ciTIzen-centric DAta pLatform (TIDAL): Sharing Distributed Personal Data in a Privacy-Preserving Manner for Health Research

Chang Sun [a,*], Marc Gallofré Ocaña [b], Johan van Soest [c,d] and Michel Dumontier [a]

[a] *Institute of Data Science, Faculty of Science and Engineering, Maastricht University, Maastricht, The Netherlands*
E-mails: chang.sun@maastrichtuniversity.nl, michel.dumontier@maastrichtuniversity.nl
[b] *Department of Information Science and Media Studies, University of Bergen, Bergen, Norway*
E-mail: marc.gallofre@uib.no
[c] *Department of Radiation Oncology (Maastro), GROW School for Oncology, Maastricht University Medical Centre+, Maastricht, The Netherlands*
E-mail: j.vansoest@maastrichtuniversity.nl
[d] *Brightlands Institute of Smart Society (BISS), Faculty of Science and Engineering, Maastricht University, Heerlen, The Netherlands*
E-mail: j.vansoest@maastrichtuniversity.nl

Abstract. Developing personal data sharing tools and standards in conformity with data protection regulations is essential to empower citizens to control and share their health data with authorized parties for any purpose they approve. This can be, among others, for primary use in healthcare, or secondary use for research to improve human health and well-being. Ensuring that citizens are able to make fine-grained decisions about how their personal health data can be used and shared will significantly encourage citizens to participate in more health-related research. In this paper, we propose ciTIzen-centric DatA pLatform (TIDAL) to give individuals ownership of their own data, and connect them with researchers to donate their personal data for research while being in control of the whole data life cycle including data access, storage and analysis. We recognize the most existing technologies focus on one particular aspect such as personal data storage, or suffer from executing data analysis over a large number of participants, or face challenges of low data quality and insufficient data interoperability. To address these challenges, the TIDAL platform integrates a set of components for requesting subsets of RDF (Resource Description Framework) data stored in personal data vaults based on SOcial LInked Data (SOLID) technology and analyzing them in a privacy-preserving manner. We demonstrate the feasibility and efficiency of the TIDAL platform by conducting a set of simulation experiments using three different pod providers (*Inrupt.net*. *Solidcommunity.net*, Self-hosted Server). On each pod provider, we evaluated the performance of TIDAL by querying and analyzing personal health data from an increasing number of participants and variables. The performance evaluation of TIDAL shows the execution time has a linear correlation between the number of pods on all pod providers. Platforms such as TIDAL can play an important role to connect citizens, researchers, and data organizations to increase the trust placed by citizens in the processing of their personal data.

Keywords: Personal Data Vault, Personal Data Sharing, Health Data Management, Data Privacy, Linked Data, Decentralized Network, Citizen Science

*Corresponding author. E-mail: chang.sun@maastrichtuniversity.nl.

1. Introduction

Giving individuals more control over who can access their personal data for what purpose and to make their data available using privacy-preserving and transparent methodologies will significantly encourage their engagement in healthcare research [1, 2]. To improve evidence-based healthcare research and empower healthcare authorities to optimize the accessibility and effectiveness of the healthcare services, we need sufficient personal health data [3]. However, personal health data is largely collected and managed by various healthcare service providers. In Europe, many citizens still have limited electronic access to their own health data, often scattered among healthcare service providers [4]. As a result, citizens have limited control over their own data, or need to control this data at various locations.

Since the General Data Protection Regulation (GDPR) and ePrivacy legislation have been released, European Union's citizens increasingly value their data rights and information privacy [5]. However, there is no mature technology and standards that enable individuals to fully exercise their data rights in a simple way. The public consultation on the European strategy for data showed that almost 88% of all respondents (806 contributors) would like to have more access and control over the data they generate [6]. A large proportion of respondents would be willing to share their data, especially for health-related research, but a majority of them considered that there are no sufficient tools and mechanisms to "donate" their data. For example, at present, if individuals are willing to donate their health data to help chronic disease research, they need to look for an ongoing research study that is recruiting new participants and has requirements that are applicable. Meanwhile, individuals need to trust and be willing to share their data with this research study. However, sharing personal data often raises concerns about privacy, security, ownership, and accountability. Examples of these concerns are: who will have access to the data and study results, how the individuals can change/revoke the permissions to (fully or partially) access the data, and whether the data is used for other purposes.

In this study, we address the research challenge of *how to engage individuals to "donate" their personal data for health-related research with maximal control in data access, storage, and analysis?* The current personal data management technologies are mostly research-driven and in their early stages. Given the gap in the existing personal data platforms, we propose a new citizen-centric data platform (called TIDAL) that gives individuals fine-grained access to their data and ensures citizen controlled data are processed in a predefined manner. We designed a prototype as proof-of-concept following an exploratory technology development process in light of our experience in the development of a privacy-preserving distributed data analysis infrastructure in the previous studies [7–10]. TIDAL consists of an integrated set of components for requesting subsets of data stored in personal data vaults using SOLID technologies [11] and analyzing them in a privacy-preserving manner. SOLID, standing for SOcial LInked Data, is a set of technologies that facilitates users to create decentralized applications using Linked Data and the W3C standards and protocols. We evaluated the performance of TIDAL by executing simulation experiments on various sizes of simulation datasets. The experiments proved the feasibility and efficiency of TIDAL using three different pod providers (*Inrupt.net*. *Solidcommunity.net*, Self-hosted Server). We believe the TIDAL platform will increase the trust placed by individuals and the transparency of the processing of their personal data.

We summarize the main contributions of this paper:

- proposing a new citizen-centric data platform (TIDAL) that facilitates individuals to store and access to their personal data using personal data vault technologies (such as SOLID) and provide direct consent to health-related research;
- applying Data Privacy Vocabulary [12, 13] to structure the personal data requests as digital consents in TIDAL to meet the requirements of GDPR;
- formulating data requests into RDF format with integrating vocabulary services and standards to improve the interoperability of personal data use;
- executing privacy-preserving data mining algorithms automatically using parameters and configurations promised in the data request and only the results are sent to the researchers; and
- evaluating the feasibility and efficiency of TIDAL in different experimental settings.

The article below is structured as follows: section 2 introduces the recent related work and remaining challenges. Section 3 describes the technologies we applied in the TIDAL platform. Section 4 presents the architecture of

TIDAL, and demonstrates how it works for researchers and participants. Section 5 describes the experimental setup and results of TIDAL in different user scenarios. Section 6 discusses our discovery and limitations of the current version of TIDAL. Finally, Section 7 outlines the conclusions and future work.

2. Related work

Researchers and companies have developed several tools with different emphases and features to enable individuals to be in control of their data. We identified the following ten projects and tools which have been applied in practice and provided a comparison table including detailed information and additional resources in the supplementary material[1]. DEecentralised Citizen Owned Data Ecosystems (DECODE) [14], MyHealthMyData (MHMD) [15] and OwnYourData [16] are based on distributed-ledger technologies such as Blockchain to provide traceable and transparent data-access control. MIDATA [17] and MedMij [18] are national programs in Switzerland and Netherlands that provide citizens with new data ecosystems to use their medical data for healthcare services and research. Digi.me [19] are CozyCloud [20] are commercial products providing mobile applications and cloud services to share personal data. The Hub of All Things (HAT) [21] – a foundation –, MyDex [22] – a community interest company –, and openPDS [23] – a research project – utilize Personal Data Store (PDS) [24] technologies to provide users with servers to store and share their personal data and execute on-device computations.

2.1. Existing personal data management tools

DECODE and MHMD were both funded by the European Union's Horizon 2020 research and innovation programme [25]. DECODE enables individuals to keep personal information private or share it for the public good using peer-to-peer networks and Blockchain technologies. This ecosystem, from an operating system to an interactive dashboard, has been developed and piloted in Barcelona and Amsterdam. However, it focuses on individual control over data sharing rather than data processing and analysis. Individuals can specify "smart rules" for their personal data to pre-define under what conditions the data can be used. To guarantee privacy-preserving transactions in the Blockchain, DECODE uses the Coconut selective disclosure credential system [26]. Since DECODE relies on its own operating system and tools, it lacks the interoperability and extensibility that would be required for data mobility across healthcare systems and national borders.

MHMD [15] is another Blockchain-based solution that connects organizations and individuals to make anonymised data available for open research. It enables individuals to provide dynamic consent for different types of potential data usage and monitor the usage. Similar to the DECODE "smart rules", the MHMD consent determines under what conditions the data can be used. MHMD supports data analysis algorithms combined with secure multiparty computation and asymmetric encryption for preserving privacy. However, individuals' data is still hosted at organizations (e.g., hospitals), which are the only ones empowered to give permission to researchers requesting data. OwnYourData [16], developed by a non-profit organization, is another personal data management product that uses Blockchain technology to make data immutable. OwnYourData stores users' personal data and provides the insights from it.

