



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

**Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare**



**Deliverable 6.6**  
**“Global discovery mechanism based on blockchains”**

---

**Work Package 6**  
**“Blockchains and user right management”**

### Disclaimer

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078. Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

### Copyright message

#### © FeatureCloud Consortium, 2023

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

### Document information

Grant Agreement Number: 826078		Acronym: FeatureCloud	
<b>Full title</b>	Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare		
<b>Topic</b>	Toolkit for assessing and reducing cyber risks in hospitals and care centres to protect privacy/data/infrastructures		
<b>Funding scheme</b>	RIA - Research and Innovation action		
<b>Start Date</b>	1 January 2019	<b>Duration</b>	60 months
<b>Project URL</b>	<a href="https://featurecloud.eu/">https://featurecloud.eu/</a>		
<b>EU Project Officer</b>	Christos Maramis, Health and Digital Executive Agency (HaDEA)		
<b>Project Coordinator</b>	Jan Baumbach, University of Hamburg (UHAM)		
<b>Deliverable</b>	D6.6 – Global discovery mechanism based on blockchains		
<b>Work Package</b>	WP6 – Blockchains and user right management		
<b>Date of Delivery</b>	<b>Contractual</b>	30/06/2023	<b>Actual</b> 19/01/2024
<b>Nature</b>	Report	<b>Dissemination Level</b>	Public
<b>Lead Beneficiary</b>	SBA		
<b>Responsible Author(s)</b>	Rudolf Mayer (SBA) and Walid Fdhila (SBA)		
	Dominik Heider (UMR)		
<b>Keywords</b>	Blockchains, user rights, consent management, GDPR compliance		



### History of changes

Version	Date	Contributions	Contributors (name and institution)
V0.1	30/11/2023	First draft	Rudolf Mayer (SBA) and Walid Fdhila (SBA)
V0.2	08/12/2023	Comments	Sotirios Tsepelakis (SBA)
V0.3	15/12/2023	Adressing comments, internal review, quality control	Rudolf Mayer (SBA) Dominik Heider (UMR)
V1.0	19/01/2024	Final version and approval	Rudolf Mayer (SBA) Nina Donner (concentris) Jan Baumbach (UHAM)



---

**Table of Content**

1 Table of acronyms and definitions	5
2 Objectives of the deliverable based on the Description of Action (DoA)	6
3 Executive Summary	7
4 Introduction (Challenge)	8
5 Methodology	9
6 Results	11
6.1 Query interface	11
6.2 Local query execution	13
6.3 Centralised Result Collection and Display	14
6.4 Data pipeline	15
7 Open issues	17
8 Deviations	17
9 Conclusion	17
10 References	18



## 1 Table of acronyms and definitions

BFT	Byzantine-Fault-Tree
concentris	concentris research management gmbh
.csv	comma-separated-values (text file format)
D	Deliverable
FHIR	Fast Healthcare Interoperability Resources
GDPR	General Data Protection Regulation
i2b2	Informatics for Integrating Biology to the Bedside
ICD	International Classification of Diseases
LOINC	Logical Observation Identifiers Names and Codes
ML	machine learning
MS	Milestone
PAML	privacy-aware machine learning
Patients	In this deliverable, we use the term “patients” for all research subjects. In FeatureCloud, we will focus on patients, as this is already the most vulnerable case scenario and this is where most primary data is available to us. Admittedly, some research subjects participate in clinical trials but not as patients but as healthy individuals, usually on a voluntary basis and are therefore not dependent on the physicians who care for them. Thus, to increase readability, we simply refer to them as “patients”.
SBA	SBA Research Gemeinnützige GmbH
SHRINE	Shared Health Research Information Network
SNOMED CT	Systematized Nomenclature of Medicine Clinical Terms
UHAM	University of Hamburg
UMR	Philipps Universität Marburg
WP	Work package

## 2 Objectives of the deliverable based on the Description of Action (DoA)

The main objective of WP6 is on blockchain and user right management.

*“Most important for the global success of a machine learning platform requiring user consent is the ability of users and data owners to control the data introduced, while allowing data discovery in a privacy-preserving manner. This is especially important in order to integrate as many federated machine learning nodes as possible, while being aware of privacy rights and regulations, especially the General Data Protection Regulation (GDPR) and regulations for medical data. In order to reach these goals, we will conduct research into blockchain-based technologies, especially so-called Byzantine-Fault-Tree (BFT) blockchains, in order to provide user rights management, consent and data discovery mechanisms.”*

This deliverable is about the **Global discovery mechanism based on blockchains**. It is partially related to objective 4 of this work package, which aims **“to provide a global blockchain-based mechanism that allows for the exchange of analysis-relevant meta-information while respecting patient privacy”**.

