, mainly because the ARK does not need external information to assign a new unique identifier. Therefore, some institutions implemented in-house ARK solutions [9], this can be more affordable for some institutions but brings other problems and limitations, such as data loss risks and data duplication (different ARKs for the same object in different institutions). Furthermore, PID systems should persist even if the hosting institution/company runs out the business, malicious agent issue ransomware attacks [5], or even if natural disasters or terrorist attack occurs. Some of the mentioned incidents could be mitigated by costly backup and high availability (HA) strategies. Notice that these strategies are not affordable for most small and medium science organizations (e.g., universities, scientific associations, and non-commercial editors).

Preservation and recovery support concerns the system's capabilities in providing continuous and reliable data access over time even in the worsts scenarios [10]. Decentralized implementations are more fault tolerant since the different copies (of fragments) of the database could be stored in different places ensuring a reliable and continuous service [11].

In both approaches, agencies or in-house solutions, there are issues on detecting different versions of the same object and reusing identifiers over different sources. Most systems do not provide a built-in mechanism to identify if the same real-world object (e.g., different PDFs for the same paper) is stored in different places [10]. Solving this issue requires complex and expensive data integration tasks that currently are carry out by big infrastructures [12]. A decentralized architecture integrates data from different institutions, even interoperating with other PID providers, and then is able to check duplication on PIDs real-time creation or running asynchronous processes, providing a mechanism to deduplicate different records into a single identifier.

From the best knowledge of the authors, this work presents the first (prof of concept) implementation of an open decentralized PID service conceived from the very beginning as a public good for the regional *Open Science* ecosystem. The technology behind, *blockchain*, allows governance and low costs to be shared by national and regional research/education networks in collaboration with individual research institutes/universities, with the provision of small computational resources.

Summarizing, in the next sections we present *Decentralized Archival Resource Key* (dARK), a decentralized implementation of the ARK PID. The basic idea of dARK is to execute and store the PID metadata required by the ARK model in a decentralized environment, *blockchain networks*. We propose to execute the ARK over every node of a *Blockchain Consortium Network* (BCN) by implementing the dARK as Blockchain Decentralized Application (DApp). This architecture layout mitigates issues related to continuous access to the PID system over time and creates built-in mechanisms to integrate PIDs from different sources at a low cost for the organizations that join the BCN. As a collateral benefit of using blockchain technology, dARK delivers native PID provenance, adding a new layer of trust to the PID metadata.

Section II presents a light background and relevance of PIDs in the scholarly publishing ecosystem, with a brief description of the ARK PID System. Sections III and IV introduce, respectively, the basic concepts of decentralized systems and blockchain consortium networks. Section V details the dARK implementation and its functionalities, including a proof of concept. Finally, in Section VI we provide our final considerations and present future work.

II. The Archival Resource Key (ARK)

The way of doing science, as well as the way of making it public and disseminating it, has changed in the last century. The standard publishing model has undergone several transformations, ranging from the more traditional format (document printed and distributed in the form of books and journals) to more alternative formats (for example, HTML 5 structured documents) [13]. In this context, Persistent Identifier (PID) systems became the way of identifying and citing digital objects providing uniqueness, integrity, and trustworthiness regardless of the identifier's application domain. In order to fulfill such requirements, PID systems also have to guarantee continuous access to the data over time [10]. In the past two decades, various communities and companies have developed PID services, employing different technologies and policies to address issues of various domains. Some known examples are i) the Digital Object Identifier (DOI) [14]; ii) the Handle Net System (hdl.net) [15]; iii) the Archival Resource Key (ARK) [16]; iv) the Persistent Uniform Resource Locator (PURL) [17]; iv) the ORCID ID [18]; v) the International Standard Serial Number (ISSN) [19]; vi) the International Standard Book Number (ISBN) [20]; and vii) the International Standard Name Identifier (ISNI) [21], Lens MetaRecord ID (LensID) [22], Research Organization Registry (ROR) [23], among others.

According to [16], ARKs (Archival Resource Keys) are a low-cost and flexible alternative for assigning persistent identifiers. Any organization has the possibility to create an unlimited amount of identifiers using a flexible metadata scheme. Nevertheless, most ARK implementations rely on in-house solutions that are isolated from each other, bringing some issues such as PID duplication, cost inefficiency, and low fault tolerance.

Archival Resource Keys (ARKs) is an open PID system that provides trusted references for information objects. ARK is a widely used PID system and has been used by more than 900 organizations. This organization uses the ARK to generate more than 8 billion identifiers, from digital objects (e.g., documents and databases) to physical objects (biological samples and artwork), and even living beings (e.g., people and orchestras) and intangible objects (places and vocabulary terms).

To be used in various applications mentioned above, the ARK was conceived to be generic. Furthermore, the fundamental design of ARK are:

1) High-density identifiers, permitting opaque and shorter identifiers;
2) Self-sufficiency, using Noid (Nice Opaque Identifiers) and a simple HTTP Server, it is possible to execute an ARKs servers;
3) Flexible metadata schema;
4) Able to provide identifiers without metadata;
5) Capable of assigning additional identifiers (e.g., a DOI) to an object;
6) Provide a query mechanism when resolving the identifiers;
7) Provide global resolvability (Name-to-Thing resolver);
8) Affordable, there are no fees to assign ARK PIDs.

These designs were incorporated in ARK and distributed in two main components; the ARK identifiers and the ARK system architecture. The core concept of the ARK identifier is that the identifier attribution process was created considering a self-sufficient away. The ARK does not need a central authority to generate a new identifier, mainly because the ARK does not need external information to assign a new unique identifier.

To enable self-sufficient identifiers, the ARK incorporates Name Assigning Authority Numbers (NAANs) identification in the object identifier. An organization (e.g., an university) must be registered within the ARK Alliance to generate an ARK identifier. In this registration process, the organization will receive a NAAN identification, e.g., the *Instituto Brasileiro de Informação em Ciência e Tecnologia* (IBICT) has the NAAN 80033. The NAAN is used as the prefix of the ARK identifier, making it impossible for any other organization to generate IDs with this prefix.

A typical ARK identifier comprises four parts, NAAN, *shoulder*, *blade*, and *tip*. Figure 1 represents hypothetical ARK identifier[1].

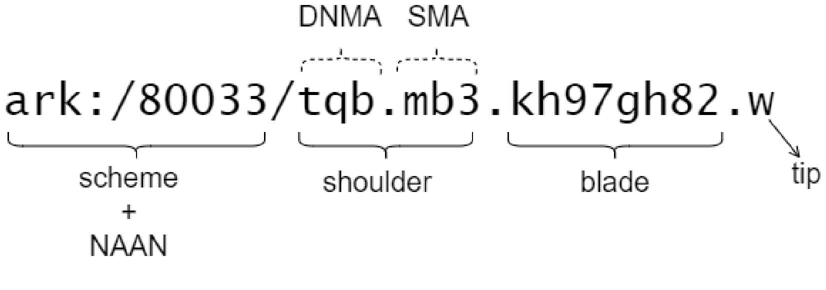


Fig. 1. ARK identifier

[1] Initials DNMA and SMA in Fig. 1 will be futher explained in Subsec. V-B

The ark:/, designates the schema of the identifiers. The 80033 is the NAAN responsible for the identifier. The shoulder (in Figure 1 the shoulder is tqbmb3) designates the organization unit (e.g., university department) responsible for the identifier, defining sub-namespace. Unique shoulders guarantee that names within the sub-namespace will also be unique outside of it, and the unit need only focus on creating the rest of the key in a unique way. The blade (kh97gh8) is a unique id generated by the unit. Finally, the tip (w) is the verification digit of all parts of the ARK identifier.

Notice that the self-sufficiency of the ARK identifier is ideal for decentralized systems. This characteristic, combined with the absence of fees, flexible metadata schema, and the open-source nature of the ARK, are the requirements that make a perfect match with the desired solution, which is more detailed explained in Section V.

Regarding ARK system architecture, it is resumed to a simple HTTP server with an application capable of querying a local database populated by the ARK identifier of the authority. To provide a universal query point, the ARK uses a well-known public resolver, Name-to-Thing (N2T).

The resolver is a simple redirecting web server that forwards the request to the ARK query web server responsible for the NAAN. The resolver redirects the request for a persistent identifier to the adequate web server. Figure 2, illustrate the resolving process.