MIDATA [17], a nonprofit cooperative in Switzerland, operates a data platform that enables Swiss citizens to selectively share their data with medical research and clinical studies. MIDATA shares the same limitations as DECODE on the interoperability and extensibility of their data ecosystem. MedMij [18] is established as a standard in the Netherlands for the secure exchange of health data between Dutch residents and healthcare providers. MedMij, serving as a high-level guideline, proposes a set of information standards to structure the health data from different sources and standardize the data exchange. However, MedMij does not yet include researchers in the network nor facilitates citizens to voluntarily share their health data for research studies or any other purposes.

Digi.me [19] and CozyCloud [20] deliver commercial products to give people control of their data when using web or mobile applications, but both host data centralized in their own cloud servers. Similarly, MyDex [22] and HAT [21] offer PDS as cloud hosted servers to store personal data and connect it with other web or moblie appli-

[1] https://doi.org/10.6084/m9.figshare.19111508

cations and services. Different from the previous tools that host the PDS in their own servers, openPDS [23] allows users to self-host the PDS and use it as a service. OpenPDS also applies the SafeAnswers framework which executes the queries inside the PDS rather than sending anonymized data and returns and aggregates results from more than one PDS. It allows users to manage data access and monitor data usage. However, SafeAnswers presents a computational challenge for complex data analysis and does not consider the scenario of conducting research studies in large populations.

2.2. Remaining challenges

The existing solutions often focus on one particular aspect such as personal data storage and overview, data access control, and data sharing with healthcare services or with mobile applications. To the best of our knowledge, there is no platform that enables individuals to connect with researchers to donate their personal data for research while being in control of the whole data life cycle including data access, storage and analysis. Only a few tools support personal data analysis over a number of participants. These tools face challenges such as the data permissions are specified by the data organizations rather than individuals and the analysis algorithms are relatively simple. We also see an urgent need for more investments in data quality and interoperability to improve the feasibility and sustainability of personal data management platforms [2]. Therefore, we propose TIDAL to fill the gaps that we have identified from the existing works.

3. Background

3.1. SOLID - Decentralized data management

SOLID (SOcial LInked Data) is a decentralized data management platform based on W3C standards, Resource Description Framework (RDF), and Semantic Web technologies, initiated by Tim Berners-Lee [27–29]. Rather than the tech giants storing and controlling personal data from their users, SOLID technologies enable users to store and manage their data independently from the applications so that users can retain sovereignty over their data. SOLID is composed of three core components - the data pod (i.e., where the data is stored), the application (i.e., the services that users can use and grant access to), and providers (i.e., where the pod and application are technically hosted).

Each SOLID user is assigned with a WebID[2] as a unique global ID for identification and authentication. SOLID data pods are web-based storage services and databases where various types of data can be stored such as RDF triples, free text, images, videos, or even webpages. However, SOLID is featured by its capability to parse and serialize structured data using RDF in syntaxes like Turtle and JSON-LD. Data in SOLID pods can be accessed and managed by a decentralized authentication[3] and Web Access Control (WAC)[4] mechanism [27] which is a decentralized cross domain access control system. WAC in SOLID provides the pod owners with a fine-grained access control for every single data element in their data pod by granting other SOLID users and applications the permissions to read, modify, and write the stored data elements. The Access Control List ontology[5] [30] is utilized in SOLID to describe the different operations over the target data elements in the pods.

SOLID applications are developed on top of the aforementioned technology stack. Most applications are developed for web or mobile platforms. Users are able to grant and revoke permissions to both SOLID applications and other users at any time. SOLID allows multiple applications to access and reuse the same data from a pod, thereby potentially minimizing data duplication and staleness. SOLID pods can be hosted on public servers by pod providers which play a similar role as the cloud storage providers. SOLID pods can also be self-hosted on personal servers, and migrated from pod providers to self-hosted. A single SOLID user can own more than one data pod which is hosted by one or multiple pod providers. Users are able to select and change their pod providers at any time based

[2] https://www.w3.org/wiki/WebID
[3] https://solid.github.io/authentication-panel/solid-oidc
[4] https://solid.github.io/web-access-control-spec
[5] https://www.w3.org/ns/auth/acl

on providers' geographical locations, responsibilities, different degrees of privacy protection and legislation. Thus, SOLID presents a distributed scenario that challenges the communication between SOLID applications and data pods, but provides fine-grained data control to users.

3.2. Personal Health Train - Distributed data analysis initiative

The Personal Health Train (PHT) initiative was designed for healthcare innovators and researchers to access heterogeneous data sources and learn from the distributed data in a privacy-preserving manner [31], [10]. The essence of this approach is to transfer the research questions and analysis algorithms (from researchers) to data rather than centralizing data. Only the analysis results are sent back to the researchers.

The PHT technology has been developed and implemented in several real-life use cases in the healthcare domain. In our previous studies, we have developed the PHT infrastructure to address horizontally and vertically partitioned data[6] problems [9], [8], [32], [33]. In this study, we further extend the PHT infrastructure from the level of information control by organizations to information control by individuals themselves.

4. OVERVIEW AND IMPLEMENTATION OF TIDAL

The primary use case of TIDAL is for researchers (data requesters) who want to analyze personal data and participants (data subjects) who are willing to donate their data for research. In this section, we will present the overview and implementation of TIDAL by describing a use case between these two types of users - the participants and researchers. They both need SOLID accounts and data pods so that they can be authenticated on TIDAL. TIDAL facilitates them to create new data elements or files, modify or delete existing RDF data elements, and query data elements from their own pods. An example of fetching RDF data from a solid pod is shown in Figure 1.

TIDAL authenticates and interacts with SOLID pods with a Javascript package - solid-node-client (V2.0.2) [34]. Solid-node-client enables pod owners to access their pods, create or modify data in their pods, and grant or revoke the permissions via a web application. To store, parse, and query RDF data from SOLID pods, TIDAL uses the rdflib.js (V2.1.6) [35] and tripledoc (V4.4.0) [36] library. Similar libraries such as solid/query-ldflex can also be used to access data in Solid pods through LDflex expressions [37].

4.1. Researcher posts participation request

In the first phase, a participation request is crafted by the researcher. The content of the request is only stored at the researcher's SOLID pod, while the Uniform Resource Identifier (URI) of the request is stored in an index file on TIDAL. Subsequently, TIDAL reads the index file to find all the existing requests and presents them to the participants for approval (Figure 2).

To post a participation request, the researcher is required to register as a "researcher role" by providing basic information such as job position, affiliation, and research topics. The researcher is issued a public-private key pair that will be used to verify the identity of the researcher and the integrity of the request. When the researcher publishes a request, the URI and content of the request will be automatically signed by the researcher's private key. Any changes to the request will

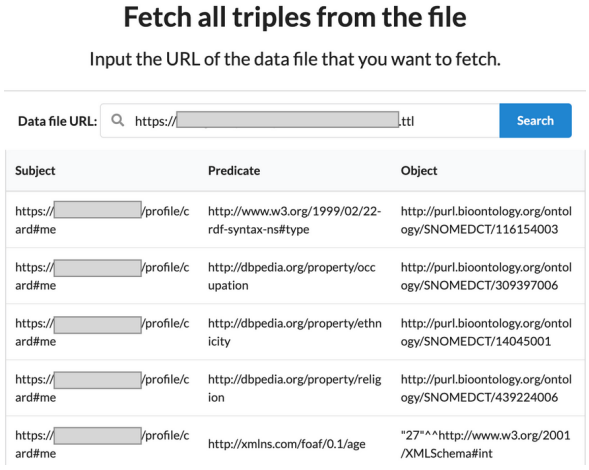

Fig. 1.: An example of query personal data in a RDF format from a SOLID pod using TIDAL.

[6]Horizontally partitioned data is that particular data from different individuals are distributed over multiple data sources, while vertically partitioned data represents different data about a particular individual are distributed over multiple sources.

cause a verification failure when the request is executed to retrieve participants' data. TIDAL uses the Ed25519 algorithm [38, 39], a high-speed and high-security signature scheme, for public-key signature encryption. Ed25519 is an implementation of the Elliptic Curve Digital Signature Algorithm (EdDSA) using SHA-512 (SHA-2) and Curve25519 with Twisted Edwards Curve [40]. It has been widely used in protocols such as TLS 1.3 and SSH [41]. TIDAL uses Ed25519 from the TweetNaCl (V1.0.3) package [42], a port of Networking and Cryptography library [43] to Javascript.