It is further partially related to task 4 of this work package, **Enabling Global Discovery by facilitating data sharing**:

*“In this task the previously developed blockchain-based user-rights-mechanisms will be extended with additional metadata which helps to describe an information particle without revealing sensitive information (SBA, RI). This will foster global discovery, allowing many partners to share metadata like data classification or consent/sharing rights, without requiring consent to do this. Furthermore, SBA and UHAM will define methods for enabling patients to participate in this process by actively allowing them to apply changes. During the prototype phase, it will also be considered which level of detail will be appropriate for non-expert users in order to be able use the FeatureCloud platform (MUG, UHAM).”*

### 3 Executive Summary

In many analyses, investigators do not know who else is in the possession of suitable data to carry out a specific study, thus not allowing them to assess the feasibility of a study up-front. Therefore, a mechanism for **global discovery** of suitable data, to assess these numbers is of great relevance to a platform like FeatureCloud.

The initial solution foreseen was to utilise blockchains to share meta-information on the data that would allow answering these feasibility questions, without revealing sensitive information. This would integrate with the block-chain based consent management to allow discovery of data that can be leveraged for further analysis.

During the project, it has shown to be **infeasible to manage the user consent via blockchain**, due to data confidentiality concerns as detailed in Deliverables D6.2 and D6.5; managing consent thus stays a matter of the local participants, either managed directly by them, or the patients themselves, but outside of the blockchain, as only management transactions of the consent are committed to the blockchain. Checking for consent before the data are used in analysis is also a matter of the local participants.

Analogous to this considerations, data discovery via the blockchain itself was considered problematic, as it would either require meta-data on the micro-level of the individual patient, which is a large amount of data, and which could further enable inference attacks, or to pre-compute statistics for the available demographic, medical or other criteria, which is intractable for any reasonable number of criteria. Therefore, the consortium decided to provide a federated querying mechanism instead to assess feasibility of studies and to enable global data discovery.

This federated querying system was implemented as a FeatureCloud app that provides a graphical user interface to compose queries. The app comes in two modes. In the first mode, it just allows feasibility querying and provides statistics, i.e. it provides the user with the **count results** of the querying, taking potential privacy concerns such as a low number of results etc. into account; this mode enables assessing the feasibility of an analysis.

In the second mode, the app actually performs the **data selection and provisioning**, and returns the retrieved data as a result to be picked up by the next app (e.g. pre-processing, or model learning), meaning, it becomes a standard app that can be integrated within any other data processing workflow.

## 4 Introduction (Challenge)

In many distributed data analysis scenarios, investigators (study coordinators) do not know who else is in the possession of suitable data to carry out a specific study, thus not allowing them to assess up-front whether a specific investigation is feasible in terms of a minimum number of matching participants being available. One aspect to this discovery of data is generally knowing which participants to the federated learning hold which type of data, i.e. knowing that a certain participant (e.g. a hospital) has data on blood lab tests, where certain information is measured with a certain unit. This already allows to preselect which participants are in principle compatible to be included in a specific investigation. However, often, the specific number of patients matching more fine-grained criteria is needed. In many cases, the criteria to include patients might be quite narrow, e.g., those who have a specific disease, in some specific risk group, or received specific medication.

The process of querying and query refinements is often called a “feasibility query” (Wettstein, R. et al., 2021, Gruendner, J. et al., 2022), and is performed before the actual data analysis, and might also be performed iteratively. This is the case if the number of matching individuals does not meet the expectations, the cause of it being the selection criteria either too restrictive (thus resulting in a too low number of matching patients), or maybe too wide.

A mechanism to estimate these numbers is thus of great relevance to a federated learning (or more general data analysis) system like FeatureCloud. The initial assumption at the time of envisioning the overall FeatureCloud system, during the proposal stage, was that blockchain technology would be utilised to provide a method for sharing meta-information on the data, without revealing sensitive information. In other words, the blockchain could contain sufficient information to answer feasibility queries, without containing actual data. Further, consent checking, and more advanced user rights management, was thought to be enabled via blockchain as well.