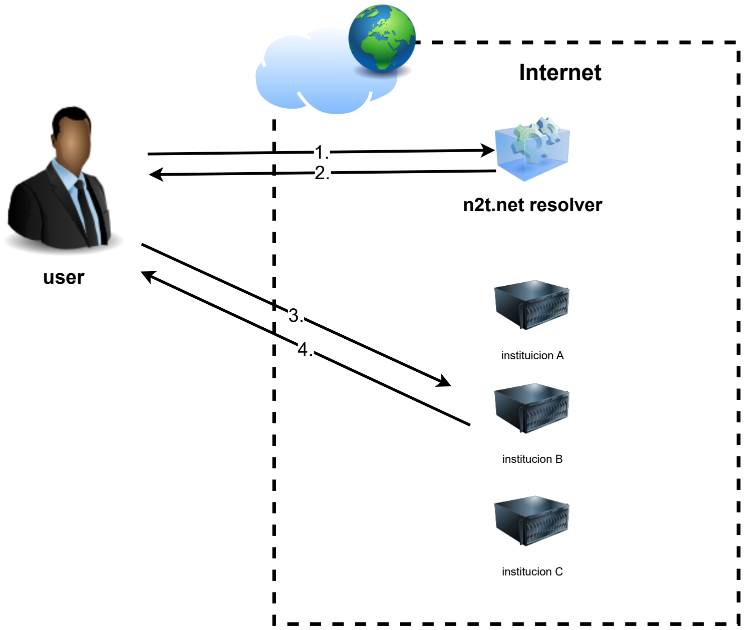


Fig. 2. n2t.net resolving process.

A user requests access to an ARK identifier to the n2t.net resolver. Knowing that the resolver does not hold the object data, it forwards the request to the appropriate organizational web server. Finally, the organization's web server handles the request and answers the user the ARK identifier data.

Notice that the resolver will fail for any identifier that a provider (organization web server) fails to resolve at the local level. The approach of a decentralized version of ARK should mitigate this very common problem (see Section V).

III. Decentralized Systems

Before presenting the notions of blockchain systems, one first needs to disclose the concepts and differences between centralized and decentralized systems in our context. If the reader is familiarized with these concepts, he/she can jump straight to Section IV.

Other paradigms stand out, such as distributed persistent identifiers systems. Using distributed databases decreases the need to commit to the long-term maintenance of the supporting infrastructure and allows resolution mechanisms to remain functional in critical events [24]. Several works, for instance [10], [25], [26], investigated the use of decentralized technology for improving PID systems. These works employ *peer-to-peer networks* [10], and *distributed ledger* (blockchain) technology [25] to create novel PID systems.

The IBI system, described in [27], has an interesting approach and consists of distributed layers of PID assignment. A counterpoint that dARK offers, beyond getting the aforementioned benefits of using blockchain networks, is that it also inherits the best properties of a globally known PID system, the ARK, in contrast with IBI which has a national scope. On another hand, the existent decentralized PID technologies (e.g., blockchain and peer-to-peer networks) [10], [25], [28] have a key difference from dARK, that is it does not propose a novel PID system; dARK is an extension of the ARK PID system. Moreover, dARK is also compatible and acts as a proxy for existing PID systems, such as DOI, IBI, ISSN, and others.

The majority of computer systems and applications considers that a central authority must control the system's data and/or functions [5], in other words systems and application assume a centralized management system. For instance, financial organizations (e.g., banks) and even Google Search, are considered centralized because there is an organization at its heart that runs the entire business, including data, designers, programmers, and advertising experts. In summary, in our context, one central authority controls and orchestrates the operations for a system to be classified as centralized. For example, a company executive board or a central node in a Database Management System (DBMS) Cluster could represent this central authority.

Centralized systems present advantages and disadvantages. Regarding the advantages of this system, we can list more straightforward implementation and management at lower costs. It is simpler and more affordable for organizations that do not need to share their operation/data to use this model. On the other hand, the disadvantages of centralized systems mainly concern the lack of transparency and single-point failure. Commonly, centralized systems do not provide transparency mechanisms, making it hard for other systems to reuse and interoperate. Moreover, as presented in Section I, a centralized system requires a complex backup and HA system to address the single point of failure disadvantage. Finally, it is worthwhile to comment on the existence of no technological failure; for instance, if a local newspaper closes, this valuable location's history could be lost in one hard drive. Furthermore, this kind of failure could also happen in the PID system context.

Decentralized systems emerged to mitigate the limitations of centralized systems. Instead of using a central authority to execute and orchestrate operations, these operations are performed by the nodes of a decentralized network. In this scenario, the system's control is equally shared among the network nodes, using cryptographic and security protocols. It is worth mentioning that within the decentralized system and blockchain context, operations are named and modeled as transactions.

By considering a decentralized system, we address the disadvantages of the centralized systems mentioned above, and we also have additional advantages:

- *Enhanced security*: Due to robust cryptographic tools employed in the core of these systems, every user must be identified and sign every transaction using their cryptographic keys. This characteristic adds a layer of security to the systems;
- *Transparency and trustworthiness*: Every transaction performed in this system is verified by every network node. All transactions and data stored in these systems are auditable, making the system transparent and trustworthy;
- *Tamper evidence guarantees*: Another benefit of these systems is that stored data can not be modified once it is stored. In a centralized system, anyone at the top of the hierarchy who has authorization is able to make changes;
- *Built-in Data Provenance*: once that data is immutable, and every transaction indicates who and when it was executed, these systems have a built-in data provenance system.

These advantages could be employed to improve the existing PID systems. For instance, the data transparency characteristic could be used to promote PID integration mechanisms, and the benefit of data provenance could be utilized in the PID context. Furthermore, regarding the single point of failure issue, considering the tamper evidence guarantees and the fact that a decentralized system replicated to every node of the network will mitigate this issue by design.

In the following section, we present the blockchain concepts, the decentralization technology used to build our decentralized PID system (dARK). This section explains the technical details that make all of the aforementioned advantages and characteristics possible.

IV. Blockchain

Blockchain, also called *distributed ledger*, is essentially an append-only DBMS maintained by a set of nodes that do not fully trust each other [29], [30]. Blockchain is a technology that maintains the states and the historical transactions, using a peer-to-peer network, without any central node to enforce compliance. Blockchain provides immutable storage (tamper evidence guarantee) of transactions in a chain of blocks, by storing data (e.g., PID records) in blocks that are linked using cryptography tools, i.e., the previous block hash, the transaction owner signature, and the identifier (id) of the machine (miner) that executed the transaction. Figure 3 provides a high-level illustration of the Blockchain internal elements.

For each block of the blockchain, Figure 3 presents a brief illustration of the most relevant data structures of the technology. Notice that each block contains the indication (hash[2]) of the previous block to maintain the order of the data and transactions. A critical element is a nonce, a considerably large random number (32 to 4,096 bits). The nonce represents a cryptographic puzzle that needs to be solved by a blockchain node (a.k.a., miner). Then, with this puzzle resolved, the miner can append the block to the chain in a

[2] The term hash refers to the result of a function called hash, which is a cryptographic operation that generates unique and unrepeatable identifiers from information provided

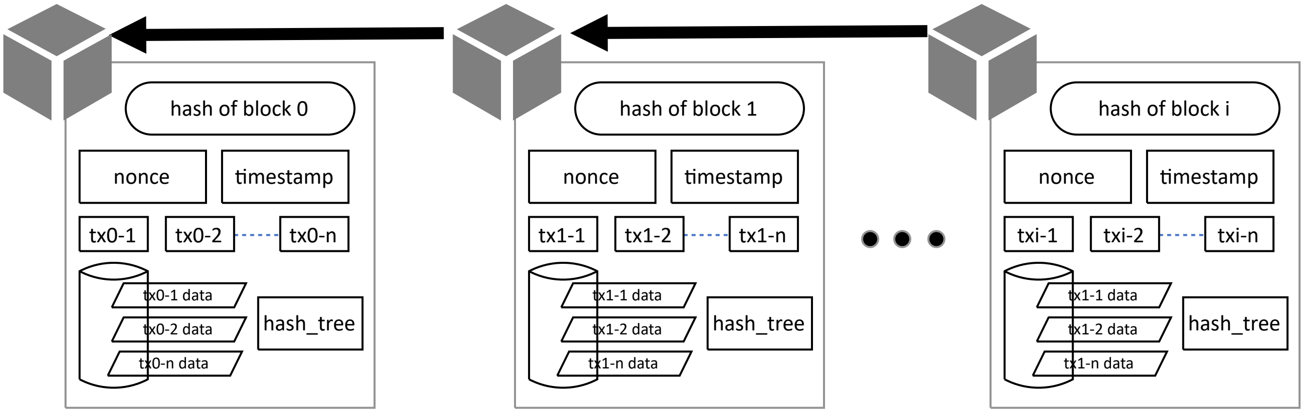


Fig. 3. Blockchain internal structures.

process named mining. For instance, *bitcoin* miners need to guess a valid nonce as they perform multiple attempts to calculate a block hash that meets certain requirements (i.e., a prime number with more than 64 bits that starts with a certain number of zeros).