The researcher creates and publishes a participation request by completing a participation request form (Figure 4). The request form was designed as a digital consent to be informed and specific. Complying with the GDPR requirements on consents, the request form describes the identifiers of the requester (researcher) and controller (trusted party – a certificated organization compliant with GDPR that executes the requests and analyses), what data elements in which personal data categories will be processed in what time frame, how the requested data will be processed for what purposes, and the possible risks and consequences of data processing such as for participants in relation to automated decision making. The request is represented using Schema.org vocabulary (*https://schema.org/*) and the Data Privacy Vocabulary (DPV, *http://www.w3.org/ns/dpv*). The DPV specifically captures the nature of data processing in relation to EU General Data Protection Regulation. The overall schema of an example participation request is illustrated in Figure 3 and the stored RDF format of the example instance is shown in Listing 1.

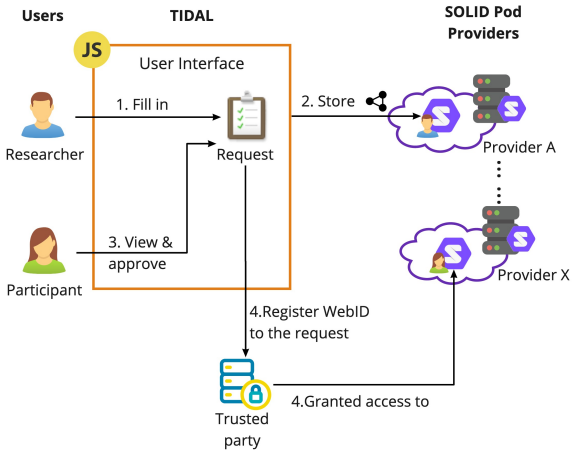

Fig. 2.: Interaction between researchers and participants on TIDAL. The researcher fills the request form and store it in their SOLID pod. The participant views and approves the published request on TIDALThe participation record is stored in the participant's pod and the trusted party.

The request form includes fields to specify the following requested field (RF):

RF 1: the purpose of the research where researchers clearly indicate the purpose of processing personal data in their research. Researchers can select one or more from a list of data processing purposes described in DPV such as `dpv:Security`, `dpv:ResearchAndDevelopment`. These elements will be described and stored as, for example, *http://www.w3.org/ns/dpv#Security* and *http://www.w3.org/ns/dpv#ResearchAndDevelopment* in the request form in the researcher's SOLID pod.

RF 2: description of the specific purpose where researchers elaborate the purpose with more details in human readable text. Researchers can fill in answers in free text such as *"Learn association between the status of Type 2 diabetes and patients' dietary patterns using linear regression"*.

RF 3: the category of requested data elements where researchers indicate which personal data category best describes the requested data elements. Researchers can select one or more from a list of personal data categories described in DPV such as `dpv:Health` *(http://www.w3.org/ns/dpv#Health)*, or `dpv:Income` *(http://www.w3.org/ns/dpv#Income)* and stored them in the researcher's SOLID pod.

RF 4: the data elements where researchers indicate the data elements (URI) are requested from the participants. Researchers can fill in one or more URIs of the requested data elements. Researchers can also search for the existing URIs from the existing ontologies and select the ones for the requested data elements. For example, instead of requesting the "*Age*" in plain text, researchers can set the URI of Age in SNOMED CT *http://purl.bioontology.org/ontology/SNOMEDCT/397669002* as requested data element in the form.

RF 5: the expiration date of consent where researchers specify an exact date when the consent will be no longer valid. Researchers can only give future dates as answer in this field such as *2025-01-23*.

RF 6: the number of individuals who agree to participate in the study where researchers specify a minimal number of participants required to initiate the data processing. Researchers can only give integer numbers as the answer in this field such as *1000*.)

RF 7: the categories of data processing where researchers indicate which category or a chain of data processing will be performed on the requested data. Researchers select one or more from a list of data processing categories described in DPV such as `dpv:Copy` *(http://www.w3.org/ns/dpv#Copy)*, `dpv:Anonymise` *(http://www.w3.org/ns/dpv#Anonymise)* and `dpv:Analyse` *(http://www.w3.org/ns/dpv#Analyse)*.

RF 8: the methods or algorithms in data processing where researchers specify how the requested data will be processed. Researchers select one or more from a list of predefined algorithms such as *Linear regression* and *logistic regression*.

RF 9: the consequences and impact where researchers communicate the possible risks and consequences of data processing to the participants such as the impact of automated decision making. Researchers answer in text that is human-readable and understandable for the general public.

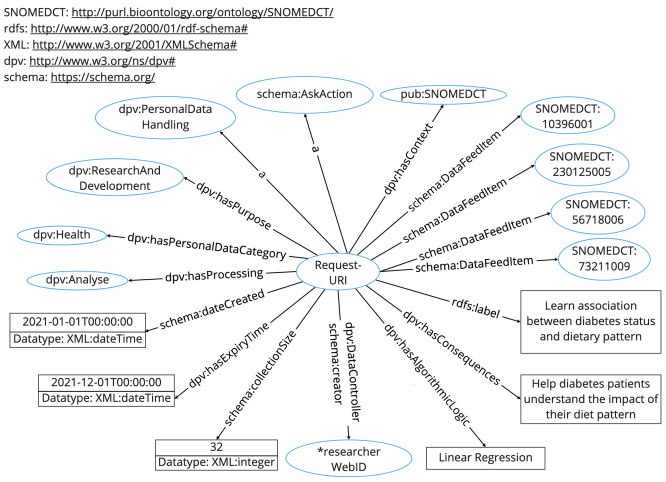

Fig. 3.: Schema of an example of a participation request.

To improve the interoperability of the requested data elements, we have integrated BioPortal API [44] in TIDAL to help researchers use standardized ontologies and terminologies for specific information elements. Bioportal is the most comprehensive ontology repository for biomedical ontologies including more than 800 ontologies. TIDAL supports researchers to search the existing biomedical ontologies and terminologies provided by Bioportal and apply them to the requested data elements. For example, instead of using "*diagnosis*" as a requested data element, researchers can look for the terms from well-established ontologies such as "*http://purl.obolibrary.org/obo/NCIT_C152*" or "*http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C15220*", by searching the keyword "*diagnosis*" in Data Elements (URI) in the request form.

Each request form is assigned with a URI when it gets published. All the information in the form is structured in RDF format as a `schema:AskAction` and `dpv:PersonalDataHandling` and stored in the researchers' SOLID pods (Step 2 in Figure 2). The request is signed with the researcher's private encryption key while it is published in order to prevent any subsequent changes. The URI and the signature of the request are stored on TIDAL, while the content of the request is only stored in the researcher's SOLID pod.

```
@prefix : <http://exampleresearcher.solidprovider.com/public/request.ttl#>.
@prefix schema: <https://schema.org/>.
@prefix exre: <http://exampleresearcher.solidprovider.com/profile/card#>.
@prefix XML: <http://www.w3.org/2001/XMLSchema#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix dpv: <http://www.w3.org/ns/dpv#>.
@prefix SNOMEDCT: <http://purl.bioontology.org/ontology/SNOMEDCT/>.

:161964062096710764675982245664
   a schema:AskAction, dpv:PersonalDataHandling;
   rdfs:label "Learn association between diabetes status and dietary pattern";
   schema:collectionSize 32;
   schema:creator exre:me;
   schema:DataFeedItem SNOMEDCT:10396001, SNOMEDCT:230125005, SNOMEDCT:56718006, SNOMEDCT:73211009;
   schema:dateCreated "2021-01-18T00:00:00Z"^^XML:dateTime;
   dpv:hasAlgorithmicLogic "Linear Regression";
   dpv:hasConsequences "Help diabetes patients understand the impact of their diet pattern";
   dpv:hasContext SNOMEDCT;
   dpv:hasDataController exre:me;
   dpv:hasExpiryTime "2021-12-31T00:00:00Z"^^XML:dateTime;
   dpv:hasPersonalDataCategory dpv:Health;
   dpv:hasProcessing dpv:Analyse;
   dpv:hasPurpose dpv:ResearchAndDevelopment.
```

Listing 1: An example of generated RDF triples from the request form stored in researcher's SOLID pod