During the implementation of the project, however, it has shown to be infeasible to pursue this idea. Two main approaches to implement feasibility query answering via blockchain conceivable would either require to (i) publish meta-data on the micro-level of the individual patient or even more fine-granular on specific subsets of data on that patient, or (ii) to pre-compute statistics per site for some criteria.

The first option is rather infeasible, as it means that on the one hand, the amount of data that needs to be written on-chain is rather large, as it is related to the total number of patients, and also needs to be updated when the data changes or increases, including deletion of some data. On the other hand, having too much (meta-)data on the microdata level, i.e. the individual patient, may become a privacy risk.

The second option is also infeasible, as the number of potential attributes (e.g. demographic, but also health-related, etc.) is intractable, and one could thus only provide an incomplete subset of possible filtering queries. Again as with the first option, the meta-data would need to be updated regularly to reflect a somewhat current patient count, which conflicts with the idea of immutability of the blockchain.

Finally, the consent information itself is not stored on the blockchain, due to data confidentiality concerns as detailed in Deliverables D6.2 and D6.5. It was decided that instead; managing consent stays a matter of the local federated learning participants (e.g. hospitals), either managed



directly by them, or the patients themselves, but outside of the blockchain. Eventually, management transactions of the consent are committed to the blockchain, which enables retrospective auditing of data usage.

Therefore, it was decided to pursue an alternative approach, and instead of exposing meta-data on the blockchain to provide a federated querying system that allows to determine, unbounded on the type of data and available attributes, the number of patients that match a certain set of criteria. These attributes can be used to select acceptance criteria, and are dynamically selected by the researchers in an interactive interface.

## 5 Methodology

Federated queries are studied in several domains. Information Retrieval, which generally deals with discovering of and searching in unstructured information, such as text, or multimedia objects, is a well-known example. There, federated queries are most commonly referred to as Federated Search (Hsiao, D.K., 1992). Federated queries is also a topic for structured databases, when querying distributed databases is of relevance (Shokouhi, M., Si, L., 2011).

In health data, several systems and customised solutions exist. One frequent assumption is that the participating nodes, e.g. hospitals or clinics, already work with the same, unified data format and semantic schema. Well-known solutions of monolithic systems are e.g. i2b2 (i2b2) or SHRINE (“Shared Health Research Information Network”) (Weber, G.M. et al., 2009).

These two solutions assume that all sites will utilise the same data scheme, the same data exchange format, and the same software solution, which is not realistic in many potential collaboration settings, where participants are likely to use a plethora of different solutions and data schemes. We thus pursue an approach that builds on an intermediate, uniform data representation that allows for integration of various original data formats and systems, but querying of the data can happen outside of any specific software solution.

Several vocabularies have been proposed for integrating data respectively standardising data exchange, in several different domains. The medical domain primarily refers to these vocabularies as coding systems, and depending on the specific information type and use case, several different systems have been developed and are in use. Among those, SNOMED CT (Systematized Nomenclature of Medicine Clinical Terms)<sup>1</sup> defines codes describing **clinical concepts**, while LOINC (Logical Observation Identifiers Names and Codes)<sup>2</sup> focuses on **laboratory and clinical observations**. ICD (International Classification of Diseases)<sup>3</sup>, currently in its 11<sup>th</sup> revision, defines codes for health conditions like **diseases** and **injuries**.

There are several initiatives trying to provide a framework to integrate multiple coding schemes. BioCypher<sup>4</sup> is a knowledge graph built on the Biolink model<sup>5</sup>, which is a large collection of concepts

---

<sup>1</sup> <https://www.snomed.org/>

<sup>2</sup> <https://loinc.org/>

<sup>3</sup> <https://icd.who.int/en>

<sup>4</sup> <https://biocypher.org/>

<sup>5</sup> <https://biolink.github.io/biolink-model/docs/Disease.html>

---

using various standard vocabularies, such as the above-mentioned SNOMED CT, LOINC, or ICD. It also allows adding self-defined codes.

FHIR (Fast Healthcare Interoperability Resources)<sup>6</sup> sets a standard for healthcare information exchange and interoperability. FHIR is not an ontology in the traditional sense, but can reference other ontologies and data vocabularies, including the above-mentioned SNOMED CT, or LOINC. Also FHIR is designed to be extensible and allows to add additional or custom vocabularies. Due to the maturity and large community support, it was decided to adopt FHIR as the intermediate representation within FeatureCloud. This decision, however, does not rule out using other schemes, as the querying interface is designed with flexibility and the option for customisation.