Another important element is the timestamp; this element is used to provide proof of computing (or existence) to the block data. For instance, this element can be used to track public records, such as Educational Degrees[3].

The data are stored in a public databaseby the transactions (tx). Notice that one block can execute multiple transactions. It is worth mentioning that the data and metadata (e.g., the miner executing the transaction) are stored in the database.

For a record (block) to be effectively stored in the distributed ledger, the record needs to be replicated in every node of the blockchain. In order to guarantee the consistency of the ledger, a party sends the record within a transaction with a cryptographic puzzle to be mined by the blockchain nodes. After the first miner solves the cryptographic puzzle (embedded with the transaction), the other nodes will verify the transaction execution and store it in its local ledger [31]. Every node in the network verifies all transactions, replicating the data and the computation in all nodes of the network. Therefore, all inputs and outputs of the transaction are publicly available from every blockchain network member.

Blockchain consortium network

As already presented, blockchain technology has several characteristics and can work in a decentralized way. An interesting proposition when talking about the use of blockchain is the concept of *blockchain consortium*.

A blockchain consortium is a type of permissioned network-style semi-decentralized network where nodes or members join the network through a regulatory entity. In general, this system is based on voting to ensure low latency and high speed rates. Each node is allowed to write transactions, but cannot add a block by itself and each block added by another node needs to be verified before adding it to the network. Figure 4 represents an example of a consortium formed by 4 organizations. All nodes communicate with

[3] http://www.ufpb.br/antigo/content/primeiros-diplomas-digitais-da-ufpb-ser\%C3\%A3o-entregues-formandos-do-centro-de-inform\%C3\%A1ticahttp://www.ufpb.br/primeiros-diplomas-digitais-da-ufpb/

each other. And in the eventual even number of organizations, other consensus algorithms other than simple majority voting can be chosen.

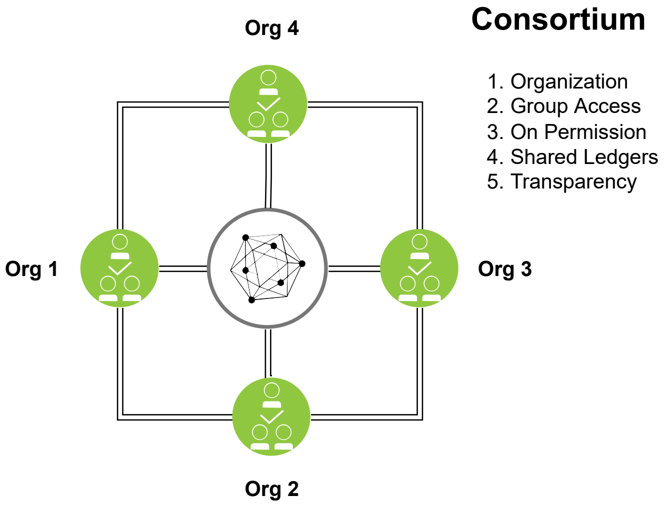


Fig. 4. Blockchian consortium.

Note that in the consortium system, each member or partner organization can make information available to the public outside the network. One or more organizations can be highlighted as a regulatory agent or moderator, responsible, for example, for analyzing the entry of new members, monitoring the quality standard and governance of the network.

Table I summarizes some advantages in adopting the blockchain system in consortium for the proposed project.

TABLE I
BLOCKCHAIN CONSORTIUM ADVANTAGES.

Nº	Description
Decentralized	Although only consortium members have access to the blockchain, unlike public blockchain networks, it is easier to reach a consensus and the participating nodes were somehow pre-selected.
Control	It is much easier to create rules to govern the use of blockchain. Instead of a single entity, a specific group of authentic participants controls the blockchain.
Transaction speed	As fewer nodes are involved compared to public and private blockchain networks, transactions are carried out at a much faster speed.
Monetization	You can create your own monetization rules. An example is the creation of tokens.
Light	By not needing to perform mining, the computational power of the system is focused to perform routine operations (like writing and reading in the Blockchain).

It is necessary to point out that even if the blockchian permissioned (e.g., a private blockchain network), organizations can create forms of access that allow the general public to make queries and add information to the system.

In our project we defined that the best architecture is the consortium format, as it allows the business rules for a persistent identifier of data using blockchain technology to be more easily built and shared.

V. The Decentralized Archival Resource Key (dARK)

dARK is a Decentralized Application (DApp) that runs in a *blockchain consortium network*, an extension of the *Archival Resource Key* PID system (see Sec. II). The key concept of the dARK is to employ an approach that allows multiple institutions to handle (collaboratively) their PID system (e.g., assigning and reusing persistent identifiers) using a common decentralized infrastructure. Moreover, the dARK was built on top of a network based on institutional blockchain nodes. In this way, the data is owned, stored, and controlled by no single organization but by all the participants of the network. It is worth mentioning that there are multiple ways to create a BCN. For instance, institutions can use commodity hardware to install blockchain nodes. Another possibility is to use a blockchain as a service provided by the LACChain[4] and the Brazilian Rede Nacional de Pesquisa (RNP)[5]. By extending the ARK in the refereed decentralized manner, one intents to match some requirements: i) PID system fault tolerance; ii) Long-term PID data preservation; iii) PID data provenance; and iv) PID system interoperability.

Moreover, the adopted solution (dARK) has an advantage regarding compatibility with other PID systems. It aggregates an attribute to the dARK record that allows recovering the data directly via dARK URI or via URIs originally assigned/adopted by other PID systems. For example, a publication record that already has a DOI can be recovered, resolving a dARK or a DOI URI.

Considering that dARK users will join a well-organized consortium blockchain network, the PID data will be replicated in a secure, reliable, auditable, and affordable manner, across every node of this network. Even if the institution disappears (e.g., loses all of its funding and closes), the data will be preserved in the remaining network nodes. Therefore, the data preservation and management techniques (detailed in Section IV) will be responsible for mitigating data loss issues.

Furthermore, considering the consortium blockchain network, the cost of storing and managing the PID data will be low. Section IV describes that the cost of operating a consortium/private blockchain network is meager mainly because this network does not suffer the influence of cryptocurrency markets, the only cost will be storage and medium port computational infrastructure. Roughly speaking, for each dARK PID assignment (detailed in Section V), it is considered 12Kb of disk storing. Moreover, it was designed a standard server instance for running the decentralized application (e.g., in our experimental environment, it was used a 4-Xeon-core virtual machine, with 16GB RAM with 30GB of storage). It is important to mention that in Section V-D, we present further detail of the technology (e.g., blockchain framework) employed in the developed solution.

In order to improve the transparency (and, therefore, thrust) of the PID data, dARK leverage the blockchain data immutability (once the data is written on the blockchain, it can not be modified, see Section IV for further details) to provide data provenance capabilities. The dARK takes record of every operation performed along the time for every PID, making it possible to know *what*, *who*, and *when* every change is performed, completely mitigating issues regarding PID data provenance. This characteristic can provide a new level of transparency and trust for the PID systems.

In interoperability, with the capabilities of different user agents in reusing a PID, one has the traditional PID systems, such as ARK, commonly used just to identify a object with its respective description.