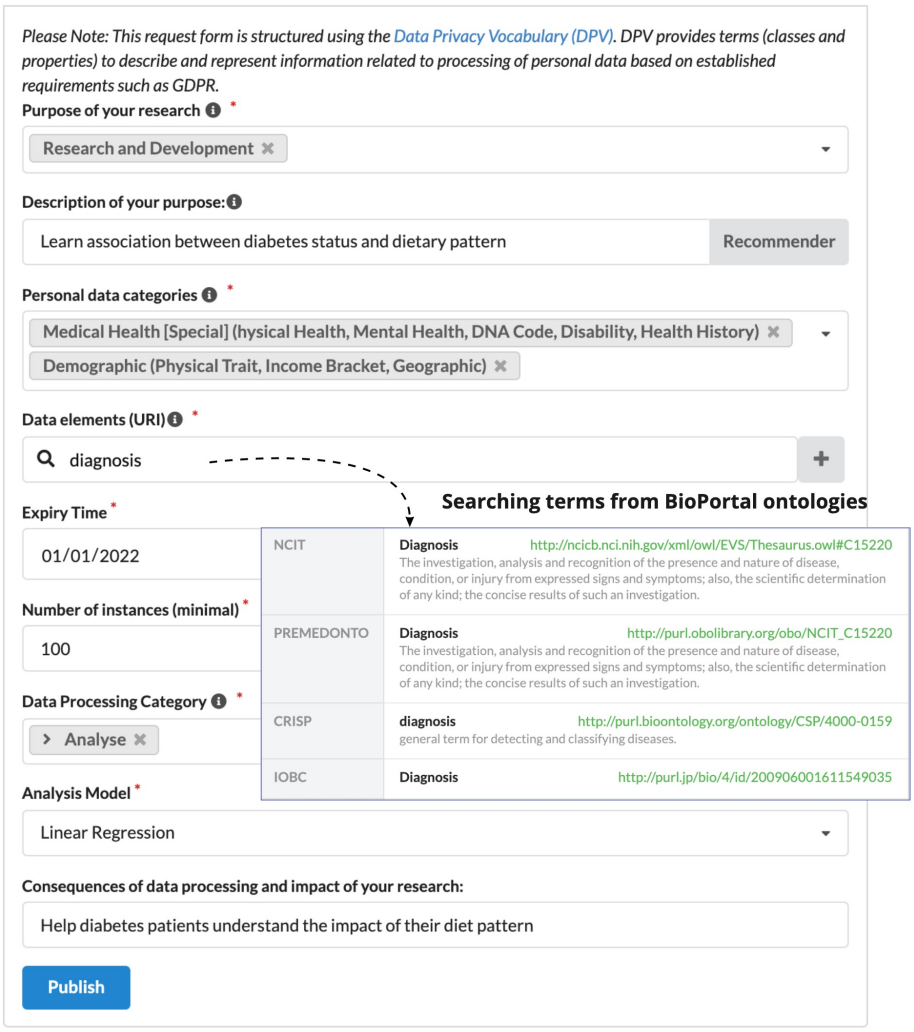

Fig. 4. Participation request form on TIDAL.

4.2. *Participant views and approves requests*

All published requests that are in the valid period (i.e., before the expiration date of the request) are visible on TIDAL. TIDAL queries RDF data from the request files and displays them in a human readable manner in a card view. We assume the participants have their personal health data (e.g., medical records, medications, lifestyle and behavior data) structured and stored using RDF in their own SOLID pods. Each card is linked to the original request file from the researcher's SOLID pod. Figure 5 shows an example of the published participation requests view on TIDAL from a participant's perspective. The research purpose, personal data category, data processing category, and data elements are linked with the valid URIs of the terms.

If TIDAL detects the requested data elements in the participant's pod, the participant can voluntarily join the data request by setting up a preferable withdrawal time (earlier than the request expiry date) and selecting the party they trust to process their personal data. TIDAL generates an instance adhering to the `schema:JoinAction` and `dpv:Consent` in RDF format describing which request (URI) has been approved at what time and until when this approval is valid. The statement is structured by using DPV and stored in a private folder in the participant's SOLID pod. Figure 6 and Listing 2 shows the schema of an example of participation and generated consent statements.

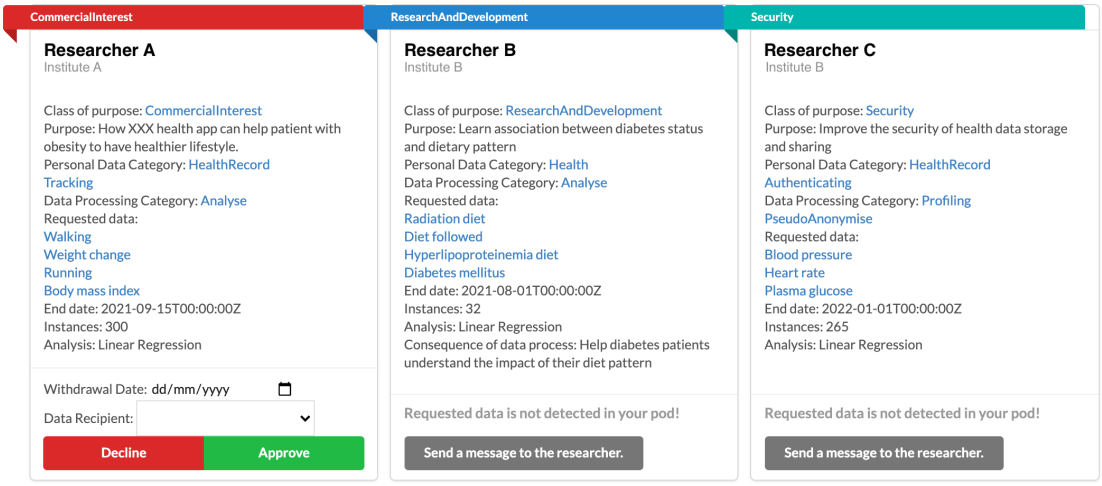

Fig. 5. An example of viewing published participation requests on TIDAL.

By approving the request, the participant gives the trusted (or authenticated) party access to the requested data elements in the pod. The participant's WebID will be registered at the trusted party under the analysis request URI (Step 4 in Figure 2). Meanwhile, TIDAL generates logging information in participant's pod including at what time the access have been granted, to whom (WebID), to what data elements, for what data request (request ID), and the valid period of the permission. The logging is readable by the participants but not editable by anyone. Until now, data elements have not been accessed and retrieved by any parties.

If TIDAL fails to detect the requested data elements in the pod, the participant is not able to join the research. It is possible that the participant does not have the requested data or the researcher and participant use different standards or ontologies to describe the same data element. In this case, the participant can send messages to the researcher anonymously on TIDAL to report this issue.

4.3. Data retrieval and analysis execution

To process the request, the following conditions need to be satisfied: (1) the request being in the inclusion period, and (2) the number of participants exceeding the minimum number set in the request. When the request meets both conditions, the researcher can communicate with the trusted party on TIDAL to trigger the data retrieval and analysis. The trusted party hosts the data analysis component including verifying the request, querying data from participants' pods, and executing the predefined analysis algorithms. The data analysis component was built using Javascript and Docker Containers. Docker Container has similar resource isolation and allocation benefits to virtual machines, creating temporary and secure sandboxes. We used the node-docker-api package (V1.1.22) [45] in a combination of solid-node-client and rdflib.js libraries to access SOLID pods from a Docker container.

Figure 7 shows the workflow of data retrieval and analysis after the researcher triggers the execution of a request. TIDAL will first generate and send a

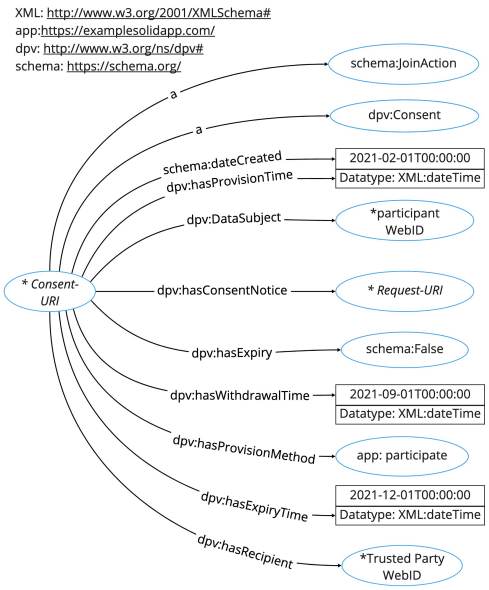

Fig. 6.: Schema of an example of participation statement.

```
@prefix : <http://exampleParticipant.solidProvider.com/private/participation#>.
@prefix schema: <https://schema.org/>.
@prefix dpv: <http://www.w3.org/ns/dpv#>.
@prefix XML: <http://www.w3.org/2001/XMLSchema#>.
@prefix part: <https://exampleparticipant.solidprovider.com/profile/card#>.
@prefix req: <https://exampleresearcher.solidprovider.com/public/request.ttl#>.
@prefix extp: <http://examplertrustedparty.solidprovider.net/profile/card#>
@prefix app: <https://examplesolidapp.com/

:161970412662952996575421551 98
   a schema:JoinAction, dpv:Consent;
   schema:dateCreated "2021-02-18T00:00:00Z"^^XML:dateTime;
   dpv:DataSubject part:me;
   dpv:hasConsentNotice req:161964062096710764675982245664;
   dpv:hasExpiry schema:false;
   dpv:hasExpiryTime "2021-12-31T00:00:00Z"^^XML:dateTime;
   dpv:hasProvisionMethod app:participate;
   dpv:hasProvisionTime "2021-02-18T00:00:00Z"^^XML:dateTime;
   dpv:hasWithdrawalTime "2021-09-18T00:00:00Z"^^XML:dateTime;
   dpv:hasRecipient extp:me.
```