A few software solutions for federated query systems exist, such as the above mentioned built-in ones in SHRINE or i2b2. Further, other initiatives such as the German “Netzwerk Universitätsmedizin”<sup>7</sup> provide data models, a query interface, and query execution. However, these are rather large, custom-built solutions for a specific purpose, and are difficult to integrate into the FeatureCloud platform. Therefore, it was decided to implement a lightweight solution for FeatureCloud, drawing on best-practices from various previous efforts.

Besides the querying allowing global discovery of data across the various participating sites, it is also important to safeguard the global discovery system from revealing additional information. To this end, the query interface must consider cases where the exact numbers of results would be revealing information on the underlying data, which can be the case if it is possible to single out data points in the dataset. This is specifically important for outliers (e.g. the single patient above a certain age in a specific region, ...).

The system should further prevent queries that would reveal marginal (e.g. single records) differences between queries to allow inference of the membership of specific records in the database.

---

<sup>6</sup> <https://fhir.org/>

<sup>7</sup> <https://github.com/num-codex>

## 6 Results

### 6.1 Query interface

The first step lies in the methodology for composing queries and executing them to retrieve relevant data from the participating sites. While a direct query input might be an option that offers great flexibility, fine-tuned control and usage of advanced query patterns, it demands a certain level of syntax familiarity and is more prone to mistakes.

Given that the end-users are often domain experts, rather than computer scientists, the focus was thus on an easy-to-use graphical user interface. The user interface simplifies the query creation process by using simple menus and GUI interaction mechanisms, thus making it more intuitive, less error-prone and less reliant on technical expertise.

The implemented query interface, shown in Figures 1 and 2, allows an interactive selection of several criteria to filter the data on; these include demographic, as well as domain-specific ones, such as specific medical conditions, medication, or diseases. The interface is dynamically created, based on a textual definition file; this definition can be created from a specific FHIR data scheme, and thus can be dynamically adapted to the specific data structures used. While the interface screenshots are specific for medical data in FHIR, it can also be adapted to different types of domains and data.

The interface allows for interactive selection of different criteria to filter the data subjects. Depending on the data types, the user can select from pre-existing possible values, or provide e.g. range queries for numerical data. The interface further enables the user to logically combine multiple criteria.

### Attributes

---

Expand All +Reset Form ↕

#### Patient

**Physical Characteristics**^

**Age**  

>=

20

**Gender**  

AND

<

60

[Add +](#)

**Demographic**▼

#### Observation

**Blood Glucose**^

**Fasting Blood Glucose (mg/dL)**  

>=

70

AND

<=

85.5

**Glycated Hemoglobin (%)**

[Add +](#)

Figure 1: Query interface

Expand All +
Reset Form ↕

Patient

**Physical Characteristics**

Age

Gender

>

30

AND

<

50

[Add +](#)

**Demographic**

---

Observation

**Blood Glucose**

**CBC**

Red Blood Cells (x10<sup>12</sup>/L)

>

4

AND

<

5.9

[Add +](#)

Hematocrit (%)  
 Neutrophils (x10<sup>9</sup>/L)  
 Mean Corpuscular Volume (fL)

Basophils (x10<sup>9</sup>/L)  
 Monocytes (x10<sup>9</sup>/L)  
 Hemoglobin (g/dL)  
 White Blood Cells (x10<sup>9</sup>/L)  
 Platelets (x10<sup>9</sup>/L)

Eosinophils (x10<sup>9</sup>/L)  
 Lymphocytes (x10<sup>9</sup>/L)

**Figure 2:** Query interface, cont.

## 6.2 Local query execution

In the next step, the federated query system transforms the selected criteria into an executable FHIR query. As the retrieved data are in FHIR format, the application will be able to extract specific attributes based on the composed query. Patient data resides locally at the participating sites, thus a local processing is needed to extract these data that fulfil the given query.

To handle different local formats, we also provide an example converter from a relational database schema to the FHIR format through the usage of mapping of the corresponding fields. This approach allows that, regardless of the data's existing schema and format, we can effectively process and transform the attributes aligned with the provided query.

### 6.3 Centralised Result Collection and Display

During the stage of performing feasibility queries, the system shows a centralised aggregation of the results of the federated query - the system presents statistics on the counts of the matching data.