[4] https://www.lacchain.net

[5] https://www.rnp.br/noticias/rnp-oficializa-participacao-na-rede-blockchain-brasil

However, the interoperability concept that we are proposing is broader. Moreover, once the PID data are available in the blockchain, methods/techniques (such as *record linkage* [2], [32], and *data mining* [32]) could leverage the PID metadata to be employed in other repository (managed by another institution) in the network to improve metadata quality. For instance, an electronic journal can reuse metadata from publication authors (eg. external PIDs, such as ORCID ID or even an email address) that are already in the blockchain. The system can also links authors to their affiliations if the respective organization records are already recorded in the blockchain network. This construction is very similar to a PID research graph [33]. In the following section, we present the basic architecture of dARK and how it works, by presenting some use cases.

A. dARK Service Architecture

This subsection provides further detail on how the dARK and the consortium will operate. The intent of the dARK architecture design is to achieve the aforementioned goals by using blockchain technology that will replicate the PID metadata across every node of the consortium network. Moreover, due to the *Turning complete capabilities* (detailed in Section IV), cryptographic and security protocols of the blockchain technology, the dARK will execute reliably and securely, regardless of the network node (machine) executing it. Therefore, describing how the consortium will operate and interact with the general ARK ecosystem is crucial to a better understanding of the contribution of this work. Section II details the interaction between the ARK Server and the public resolver. Figure 5 illustrates the interaction between the public resolver (n2t.net) and the dARK consortium network.

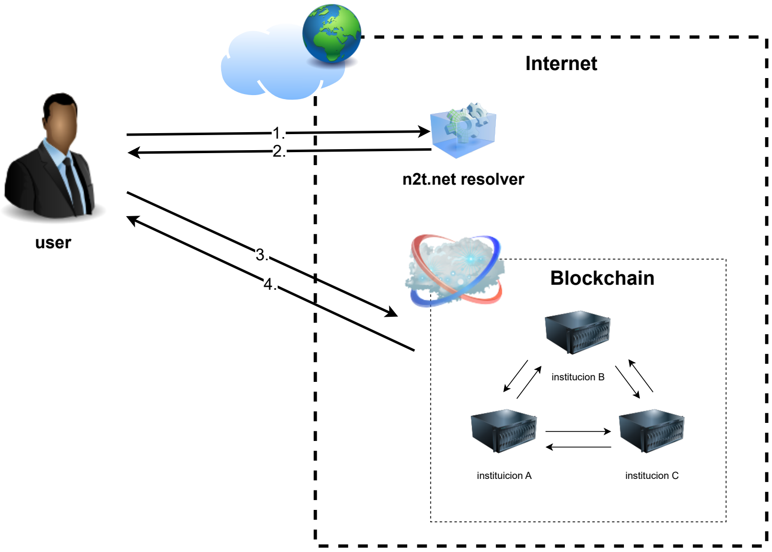


Fig. 5. dARK Service.

The initial steps of the interaction (1 and 2) are the same the user asks the public resolver for the address of the server responsible for the identifier. The resolver uses the NAAN and answers the user with the server address. Notice that, by considering the dARK, the public resolver will return the address of the blockchain network instead of a unique server address. For instance, if the dARK ID represented in Figure 1 is employed in a query to the n2t resolver, the resolver will answer with the blockchain network consortium address. This address is routed in the network using a *consortium resolver address*. It is important to mention that the blockchain network stores and processes every action of the PID system. The consortium resolver is a simple HTML web page that points to the blockchain making it inexpensive and easy to employ high availability and load-balancing strategies with the consortium members. In step 3 and 4 of Figure 5, the user will use the consortium resolver address and will receive the answer from any node of the blockchain network.

B. dARK Identifiers

To enable the decentralization of the ARK, first, we have to extend the ARK identifier to be compatible with the new organizational structure of the consortium of institutions that will use the dARK. Thus the proposed extension is fully compatible with ARK identifier assigning rules. This provides an automatic compatibility with standard ARK interoperability protocols. In this structure the ARK Alliance [34] is able to issue just a unique NAAN prefix to an entire blockchain consortium network. For instance, in Brazil, a consortium should be founded by a federal governmental research institute related to science and technology, the *Instituto Brasileiro de Informação em Ciência e Tecnologia* (IBICT)[6].

In order to decentralize the ARK identifiers, the dARK system was designed with PID management capabilities for the curators of each institution that composes the consortium. To accomplish the aforementioned goal, two new elements were incorporated into the shoulder of the ARK identifier, the *Decentralized Name Assigning Authority* (DNA) and *Section Name Authority* (SNA).

The DNA is a shoulder prefix employed to identify which consortium member is responsible for a specific PID assignment. DNA is similar to a NAAN; however, it identifies the institution (authority) responsible within the consortium context. The SNA is used to assign a section authority within an institution. For example, a university could indicate a curator for different libraries by assigning different SNAs for each collection. Figure 1 presents an example of an assigned identifier.

Notice that a dARK identifier has the same elements disciplined by ARK (scheme, NAAN, shoulder, blade, and tip) as the regular ARK identifier. However, the shoulder is wider to accommodate the two new elements, DNA and SNA. In addition, notice that the shoulder prefix is ended with a number that is used to delimiter the places for these two new referred strings. In our implementation (see Sec. V-D), the DNA and SNA are previously configured with fixed lengths.

Notice that the blade (representing the object) and tip (verification digit) are generated according to the NOID and the Noid Check Digit Algorithm (NCDA) algorithms[7]. A decentralized implementation of these Algorithms is provided in a code repository[8]. In the following section, we provide further detail of the dARK record description.

[6] https://www.gov.br/ibict/pt-br

[7] http://n2t.net/e/noid.html

[8] https://github.com/ceois-ibict/

C. dARK Record Description

In the previous subsection, it was detailed how the dARK ID is generated. This section details the data structures used to manipulate the information in the dARK system. It is worth remarking that ARK is an *out-of-the-box* flexible PID system that does not enforce any metadata schema in particular. It is even possible to create an ARK ID containing no descriptive metadata. Thus dARK follows the same agnostic metadata approach.

Moreover, dARK must be compatible, acting as a *proxy*, for multiple PID systems (e.g., DOI, ORCID, PURL, ISSN and others). An *external PIDs* dARK ID record attribute stores information from different PID systems, so it is a key component in integrating dARK system with other PID systems. Furthermore, dARK must also provide reliable query capabilities over the stored metadata.

Complying with the dARK requirements and addressing the existing PID system limitations, several data structures are proposed in order to store and manipulate information. Figure 6 illustrates one of these data structures. The dARK object is responsible for associating the generated PID to its metadata.

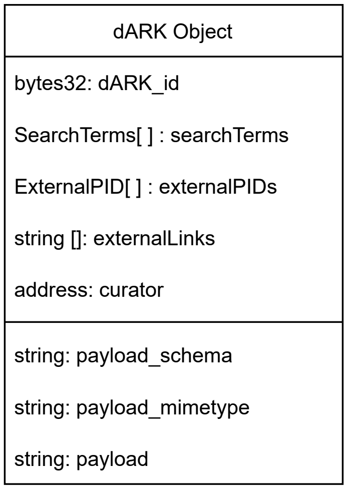


Fig. 6. dARK Object structure.

Besides storing necessary PID metadata, the dARK object is important to secure capabilities like: i) enhanced and efficient storage capacity; ii) metadata schema-agnostic; iii) query mechanisms; iv) native data quality mechanism; v) compatibility with other PID systems; and vi) data provenance. In the following, it is detailed how the data structure (depicted in Figure 6) was employed to achieve the aforementioned capabilities.

The dARK ID is represented as a 32 bytes field. Then, hash functions are used to encode that dARK ID and store the hashed identifier. By storing the id's hash, the dARK can store a significant amount of PIDs using minimal computational resources. In this approach, one can represents $(2^{32\times 8}-1)$ different identifiers per institution, approximately 19.9^{255} identifiers. It makes each institution in the dARK consortium able to assign one identifier for almost atom in the universe (10^{88}) [35].

The dARK object has three attributes: i) *payload*; *payload schema*; and *payload mime type*. These attributes designate which metadata and schema are employed in each object. For instance, one can store the metadata of an object as a JSON or a XML format description, and the *payload schema* attribute designates the schema of the stored metadata. Furthermore, one can use automatic strategies to extract the keywords from the metadata, and dARK system will perform their queries and operation using what was denominated by *search terms*. Later in this section, the use of this attribute will be detailed.

Before explaining the aforementioned dARK capabilities (iii to vi), one has to detail where and how the dARK DApp stores and uses the information of the dARK Objects. Figure 7 depicts the dARK system for two institutions (Universities A and B).