Listing 2: An example of generated participation statements in a RDF format in the participant's SOLID pod.

```
@prefix : <http://examplertrustedparty.solidprovider.net/inbox/triggermessage#>.
@prefix : <http://trustedparty.solidprovider.com/profile/card#>.
@prefix schema: <https://schema.org/>.
@prefix req: <http://exampleresearcher.solidprovider.com/public/request.ttl#>.
@prefix exre: <http://exampleresearcher.solidprovider.com/profile/card#>.
@prefix XML: <http://www.w3.org/2001/XMLSchema#>.

:160622932739325095672093710975
   schema:actionStatus schema:ActivateAction;
   schema:creator exre:me;
   schema:dateCreated "2021-04-20T09:37:57.499Z"^^XML:dateTime;
   schema:target req:161964062096710764675982245664.
```

Listing 3: An example of generated trigger message (Activate Action) sent by the researcher.

`schema:ActivateAction` message (Listing 3) to the trusted party. The request file is retrieved from the researcher's pod, parsed, and verified using the public key. The data must specify the docker image identifiers (`dpv:hasAlgorithmicLogic`), requested data elements (`schema:DataFeedItem`), valid period of the request (`dpv:hasExpiryTime`) and other input parameters for the trusted party to retrieve the Docker image from the central repository and execute the analysis. TIDAL can manage multiple data retrieval and analysis request from researchers simultaneously.

If the integrity of the request is verified, the trusted party fetches the requested data elements from each participant's pod (adhering to participation constraints such as participation time period) without storing their identifiers (i.e., WebIDs). This fetching process includes querying full RDF files from participants' pods, parsing them using the rdflib.js library, and extracting the requested data elements. When any data are being retrieved from the participants, TIDAL writes logging records in participants pods. These logging records identify what data elements are extracted, by whom (WebID), at what time, for which data request (request ID), and whether the analysis is executed. The queried data is then fed into the data analysis model which is pre-defined in the Docker image. Finally, the results of the analysis will be generated automatically and sent back to the researcher's SOLID pod. All received and created information at the trusted party such as queried data and intermediate results are destroyed.

5. EXPERIMENTS AND RESULTS

At the moment of implementing TIDAL (December 2020), there were two public SOLID pod providers: *Inrupt.net* and *Solidcommunity.net*. We tested the feasibility and efficiency of TIDAL using these two public pods providers and one self-hosted server. Each pod provider hosts 256 SOLID pods, corresponding to 256 participants. Each participant has a data file containing 128 generated variables and values structured by SNOMED CT [46] vocabularies in RDF/turtle format in their SOLID pods. A simplified data example is presented in Listing 4.

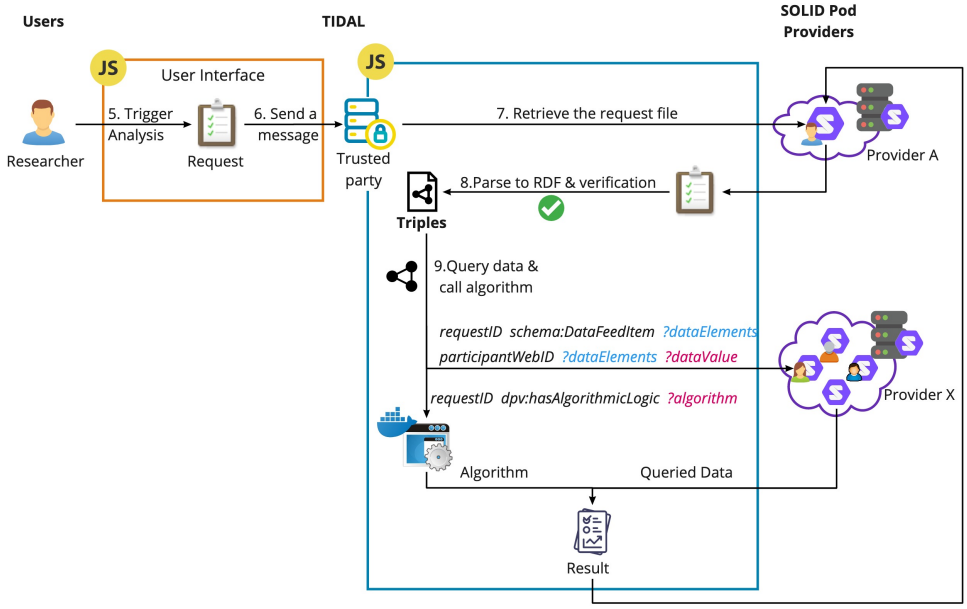

Fig. 7. Workflow of data retrieval and analysis triggered by the researcher.

Using each pod provider, we conducted a set of experiments using an increasing number of participants and variables. We started with requesting 4 variables from 4 to 256 participants, and ended with requesting 128 variables from 4 to 256 participants. The experiments focused on the steps after the researcher collects enough responses from the participants and triggers the analysis. The execution time has been measured from:

1. querying data request URI;
2. querying signature and verification key of data request;
3. verifying the signature to ensure the data request has not been modified;
4. (if the verification succeeds) querying the content of data request and WebIDs of participants; and
5. querying RDF data from all participants' pods.

The web interface of TIDAL was developed using the Semantic User Interface Framework (V2.4.2) [47] with responsive and scalable layout. We tested the web interface in the recent versions of Safari, Chrome, and Firefox. Data retrieval and analysis is performed on a 2.3 GHz PC using Dual-Core Intel Core i5 with 16GB RAM and 500GB hard disk running MacOS 10.15.7. To run the simulation experiment, we created 256 SOLID pods, generated and stored simulation data in each pod, and granted permission to the requests in an automatic way. The scripts are published at: https://sunchang0124.github.io/.

```
@prefix : <https://exampleparticipant.solidprovider.com/profile/card#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix SNOMEDCT: <http://purl.bioontology.org/ontology/SNOMEDCT/>.
@prefix schema: <https://schema.org/>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

:me a SNOMEDCT:116154003; # Patient
  SNOMEDCT:397669002 "27"^^xsd:int; # Age
  SNOMEDCT:50373000 "165"^^xsd:int; # Height
  SNOMEDCT:726527001 "55"^^xsd:int; # Weight
  SNOMEDCT:263495000 SNOMEDCT:248152002; # Gender, Female
  SNOMEDCT:271649006 "110"; # Systolic blood pressure
  SNOMEDCT:271650006 "90"; # Diastolic blood pressure
  SNOMEDCT:405751000 SNOMEDCT:44054006. # Type 2 diabetes
```

Listing 4: An example of the RDF data file in a participant's SOLID pod.