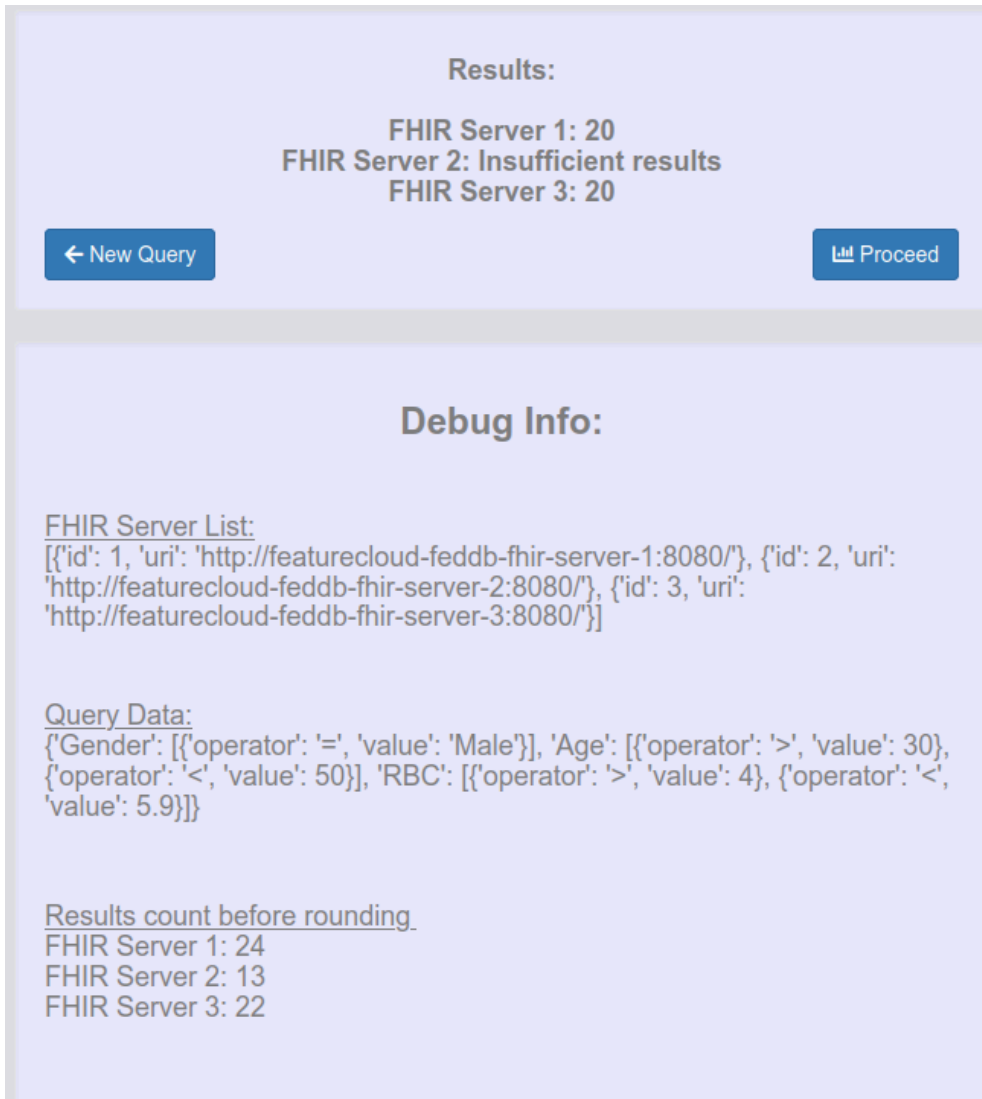
Result suppression for small counts, and result perturbation for reducing the level of detail to potentially carry out attacks to single out individuals, are of great importance to retain the privacy of patients. In the case where the result size is too small, results, either globally or locally per site, will be suppressed; in other cases, results are rounded to perturb them.

Figures 3 and 4 show such cases. Before displaying the aggregate results, they are rounded as seen in Figure 3 (for illustration, we also show the original counts in this figure - they are unavailable in normal operating mode).



**Figure 3:** Query results, obfuscated

Figure 4 shows a case when one specific site has too few individuals to be displayed, i.e. they are below a definable threshold (in this example, the threshold is 20). In this case, the numbers are suppressed.