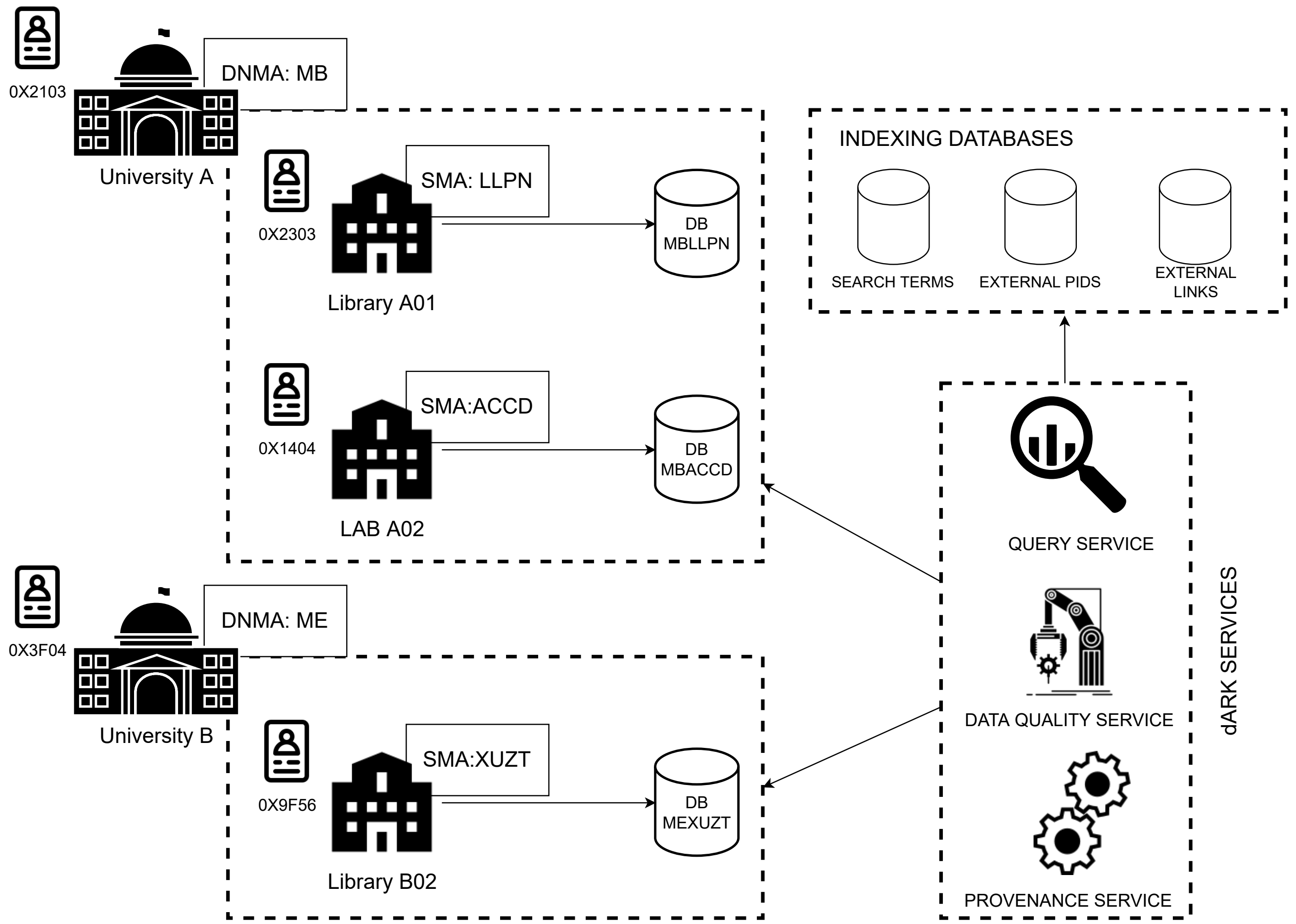


Fig. 7. dARK components overview.

In Figure 7, the responsible (or curator) for the institution/repository is designated by a *blockchain wallet* (e.g., for University A, the wallet 0x2103 has the authority over the institution). The DNA and SNA for the institutions and the sections/repositories are also plotted. As explained in the previous subsection, the DNA and SNA indicate the responsible for the record. In Figure 7, it is possible to observe that the SNA creates a database (*key-value* database) for their records. It is important to highlight that the common database considers a key-value routing model [36]. Moreover, the DNA and SNA act as routes to the correct database, making the dARK system answer queries in constant time ($O(1)$).

Notice that the information regarding the *search keys*, *external URLs*, and *external PIDs*, for every SNA, are indexed by a set of common databases (*indexing databases*). For every dARK Object created by the system, the attributes mentioned above are indexed in a common database, providing: i) a single query endpoint; ii) a native data quality mechanism; and iii) compatibility with other PID systems, regarding querying.

dARK querying mechanisms have two distinct methods. The first is using an object ID (e.g., dARK ID or DOI) to retrieve the object attributes. DNA, SNA and hashing strategies are used in order to redirect the queries to the proper database. The second method is searching for an object looking for *search terms*. For instance, one can search for scientific publications by their *author names*, *titles*, *keywords*, and other desired fields that are in the metadata description. To enable this search method, the *search terms* are criteriously configured to index searchable fields by the common database. The responses in this method are also performed in $O(1)$ time.

Moreover, besides the query capabilities, *search terms* and *external links* attributes are employed in the *data quality strategies*. It is used the information of these attributes to identify objects that they refer (or may refer) to the same real-world object. For instance, one can use the *search terms* as tokens to *record linkage blocking strategies* [3]. The dARK system uses the *external links* and *external PIDs* attributes to perform an online record linkage task. For instance, the dARK check whether or not an URL is assigned to any identifier. Thus if the URL is linked to another dARK object, the curator is notified of possible duplicated records. In future works, these strategies should be improved.

Regarding *data provenance capability*, one uses the *blockchain transaction log* (detailed in Section IV) to provide data provenance to the dARK ID assignment. The transaction log and the block attributes of the blockchain are used to identify *what*, *when* and *who* inserted a new record or changed a dARK object attribute. For instance, if the curator 0x2303 added a new search term for one of the dARK objects, the blockchain will register the: i) curator identification (address); ii) when the transaction happens; and iii) the state before and after the transaction. Therefore, the blockchain transaction logs and native data structures (e.g., *Merkel tree*) provide the necessary *data provenance capability* for the dARK system.

It is worth mentioning that in the blockchain context, wallets are used to perform *authentication*, *authorization*, and *accounting* (AAA *operations*). Moreover, AAA operations are performed employing sophisticated cryptographic tools. Every operation performed by the blockchain must be signed (using an asymmetric cryptographic function) by a wallet (e.g., curator) to be executed. This AAA model ensures enhanced security for the operations performed by the dARK system.

The concepts mentioned above were employed to design the DNA and SNA decentralized data structures. Figure 8 depicts a simplified version of this structure.

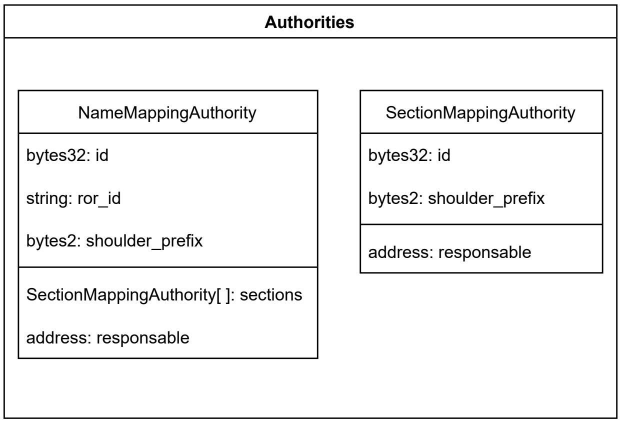


Fig. 8. dARK Auths structure.