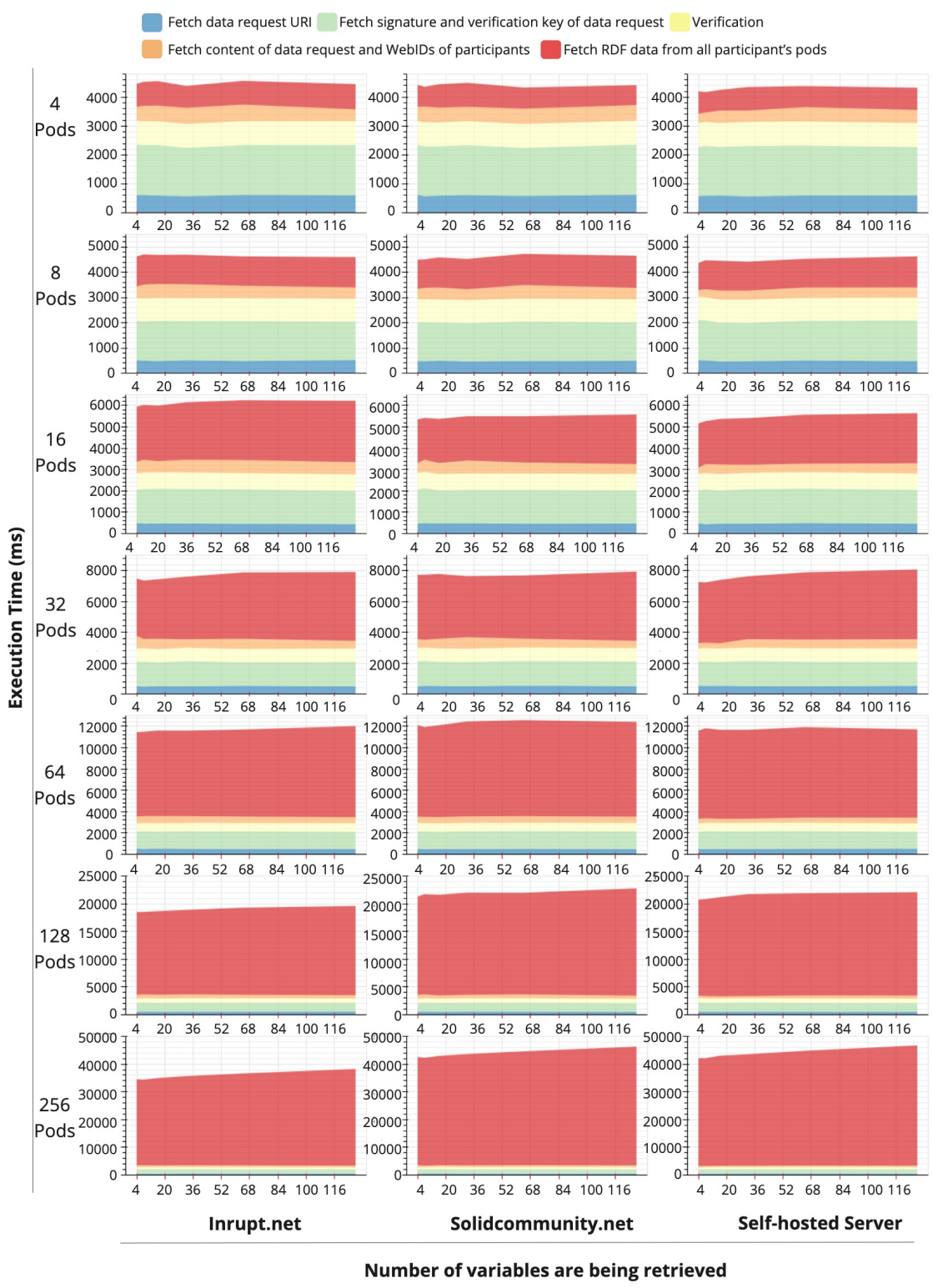

Fig. 8. Time costs in each execution steps in quering and analyzing data from SOLID pods with increasing the number of variables and pods hosted by *Inrupt.net*, *Solidcommunity.net*, and self-hosted server respectively.

Figure 8 shows how TIDAL scales for querying and analyzing data from individual pods as we increase the number of variables from 4 to 128 and the number of pods from 4 to 256 hosted by *Inrupt.net*, *Solidcommunity.net*, and the self-hosted server. The pod providers limit the number of requests that can be responded to at one time by the servers. Considering the scalability, we enable TIDAL to access participants' pods in a concurrent way using HTTP requests. TIDAL only queries the required data elements from SOLID pods of 64 participants simultaneously. Once a task gets finished, a new task is scheduled in the execution queue. We ran each experiment 10 times and presented the average time of the 10 experiments to avoid possible network latency fluctuations.

Figure 8 shows that the total time costs in querying 4 and 8 pods is approximately 4 to 5 seconds with a negligible increase as the number of variables increases. When we query data from a large number of pods, the time costs in fetching data from participants' pods becomes substantial. It rises linearly when we increase the number of pods using all pod providers. In the case of querying data from 256 pods, a gradual rise in time costs is observed as the number of variables increases. In all experiments, the time costs of the first 4 execution steps are constant and independent of how many variables and pods are required because they query information from a fixed number of pods from researchers or trusted parties.

Figure 9 shows the total time cost when querying the number of variables from 4 to 128 and the number of pods from 4 to 256 on three pod providers. From the experiments on all pod providers, the total time cost linearly scales when the number of pods is increased. The more variables are queried from each pod, the steeper the increase in time cost is presented. By contrast, the *Inrupt.net* server has a more stable rate and the least time consumption than the other two pod providers when querying data from more than 64 pods.

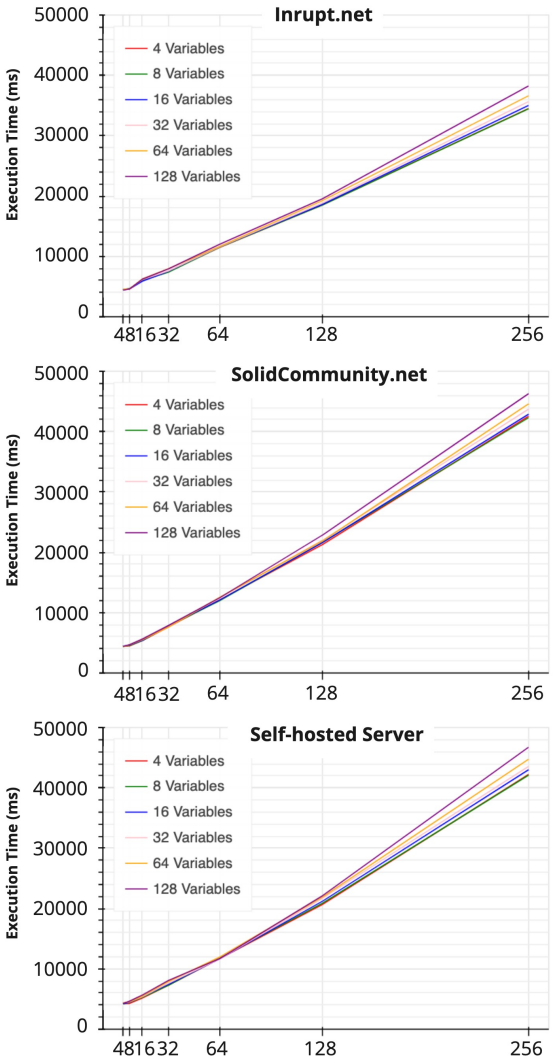

Fig. 9.: Total time costs in querying the increasing number of variables and pods on three pod providers Inrupt.net, Solidcommunitynet, and self-hosted server respectively.

6. DISCUSSION

We have demonstrated and tested a ciTIzen-centric DatA pLatform (TIDAL) using an increasing number of requested data elements retrieved from an increasing number of SOLID pods. From the performance evaluation of TIDAL, the execution time shows a linear correlation between the number of pods and the number of variables. The process expends the most of the time in querying data from all the participants. However, it only requires an average of 40 seconds to query 128 variables from 256 participants' SOLID pods. For a limited set of participants, this can be considered as an acceptable time for a batch process for use cases which do not demand instant results. In the future, we will improve the workflow and reduce the processing time.

When querying data from enormous pods, the number of variables being queried influences the total querying time (Figure 9). A possible solution to improve the query performance would be the provision of SPARQL support on SOLID pods which is missing in the current SOLID specification. A SPARQL endpoint would facilitate the execution of complex queries on pods instead of retrieving full RDF files and post-processing them on the client side to extract the requested data elements, decreasing applications performance. The increase in time of the "Fetch RDF data from all participants pods" (Figure 8) and the execution (Figure 9) is also influenced by the limited number of simultaneous requests being handled by the SOLID server. Additionally, the processing capability of the experimental hardware also created a bottle-neck on the querying and analyzing data processes. Therefore, for a practical application we advise the allocation of sufficient computational resources at key architectural locations to reduce the potential bottle-neck when querying and analyzing data from a large number of participants' pods.