The screenshot displays a web interface with two main sections. The top section, titled "Results:", shows the following data: "FHIR Server 1: 20", "FHIR Server 2: Insufficient results", and "FHIR Server 3: 20". Below this are two buttons: "← New Query" on the left and "Proceed" on the right. The bottom section, titled "Debug Info:", contains three sub-sections: "FHIR Server List:" with a list of three server URIs, "Query Data:" with a JSON object containing filters for Gender, Age, and RBC, and "Results count before rounding:" with counts for each server (24, 13, and 22 respectively).

```
Results:
FHIR Server 1: 20
FHIR Server 2: Insufficient results
FHIR Server 3: 20

Debug Info:

FHIR Server List:
[{"id": 1, "uri": "http://featurecloud-feddb-fhir-server-1:8080/"}, {"id": 2, "uri": "http://featurecloud-feddb-fhir-server-2:8080/"}, {"id": 3, "uri": "http://featurecloud-feddb-fhir-server-3:8080/"}]

Query Data:
{"Gender": [{"operator": "=", "value": "Male"}], "Age": [{"operator": ">", "value": 30}, {"operator": "<", "value": 50}], "RBC": [{"operator": ">", "value": 4}, {"operator": "<", "value": 5.9}]}

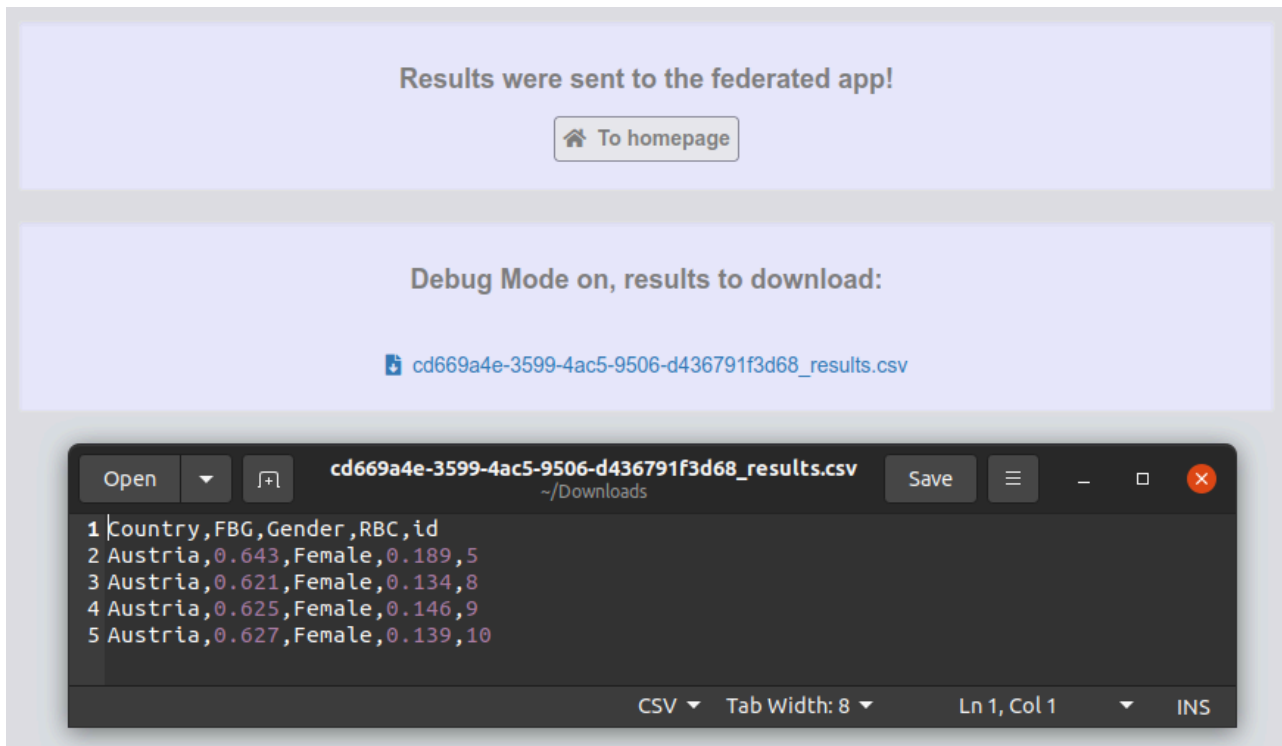
Results count before rounding.
FHIR Server 1: 24
FHIR Server 2: 13
FHIR Server 3: 22
```

**Figure 4:** Query results, obfuscated

## 6.4 Data pipeline

Finally, once the feasibility query stage is completed and final criteria have been selected, the system can be used in a mode that results are not just displaying counts, but actually select and pass forward the data. To this end, the result of the querying is the required patient data. To allow subsequent apps in the workflow to utilise the data, a conversion step transforms the locally returned FHIR results into a CSV format that apps can easily read. This data is then the input to the next app in the workflow, e.g. a data normalisation app, or a machine-learning model training app.