Notice that the DNA uses the Research Organization Registry (ROR)[9] ID. Moreover, each SNA is linked to its respective DNA (*SectionMappingAuthority array*) and both entities (DNA and SNA) use blockchain wallets (represented by the address) to perform AAA operations. For example, Figure 9 illustrates the process of registering/updating a record in the system:

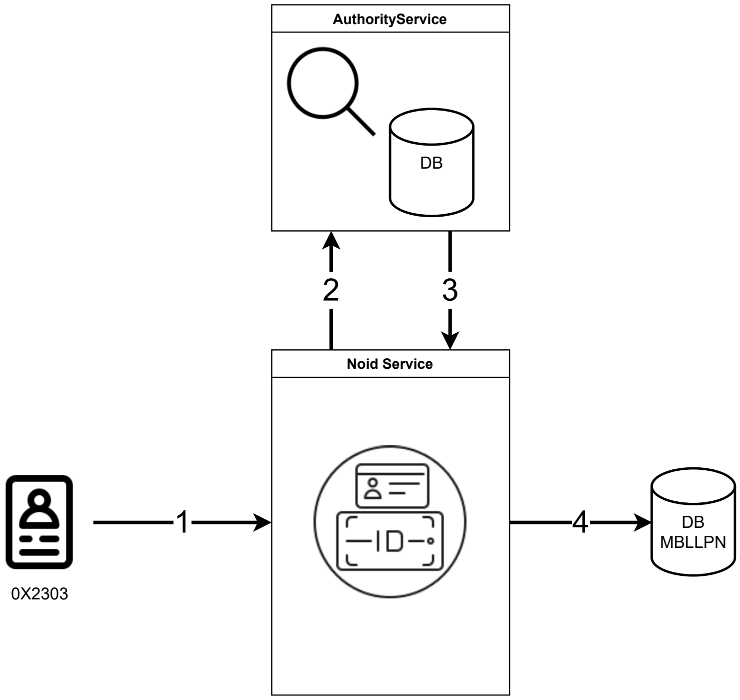


Fig. 9. dARK Authentication, Authorization, and Accounting example.

1) First, the curator (wallet id 0x2303) of Library A01 (shown in Figure 7) uses its cryptographic private key to sign a transaction to create a new dARK Id and send this message to the dARK NOID Service;
2) Then, in the second step, the NOID service uses the built-in blockchain cryptographic mechanisms to perform the AAA operations;
3) The NOID service identifies the curator, verifies if the message is valid and then checks, with the dARK Auth Service (step 3), whether the curator has privileges on the SNA/DNA.
4) The dARK Auth service employs the information depicted in Figure 8 in order to verify the permission of the curator and then forwards the authorization and the address of the database to the NOID Service, detailed in step 4;
5) Finally, the NOID service generates a new ID considering the NCDA and the NOID algorithms.

D. dARK Proof of Concept

This subsection presents some details regarding a *proof-of-concept* (PoC) of the dARK system implementation. The dARK system can be described as a *server-client* application, where the *server* is the dARK blockchain consortium network, and the *client* can be implemented in several technologies

[9] https://ror.org/

(e.g., HTML web pages, repository, or electronic journal systems). Figure 10 illustrates the architecture implemented in the PoC. It was used the *ethereum-based technology*, named *Hyperledger Besu*[10], as the blockchain consortium network. Moreover, the option for this technology was done based on some pros:

1) A widely-used open-source technology (supported by the *Linux Foundation*);
2) Able to create private blockchain consortium network;
3) Compatible with multiple open blockchain networks (e.g., *Ethereum mainnet*);
4) Provides a Turing complete Programing Language;
5) Has native libraries for several programming languages (e.g., PHP, Java, Python, Go, C/C++, JavaScript);

In summary, the *Hyperledger Besu* can be used to create blockchain consortium networks that are compatible with the requirements presented in this and the previous sections. It is important to mention that it is also provided the instructions and the containerized (*dockerfile*) solution in a GitHub code repository[11] to create the blockchain consortium network. Furthermore, this PoC was built based on four institutional nodes, working in a private blockchain.

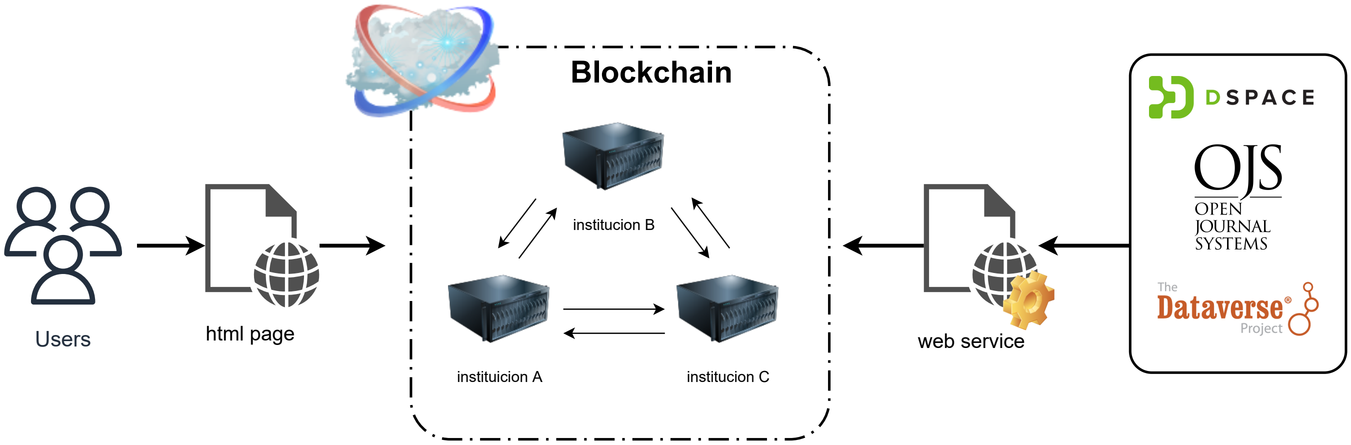


Fig. 10. dARK components.

Regarding the client side of the dARK, Figure 10 illustrates two different types of client. On the left side of the figure, one has an example where the curators (users) are acting directly over their web browsers to interact with the dARK system. It is important to mention that the page is a simple HTML+JavaScript application (see Appendix A, Fig. 12). Moreover, once every operation is performed by the blockchain (acting as a web server), the HTML page is used to test interacting with the dARK system (creating a friendly user interface). On the right side of Figure 10, it is illustrated a web service that can integrate the dARK system to existing software, like DSpace[12], Open Journal System (OJS)[13], or Dataverse[14]. The actions of interoperating via web services are the same as in the HTML page test. However, it is necessary to implement a plugin from the client side (DSpace and others), to generate the configuration

[10] https://www.hyperledger.org/use/besu
[11] https://github.com/projetos-codic-ibict/dark-algorithms
[12] https://dspace.lyrasis.org/
[13] https://pkp.sfu.ca/ojs/
[14] https://dataverse.org/

of the blockchain wallet. The plugin can be a JavaScript snippet developed for different client platforms, that can easily be added to the pipeline of the local object recording.

The process of assigning a dARK happens transparently to the user, in the way that it occurs while the human user full-fills the form of a client platform (see an example in the Figure 12). An artifact of code executes the tasks to communicate with the dARK system and generates a unique identifier, also grabbing metadata from the dARK system, that will be automatically filled in the client database (see Fig. 11). Between this information that is obtained as a response from the dARK system, is the dARK ID, and possible complementary information of other related entities that are already recorded in the dARK system. As already mentioned, during the recording of a publication, one could get information on the dARK IDs / ORCID IDs of the authors of this publication, and consequently, the dARK IDs / ROR IDs organizations that are declared as affiliations of these authors. It is not necessary for the end user to have any knowledge about blockchain. The user simply follows the step-by-step procedure that he already has the habit of doing when depositing an object in its own environment.