TIDAL supports users to store and request personal data in a structured RDF format using well-established ontologies and terminologies by integrating the Biopartal API. These structured data are human and machine readable, provide language neutrality, unambiguous definitions, and clear relationships. It will contribute to enrich and improve the quality of personal data in the SOLID pods by linking data from multiple data sources. TIDAL works on a decentralized network where people can choose to store their data in different and multiple pods, or self-host the pod individually. Self-hosted data pods in our platform do not differ from other scenarios in terms of granting access, managing data (creating, modifying, deleting), donating their data to some research studies. Furthermore, to align with the data protection laws such as EU General Data Protection Regulation (GDPR), TIDAL applied the Data Privacy Vocabulary in the participation request to describe and represent information related to requesting and processing of personal data. Data protection laws grant data subjects (participants) rights to withdraw or modify their data anytime they want. On TIDAL, these rights are respected. After the participants approve the data request, they can still update the data elements or withdraw the approval decision anytime. The analysis can be done on the updated value of the data element or without the data elements which have been withdrawn. This process can also enhance reproducibility in research, as researchers can expand and scale their research, both in participants as future long-term effects/follow-up studies.

Furthermore, participants' data are queried and analyzed only at the trusted party. The trusted party can be a separate, independent entity in comparison to the researcher, SOLID provider and/or participant. Researchers can only formulate the request, define the parameters of the algorithm, and receive the final analysis results but never have access to the data. However, if the SOLID provider hosts the trusted party, this trusted party can become a node in a Personal Health Train (PHT) or Federated Learning (FL) infrastructure. In such an infrastructure, the research question travels to the data rather than data being transported to the research question. PHT or FL methodologies connect multiple distributed data sources (e.g., hospitals, clinics) and enable researchers to send analysis models to each data source (e.g. SOLID providers) and get the final learning results. However, in addition to strengthening the binary connection between data sources and researchers, TIDAL emphasizes on engaging individuals in health research and connecting them with both researchers and data sources. This is currently still missing in most PHT/FL implementations.

However, our development has to be seen in light of some potential limitations. First, we assume participants have their personal data structured and stored in their own SOLID pods. In practice, people who do not have enough knowledge about the data or the technologies will face challenges to structure and store their data correctly. To tackle this limitation, one solution can be encouraging data collectors such as hospitals or pharmacies to help participants structure their own data. For example, if the patients' medical records have been structured and linked with some international terminologies by the hospital, then the hospital can request to store the structured medical records data to patients' SOLID pods directly.

Furthermore, the current version of TIDAL presents every published data request that is in the valid period to all participants. In this case, participants receive some data requests that are not relevant to them. As researchers do not know which participants have the relevant data for their research, they are not able to send the data request to the target cohort instead of the general public. Therefore, to improve TIDAL, we are investing in generating privacy-preserving metadata of each SOLID pod. The privacy-preserving metadata is supposed to describe sufficient information about one pod but not reveal any sensitive information. One of the potential solutions is to employ Bloom Filter which is a probabilistic data structure for efficient set membership querying [48]. Bloom filter tests whether the participant has the requested data elements in their data pods and return two possible answers - "probably in the

pod" or "definitely not in the pod". With this method, we can prevent the participants in a specific study (e.g. for psychological disorders) from being identified that they are diagnosed with a specific disease or disorder. Another approach is that TIDAL asks participants to indicate their preference on the type of the research and data request. For example, if the participant is only interested in diabetes research, then TIDAL will only present data requests that are related to diabetes research in order to decrease the complexity of using TIDAL for general users.

7. CONCLUSION

In this paper, we presented a novel citizen-centric data platform (called TIDAL) to give individuals fine-grained access to their data and facilitate health research. TIDAL platform does not only collects data and manages digital consents, but also structures data requests with integrating the vocabulary services and standards such as Data Privacy Vocabulary. The data requests are used as a digital consent and provide the algorithm parameters and model configuration for the pre-defined data analyses. The analyses are executed in an automatic manner which ensures the data to be exactly analyzed as promised by the researchers in their data requests. Finally, only the analyses results are sent to the researchers. We demonstrated the feasibility and efficiency of TIDAL by running a set of simulation experiments using different numbers of variables and SOLID pods hosted on three different providers (*Inrupt.net*, *Solidcommunity.net* and a self-hosted server). TIDAL is not only limited to health research, it can be used in other fields such as social sciences (e.g., demographic and anthropology studies), economics and finance studies, political, marketing and education research.

To improve the user experience, we intend to recruit a group of users to assess the human interaction of TIDAL and collect their feedback. In the future, we will evaluate TIDAL in a real-life use case with real participants and health researchers. We will evaluate how usable the request form is for researchers, and how long it will take researchers to complete the entire request form. Meanwhile, we will also investigate how understandable the data request cards are for general participants, and how easy they feel to approve and withdraw the permissions.

The current version of TIDAL allows researchers to only perform a predefined set of analysis models. More complex analysis models will be designed in future work to facilitate researchers to perform experiments according to their scientific questions. Researchers can apply the needed model and tune the parameters instead of coding or modifying the entire model. The risk of hacking or data leak in the analysis process can be minimized. Another future work can be considered is to improve the logging process. The logging files in the current version of TIDAL stores the data access records in participants SOLID pods when the participants grant permission or anyone access to their data. Next, we intend to investigate in applying Blockchain technologies for handling loggings in a more transparent and secure manner. Several studies have developed tools integrating SOLID and Blockchain [49], [50].

Furthermore, the current version of TIDAL only handles static data. In the further development, we consider extending TIDAL to also handle streams of RDF data (RDF triples or graphs with temporal annotations) or real-time data processing [51]. For example, TIDAL users can synchronize their health or fitness data from their wearable devices such as mobile phones or fitness watches to their SOLID pods. These data are first converted to RDF stream data and stored in the users' pods. Then, we consider integrating with RDF Stream processing engines in TIDAL to handle the long-standing query, which is continuously executed, over RDF stream data from the distributed SOLID data pods.

Acknowledgments

Financial support for this study was provided by a grant from the Dutch National Research Agenda (NWA; project number: NWA.1418.20.006). We would like to thank our past colleagues (Federico Igne, Gianmarco Spinaci, Glenda Amaral, Kabul Kurniawan), and our tutor (Dr. John Domingue) from the International Semantic Web Summer School 2019 (ISWS 2019) for brainstorming and generating the idea. We gratefully acknowledge the time and effort devoted by Dr. Andre Dekker and Dr. Leonard Wee for their generous feedback and suggestions to help us construct and improve the platform. Special thanks are given to Vincent Emonet and Tim Hendriks for their valuable technical support for the platform.

References

[1] J. Chen, C.D. Mullins, P. Novak and S.B. Thomas, Personalized strategies to activate and empower patients in health care and reduce health disparities, *Health Education & Behavior* **43**(1) (2016), 25–34.

[2] T. Hulsen, Sharing is caring—data sharing initiatives in healthcare, *International journal of environmental research and public health* **17**(9) (2020), 3046.

[3] E. Commission, White paper: A European strategy for data, Technical Report, COM(2020) 66 final, European Commission, 2020. https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1593073685620&uri=CELEX%3A52020DC0066.

[4] E. Commission, White paper: on enabling the digital transformation of health and care in the Digital Single Market; empowering citizens and building a healthier society, Technical Report, COM(2018) 233 final, European Commission, 2018. https://ec.europa.eu/digital-single-market/en/news/communication-enabling-digital-transformation-health-and-care-digital-single-market-empowering.

[5] E. Parliament, Understanding EU data protection policy, Technical Report, European Commission, 2020. https://www.europarl.europa.eu/RegData/etudes/BRIE/2020/651923/EPRS_BRI(2020)651923_EN.pdf.

[6] E. Commission, Summary report of the public consultation on the European strategy for data, Technical Report, COM(2018) 233 final, European Commission, 2020. https://ec.europa.eu/digital-single-market/en/news/summary-report-public-consultation-european-strategy-data.

[7] D.T. for Life Sciences (DTL), Personal Health Train, https://www.dtls.nl/fair-data/personal-health-train/, Access on 12-8-2021.

[8] C. Sun, L. Ippel, J. Van Soest, B. Wouters, A. Malic, O. Adekunle, B. van den Berg, O. Mussmann, A. Koster, C. van der Kallen et al., A Privacy-Preserving Infrastructure for Analyzing Personal Health Data in a Vertically Partitioned Scenario., in: *MedInfo*, 2019, pp. 373–377.

[9] J. Van Soest, C. Sun, O. Mussmann, M. Puts, B. van den Berg, A. Malic, C. van Oppen, D. Townend, A. Dekker and M. Dumontier, Using the Personal Health Train for Automated and Privacy-Preserving Analytics on Vertically Partitioned Data., in: *MIE*, 2018, pp. 581–585.