Figure 5 shows an example debug mode output that displays the actually selected data; again, in the actual deployment, this information is not available, and only the output of the querying in CSV format is passed on.



**Figure 5:** Query results in CSV format, e.g for use of further steps in the workflow.



## 7 Open issues

The chosen solution for global discovery expects all local sites to be able to produce data in the same representation, and following the same vocabulary. While approaches that would dynamically translate between different formats are conceivable, we expect that the conversion would be rather on an interval basis, with the system actually querying a local, transformed copy of the data from a given time point. Thus, data might not always reflect the exact state of the system at the time of the query. This might however also be beneficial, as it will further reduce the capabilities of an attacker issuing multiple queries to infer detailed information on particular records.

The definition of how to map the original data formats and vocabularies to the one used by FeatureCloud is not addressed within this project. While this is a highly relevant point, it is addressed within several other initiatives in e.g. the medical domain, and also beyond the scope of the FeatureCloud project, and thus deliberately left out.

## 8 Deviations

The initial plan of the global data discovery system, as described in the proposal, envisions a major role for blockchain for this aspect. However, during the project implementation, it became apparent that such a solution might be impractical due to the large amount of data that needs to be stored on change, due to the mutability of the data to be discovered, due to potential additional privacy risks emerging from the on-chain data, and the lack of direct consent representation on-chain.

Therefore, we decided to refocus this task with a different underlying technology, providing a proof-of-concept federated querying mechanism, and demonstrating how it can be integrated within the overall FeatureCloud platform. This solution is relevant to the overall system, and is an important aspect to provide a holistic and integrated solution to data discovery within the project, as federated data querying is a prerequisite to a practical federated learning system.

## 9 Conclusion

In this deliverable, we outlined the concept and implementation of a mechanism to discover data at participating federated clients. While the initial design for this mechanism was to be based on blockchains, eventually we developed a system based on federated querying. This is motivated on the one hand by the amount of data that would need to be written to the blockchain, and concerns over changing data respectively inference from the data.

The federated querying system is fully integratable into the FeatureCloud platform, and its provision via apps allows users to easily use the data selection to be part of any workflow. To mitigate inference risks from the discovery mechanism, we limit the precision of information displayed towards the end users.

## 10 References

Gruendner, J., Deppenwiese, N., Folz, M., Köhler, T., Kroll, B., Prokosch, H.-U., Rosenau, L., Rühle, M., Scheidl, M.-A., Schüttler, C., Sedlmayr, B., Twrdik, A., Kiel, A., Majeed, R.W., **2022**. The Architecture of a Feasibility Query Portal for Distributed COVID-19 Fast Healthcare Interoperability Resources (FHIR) Patient Data Repositories: Design and Implementation Study. *JMIR Med Inform* 10, e36709. <https://doi.org/10.2196/36709>

Hsiao, D.K., **1992**. Federated databases and systems: Part I - A tutorial on their data sharing. *VLDB Journal* 1, 127–179. <https://doi.org/10.1007/BF01228709>

i2b2: Informatics for Integrating Biology to the Bedside, Partners Healthcare Systems, n.d. URL <http://www.i2b2.org>

Shokouhi, M., Si, L., **2011**. Federated Search. *FNT in Information Retrieval* 5, 1–102. <https://doi.org/10.1561/1500000010>

Weber, G.M., Murphy, S.N., McMurry, A.J., MacFadden, D., Nigrin, D.J., Churchill, S., Kohane, I.S., **2009**. The Shared Health Research Information Network (SHRINE): A Prototype Federated Query Tool for Clinical Data Repositories. *Journal of the American Medical Informatics Association* 16, 624–630. <https://doi.org/10.1197/jamia.M3191>

Wettstein, R., Hund, H., Kobylinski, I., Fegeler, C., Heinze, O., **2021**. Feasibility Queries in Distributed Architectures – Concept and Implementation in HiGHmed, in: *Studies in Health Technology and Informatics*. IOS Press. <https://doi.org/10.3233/SHTI210061>