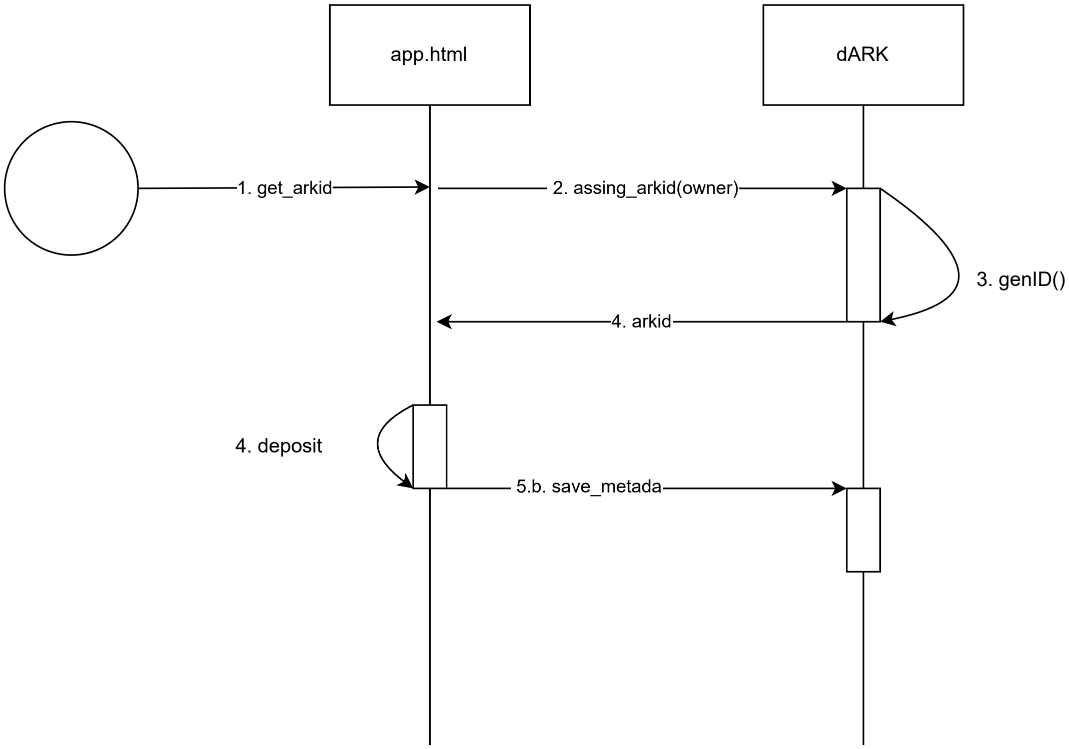


Fig. 11. Flow diagram.

After recording a publication object, an example of query response by its dARK ID is seen in Appendix A, Fig. 13. The presented JSON object has some attributes that are explained below:

1) `noid`: It is the dARK identifier suffix string;
2) `external_pids`: An array containing information of the other mapped identifiers that have already been assigned to the object record;
3) `mime_type`: The *mime-type* of the `payload` content;
4) `schema`: The metadata schema of the metadata description in the `payload` content;
5) `resource_type`: The type of the object record;

6) `resource_subtype`: The subtype of the object record;
7) `payload`: The payload of the object record;
8) `external_links`: External links URLs that point to the object record;
9) `search_terms`: The indexed terms that allows recovering the object record;
10) `owner`: The wallet address of the registrant of the object record.

As already explained in this Section, the object record can be recovered by its: i) *dARK ID*; ii) *external PIDs*; and iii) *search terms*. Moreover, queries by: iv) *schema*; v) *mime type*; vi) *resource type*; vii) *resource subtype*; and viii) *owner* are also possible.

Acknowledgements

Before concluding, we would like to thank our friends Luis Eliecer Cadenas; Leandro Ciuffo; Fábio Gouveia, for reviewing this document and making precious comments and suggestions.

VI. Conclusion

In this work, we presented a light explanation of the importance of PID and PID systems, together with a description of ARK System, its advantages and requirements, and an explanation of decentralized systems and blockchain consortium networks. Moreover, the presented key result was a detailed specification of the dARK system, and how it was implemented in a proof of concept.

Future work includes improvements in the integration between the dARK and the ARK systems, beyond an implementation with better use of the *search keys*, recovering records with more advanced queries to the PID metadata, but keeping the simplicity and fast responses of the blockchain. In addition, we plan to advance in client implementations to dARK, mainly over open-source platforms, such as DSpace, OJS, and Dataverse.

Also, massive charge tests with large datasets, such as the OpenAIRE Research Graph dump[15], should be performed, as well as the implementation of curation tasks, that would be performed by the consortium leading institutions or by a curator of an organization member.

Expanding the tests abroad is also a desire. A natural target is the LA Referencia network[16]. The presented developments are yet in the class of prototype, but we plan to launch the first production version of the dARK consortium by the end of 2023, at least at the national level of Brazil, and possibly at the Latin America level, involving other interested countries of LA Referencia.

We hope this work can contribute to the advance of Open Science, mainly in the global south, where just a small portion of scholarly communication has access to PIDs, in order to better disseminate their findings in a more efficient way.

[15] https://doi.org/10.5281/zenodo.6616871
[16] https://www.lareferencia.info/

References

[1] P. Christen, *Data Matching*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. [Online]. Available: http://link.springer.com/10.1007/978-3-642-31164-2

[2] T. P. Nóbrega, C. E. S. Pires, T. B. Araújo, and D. G. Mestre, "Blind attribute pairing for privacy-preserving record linkage," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing - SAC '18*. New York, New York, USA: ACM Press, 2018, pp. 557–564. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3167132.3167193

[3] T. Araújo, T. Da Nóbrega, C. Pires, D. Do Cassimiro, D. Mestre, and K. Stefanidis, "A noise tolerant and schema-agnostic blocking technique for entity resolution," in *Proc. ACM Symp. Appl. Comput.*, vol. Part F1477, 2019.

[4] C. Batini and M. Scannapieco, *Data and Information Quality*, 1st ed., ser. Data-Centric Systems and Applications. Springer International Publishing, 2016. [Online]. Available: http://link.springer.com/10.1007/978-3-319-24106-7

[5] T. Nóbrega, C. E. S. Pires, and D. C. Nascimento, "Blockchain-based privacy-preserving record linkage: enhancing data privacy in an untrusted environment," *Information Systems*, vol. 102, p. 101826, 2021.

[6] J. A. McMurry, N. Juty, N. Blomberg, T. Burdett, T. Conlin, N. Conte, M. Courtot, J. Deck, M. Dumontier, D. K. Fellows *et al.*, "Identifiers for the 21st century: How to design, provision, and reuse persistent identifiers to maximize utility and impact of life science data," *PLoS biology*, vol. 15, no. 6, p. e2001414, 2017.

[7] A. S. Figueiredo, "Data sharing: convert challenges into opportunities," *Frontiers in public health*, vol. 5, p. 327, 2017.

[8] S.-A. Sansone, P. McQuilton, P. Rocca-Serra, A. Gonzalez-Beltran, M. Izzo, A. L. Lister, and M. Thurston, "Fairsharing as a community approach to standards, repositories and policies," *Nature biotechnology*, vol. 37, no. 4, pp. 358–367, 2019.

[9] C. N. Ferreyra, Diego Andrés e Authier. (2020) Identificadores persistentes para a ciência argentina: Ark-caicyt, comcient: repositório institucional caicyt especializado em informação e comunicação científica, consultado em 13 de fevereiro de 2023. [Online]. Available: http://www.caicyt-conicet.gov.ar/comcient/ark%3A/16680081/rsccgr.

[10] P. Golodoniuc, N. J. Car, and J. Klump, "Distributed persistent identifiers system design," *Data Science Journal*, vol. 16, 2017.

[11] T. Tuan, A. Dinh, R. Liu, M. Zhang, G. Chen, T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1366–1385, 2018.

[12] OpenAIRE. (2000) Openaire research graph. [Online]. Available: https://graph.openaire.eu

[13] E. Herman, J. Akeroyd, G. Bequet, D. Nicholas, and A. Watkinson, "The changed–and changing–landscape of serials publishing: Review of the literature on emerging models," *Learned Publishing*, vol. 33, no. 3, pp. 213–229, 2020.

[14] N. Paskin, "Digital object identifier (doi®) system," *Encyclopedia of library and information sciences*, vol. 3, pp. 1586–1592, 2010.

[15] S. Sun, L. Lannom, and B. Boesch, "Handle system overview," Internet Engineering Task Force, Tech. Rep., 2003.

[16] J. Kunze and R. Rodgers, "The ark identifier scheme," 2008.

[17] A. Nagaraja, S. A. Joseph, H. H. Polen, and K. A. Clauson, "Disappearing act: Persistence and attrition of uniform resource locators (urls) in an open access medical journal," *Program*, 2011.

[18] L. L. Haak, M. Fenner, L. Paglione, E. Pentz, and H. Ratner, "Orcid: a system to uniquely identify researchers," *Learned publishing*, vol. 25, no. 4, pp. 259–264, 2012.