[10] A. Jochems, T.M. Deist, J. Van Soest, M. Eble, P. Bulens, P. Coucke, W. Dries, P. Lambin and A. Dekker, Distributed learning: developing a predictive model based on data from multiple hospitals without data leaving the hospital–a real life proof of concept, *Radiotherapy and Oncology* **121**(3) (2016), 459–467.

[11] E. Mansour, A.V. Sambra, S. Hawke, M. Zereba, S. Capadisli, A. Ghanem, A. Aboulnaga and T. Berners-Lee, A demonstration of the solid platform for social web applications, in: *Proceedings of the 25th International Conference Companion on World Wide Web*, International World Wide Web Conferences Steering Committee, 2016, pp. 223–226.

[12] H.J. Pandit, A. Polleres, B. Bos, R. Brennan, B. Bruegger, F.J. Ekaputra, J.D. Fernández, R.G. Hamed, E. Kiesling, M. Lizar, E. Schlehahn, S. Steyskal and R. Wenning, Creating a Vocabulary for Data Privacy, in: *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, Springer International Publishing, Cham, 2019, pp. 714–730. ISBN 978-3-030-33246-4.

[13] Data Privacy Vocabulary (DPV) - version 0.2, Accessed on 12-08-2021.

[14] T. Symons and T. Bass, Me, my data and I: The future of the personal data economy, Technical Report, DECODE (DEecentralised Citizen Owned Data Ecosystems), 2017.

[15] M. Koscina, D. Manset, C. Negri and O. Perez, Enabling Trust in Healthcare Data Exchange with a Federated Blockchain-Based Architecture, in: *IEEE/WIC/ACM International Conference on Web Intelligence - Companion Volume*, WI '19 Companion, Association for Computing Machinery, New York, NY, USA, 2019, pp. 231–237–. ISBN 9781450369886. doi:10.1145/3358695.3360897.

[16] Own Your Data, Accessed on 12-08-2021.

[17] MIDATA: My Data - Our Health, Accessed on 12-08-2021.

[18] MedMij: Personal health data in the palm of your hand, Accessed on 12-08-2021.

[19] Digi.me, https://digi.me/, Access on 12-08-2021.

[20] Cozy Cloud, https://cozy.io/, Access on 12-08-2021.

[21] Hub of All Things (HAT), https://www.hubofallthings.com/, Access on 06-10-2021.

[22] MyDex, https://mydex.org, Access on 06-10-2021.

[23] Y.-A. de Montjoye, E. Shmueli, S.S. Wang and A.S. Pentland, openPDS: Protecting the Privacy of Metadata through SafeAnswers, *PLOS ONE* **9**(7) (2014). doi:10.1371/journal.pone.0098790.

[24] H. Janssen, J. Cobbe, C. Norval and J. Singh, Decentralised data processing: Personal data stores and the gdpr, *International Data Privacy Law* **10**(4) (2020), 356–384.

[25] The European Union's Horizon 2020 research and innovation programme, https://ec.europa.eu/programmes/horizon2020/, Access on 12-08-2021.

[26] A. Sonnino, M. Al-Bassam, S. Bano and G. Danezis, Coconut: Threshold Issuance Selective Disclosure Credentials with Applications to Distributed Ledgers, in: *Network and Distributed Systems Security (NDSS) Symposium 2019*, 2019. doi:10.14722/ndss.2019.23272.

[27] A.V. Sambra, E. Mansour, S. Hawke, M. Zereba, N. Greco, A. Ghanem, D. Zagidulin, A. Aboulnaga and T. Berners-Lee, Solid: A platform for decentralized social applications based on linked data, *MIT CSAIL & Qatar Computing Research Institute, Tech. Rep.* (2016).

[28] Solid: Your data, your choice, https://solidproject.org/, Access on 12-08-2021.

[29] S. Lohr, He Created the Web. Now He's Out to Remake the Digital World., *The New York Times* (2021). https://www.nytimes.com/2021/01/10/technology/tim-berners-lee-privacy-internet.html.

[30] F. Giunchiglia, R. Zhang and B. Crispo, Ontology driven community access control, Technical Report, University of Trento, 2008.

[31] T.M. Deist, F.J. Dankers, P. Ojha, M.S. Marshall, T. Janssen, C. Faivre-Finn, C. Masciocchi, V. Valentini, J. Wang, J. Chen et al., Distributed learning on 20 000+ lung cancer patients–The Personal Health Train, *Radiotherapy and Oncology* **144** (2020), 189–200.

[32] Z. Shi, I. Zhovannik, A. Traverso, F.J. Dankers, T.M. Deist, P. Kalendralis, R. Monshouwer, J. Bussink, R. Fijten, H.J. Aerts et al., Distributed radiomics as a signature validation study using the Personal Health Train infrastructure, *Scientific data* **6**(1) (2019), 1–8.
[33] O. Beyan, A. Choudhury, J. van Soest, O. Kohlbacher, L. Zimmermann, H. Stenzhorn, M.R. Karim, M. Dumontier, S. Decker, L.O.B. da Silva Santos et al., Distributed analytics on sensitive medical data: The Personal Health Train, *Data Intelligence* **2**(1–2) (2020), 96–107.
[34] Solid access to Pods, local file systems, and other backends via nodejs., https://www.npmjs.com/package/solid-node-client, Access on 12-08-2021.
[35] Javascript RDF library for browsers and Node.js, https://github.com/linkeddata/rdflib.js/, Access on 28-02-2021.
[36] Tripledoc - The easiest way to get started writing Solid apps., https://vincenttunru.gitlab.io/tripledoc/, Access on 28-02-2021.
[37] R. Verborgh and R. Taelman, LDflex: A Read/Write Linked Data Abstraction for Front-End Web Developers, in: *International Semantic Web Conference*, Springer, 2020, pp. 193–211.
[38] D.J. Bernstein, N. Duif, T. Lange, P. Schwabe and B.-Y. Yang, High-speed high-security signatures, *Journal of cryptographic engineering* **2**(2) (2012), 77–89.
[39] D.J. Bernstein, S. Josefsson, T. Lange, P. Schwabe and B.-Y. Yang, EdDSA for more curves, *Cryptology ePrint Archive* (2015).
[40] S. Josefsson and I. Liusvaara, Edwards-curve digital signature algorithm (eddsa), in: *Internet Research Task Force, Crypto Forum Research Group, RFC*, Vol. 8032, 2017, pp. 257–260.
[41] J. Brendel, C. Cremers, D. Jackson and M. Zhao, The provable security of ed25519: theory and practice, *IEEE Security & Privacy* (2021).
[42] TweetNaCl.js - a port of TweetNaCl / NaCl to JavaScript., https://www.npmjs.com/package/tweetnacl, Access on 28-02-2021.
[43] NaCl: Networking and Cryptography library., http://nacl.cr.yp.to/, 2016, Access on 28-02-2021.
[44] N. Noy, N. Shah, P. Whetzel, B. Dai, M. Dorf, N. Griffith, C. Jonquet, D. Rubin, M. Storey, C. Chute and M. Musen, BioPortal: Ontologies and integrated data resources at the click of a mouse, *Nucleic Acids Research* **37**(SUPPL. 2) (2009), W170–W173, Funding Information: National Center for Biomedical Ontology, under roadmap-initiative from the National Institutes of Health [grant U54 HG004028]. Funding for open access charge: National Institutes of Health [grant U54 HG004028]. doi:10.1093/nar/gkp440.
[45] Docker Remote API driver for node.js., https://www.npmjs.com/package/node-docker-api, Access on 28-02-2021.
[46] K. Donnelly et al., SNOMED-CT: The advanced terminology and coding system for eHealth, *Studies in health technology and informatics* **121** (2006), 279.
[47] Semantic - a UI framework designed for theming., https://semantic-ui.com/, 2013, Access on 20-02-2021.
[48] B.H. Bloom, Space/Time Trade-Offs in Hash Coding with Allowable Errors, *Commun. ACM* **13**(7) (1970), 422–426–. doi:10.1145/362686.362692.
[49] A. Third and J. Domingue, Decentralised Verification Technologies and the Web, in: *Media, Technology and Education in a Post-Truth Society*, Emerald Publishing Limited, 2021.
[50] M. Eisenstadt, M. Ramachandran, N. Chowdhury, A. Third and J. Domingue, COVID-19 antibody test/vaccination certification: there's an app for that, *IEEE Open Journal of Engineering in Medicine and Biology* **1** (2020), 148–155.
[51] S. Sakr, M. Wylot, R. Mutharaju, D. Le Phuoc and I. Fundulaki, Processing of RDF Stream Data, in: *Linked Data*, Springer, 2018, pp. 85–108.