[19] D. G. Muriuki, P. K. Waweru, and A. N. Wali, "Properties of the international standard serial number," *Global Society of Scientific Research and Researchers*, 2019.

[20] F. Ayers, "The universal standard book number (usbn): why, how and a progress report," *Program*, 1976.

[21] A. MacEwan, A. Angjeli, and J. Gatenby, "The international standard name identifier (isni): The evolving future of name authority control," *Cataloging & Classification Quarterly*, vol. 51, no. 1-3, pp. 55–71, 2013.

[22] O. A. Jefferson, D. Koellhofer, B. Warren, and R. Jefferson, "The lens metarecord and lensid: An open identifier system for aggregated metadata and versioning of knowledge artefacts," *LIS Scholarship Archive*, 2019.

[23] R. Lammey, "Solutions for identification problems: a look at the research organization registry," *Science Editing*, vol. 7, no. 1, pp. 65–69, 2020.

[24] S. K. Rahimi and F. S. Haug, *Distributed database management systems: A Practical Approach*. John Wiley & Sons, 2010.

[25] E. Bellini, "A blockchain based trusted persistent identifier system for big data in science," *Foundations of Computing and Decision Sciences*, vol. 44, no. 4, pp. 351–377, 2019.

[26] N. Pritchard, "Using blockchain technology to enable reproducible science," Ph.D. dissertation, The University of Western Australia, 2021.

[27] G. J. F. Banon, "Identificador com base na internet (ibi): Sistema de identificação," *São José dos Campos: INPE*, 2011.

[28] M.-A. Sicilia, E. García-Barriocanal, S. Sánchez-Alonso, and J.-J. Cuadrado, "Decentralized persistent identifiers: a basic model for immutable handlers," *Procedia computer science*, vol. 146, pp. 123–130, 2019.

[29] M. El-Hindi, M. Heyden, C. Binnig, R. Ramamurthy, A. Arasu, and D. Kossmann, "BlockchainDB - Towards a Shared Database on Blockchains," in *Proceedings of the 2019 International Conference on Management of Data - SIGMOD '19*, vol. 12, no. 11. New York, New York, USA: ACM Press, 2019, pp. 1905–1908. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3299869.3320237

[30] S. Nathan, C. Govindarajan, A. Saraf, M. Sethi, and P. Jayachandran, "Blockchain meets database," *Proceedings of the VLDB Endowment*, vol. 12, no. 11, pp. 1539–1552, 7 2019. [Online]. Available: http://arxiv.org/abs/1903.01919http://dl.acm.org/citation.cfm?doid=3342263.3360362

[31] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, and A. V. Vasilakos, "Security and privacy for storage and computation in cloud computing," *Information Sciences*, vol. 258, pp. 371–386, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.ins.2013.04.028

[32] P. Christen, T. Ranbaduge, and R. Schnell, *Linking Sensitive Data*. Cham: Springer International Publishing, 2020.

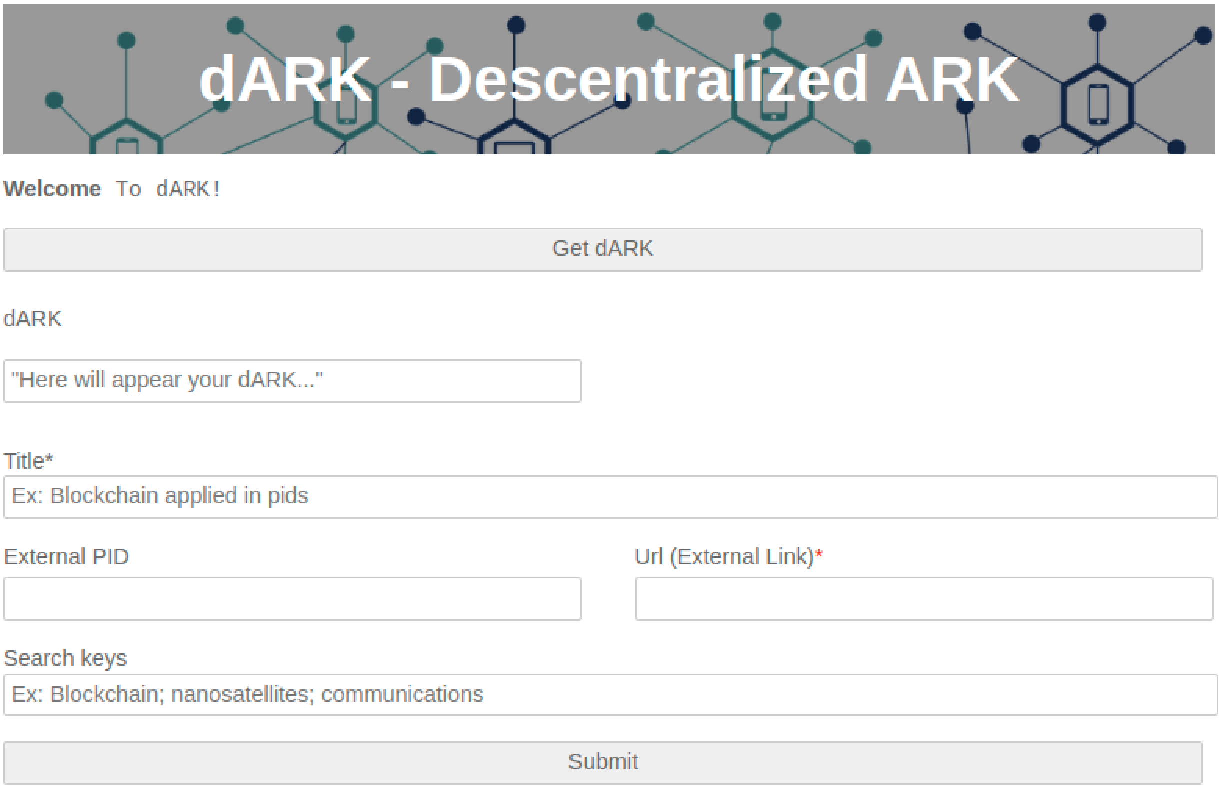
[33] H. Cousijn, R. Braukmann, M. Fenner, C. Ferguson, R. van Horik, R. Lammey, A. Meadows, and S. Lambert, "Connected research: The potential of the pid graph," *Patterns*, vol. 2, no. 1, p. 100180, 2021.

[34] J. Kunze, "Towards electronic persistence using ark identifiers," 2003.

[35] J. R. G. III, M. Jurić, D. Schlegel, F. Hoyle, M. Vogeley, M. Tegmark, N. Bahcall, and J. Brinkmann, "A map of the universe," *The Astrophysical Journal*, vol. 624, no. 2, p. 463, may 2005. [Online]. Available: https://dx.doi.org/10.1086/428890

[36] F. Bugiotti, L. Cabibbo, P. Atzeni, and R. Torlone, "Database design for NoSQL systems," in *International Conference on Conceptual Modeling*. Springer, 2014, pp. 223–231.

APPENDIX A



dARK

"Here will appear your dARK..."

Title*

Ex: Blockchain applied in pids

External PID

Url (External Link)*

Search keys

Ex: Blockchain; nanosatellites; communications

Fig. 12. Example of an assigning PID interface.

```
{

  "noid": "8003/fkwff300001v",

  "external_pids": [
    {
      "id": "0x6cd32058785840306ddaaf126d57999315739722ce57131dde41aa3690a8afdd",
      "schema:": "DOI",
      "value": "10.1016/J.IS.2021.101826",
      "owner:": "0xf17f52151EbEF6C7334FAD080c5704D77216b732"
    }
  ],

  "mime_type" : "JSON",

  "schema" : "json test",

  "record_type" : "Publication",

  "record_subtype" : "Journal Article"

  "payload": "{
                ’title’: ’Blockchain-based Privacy-Preserving Record Linkage: enhancing data privacy in an untrusted environment.’,
                ’author’: ’Thiago Nobrega’,
                ’type’: ’Article’
             }",

  "external_links": [
    "https://doi.org/10.1016/j.is.2021.101826"
  ],

  "search_terms" : ["Blockchain-based Privacy-Preserving Record Linkage: enhancing data privacy in an untrusted environment.", "Thiago Nobrega"]

  "owner": "0xf17f52151EbEF6C7334FAD080c5704D77216b732"

}
```

Fig. 13. dARK JSON response example.