Real-world and Scalable Deployment of Secure Multiparty Computing in Healthcare

Paul Bunn[1], Howard S. Gordon MD[2], Quinn Grier[3], Jacob Kean[4], Abel Kho[5], Steve Lu[6], Rafail Ostrovsky[7], Xiao Wang[8], Jennie Rogers[9]

[1,3,6] Stealth Software Technologies, Inc.
[2] Department of Medicine, University of Illinois Chicago. Medical Service, Jesse Brown VA Medical Center.
[4] Division of Epidemiology, Department of Internal Medicine, University of Utah
VA Informatics and Computing Infrastructure, VA Salt Lake City Health Care System
[5] Feinberg School of Medicine, Northwestern University.
[7] Department of Computer Science and Department of Mathematics, UCLA. Work done while consulting for Stealth Software Technologies, Inc.
[8,9] Department of Computer Science, Northwestern University.

1. Background/Introduction

Healthcare data are often fragmented across multiple settings, thereby limiting care coordination, accurate estimation of population health measures, and the generation of a complete picture of the health of individuals. As a highly regulated industry, healthcare institutions often cite technical, privacy, and data governance concerns as impediments to data sharing and access across care settings.

The Chicago Area Patient-Centered Outcomes Research Network, or CAPriCORN, is a multi-institutional clinical research network representing an ethnically and socioeconomically diverse population in the Chicago area that has successfully established a set of legal agreements and governance structures to enable data sharing for multi-institutional research. Through the real-world experience of executing multi-institutional data queries and analyses, the CAPriCORN network necessarily developed and deployed Privacy Enhancing Technologies (PETs) such as Privacy Preserving Record Linkage, and served as a key development partner to develop, test, and deploy an initial pilot of secure multi-party computation (MPC). This pilot is, to our knowledge, the only HIPAA-approved use of MPC to date.

In this white paper, we introduce Catalyst, a second-generation framework for bringing secure analytics to federated electronic health records (EHRs). Catalyst brought together a team of investigators experienced in the development and deployment of Privacy-Enhancing Technologies (VaultDB, PULSAR) to create a novel, secure, and scalable multi-party computing platform for the query and analysis of health records from multiple healthcare institutions while keeping individual records at their site of origin. In addition to providing cryptographic security, Catalyst overcame additional barriers that hinder research on health data, demonstrating that MPC can address the legal and technical barriers to entry to conduct such studies in real-world health settings.

Catalyst was designed with flexibility, scalability, and usability in mind so that it could be incorporated as a reproducible, standard process for healthcare organizations seeking to do aggregate analysis on their combined records. In this white paper, we demonstrate how these features were showcased in the development and deployment of a pilot study across multiple healthcare settings: existing healthcare institutions, including those that already are collaborating with our team (as part of the CAPriCORN network), the Veterans Health Administration (VHA), and public health organizations. For example, the study featured a novel browser-based portal that we developed that allowed study sites to encrypt and distribute secret shares of their private data to the Catalyst sites that performed the MPC protocols.

1.1 Acknowledgements

The authors thank the valuable contributions of the following collaborators:

The findings reported in this report were enabled through a collaboration with the Chicago Area Patient-Centered Outcomes Research Network (CAPriCORN). CAPriCORN is a partnership between healthcare and research institutions that provides data through a federated harmonized common data model and works jointly with a Patient Community Advisory Committee, community-based organizations (CBOs), and non-profit organizations committed to enabling and delivering patient-centered clinical research and public health projects. We acknowledge CAPriCORN's partners, the Chicago Area Institutional Review Board (CHAIRb), which serves as the central IRB of record for CAPriCORN-supported research, and the Medical Research Analytics and Informatics Alliance (MRAIA), which serves as the network's honest data broker.

We thank Marc Rosenman, MD (Ann & Robert H. Lurie Children's Hospital of Chicago), Raj C. Shah, MD (Rush University Medical Center), Amro Hassan, MSC, MSE (AllianceChicago), Vesna Mitrovic, MS (Northwestern University), Andrew Hamilton, RN, BSN, MS (AllianceChicago), and Anthony Solomonides, PhD, MSc(Math), MSc(AI), FAMIA, FACMI (Endeavor Health) for their helpful contributions.

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA) under Contract #HR001120-C-0087. Work of Jennie Rogers was also supported by NSF awards #CNS-1846447 and NSF #CNS-2016240. Work of Xiao Wang was also supported by NSF award #2236819. Work of Jacob Kean was supported using resources and facilities of the Department of Veterans Affairs (VA) Informatics and Computing Infrastructure (VINCI) under ORD 24-VINCI-01. The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

Distribution Statement A. Approved for public release: distribution is unlimited.

2. Methods

2.1 Engaging with Participating Institutions

In this section, we introduce the design of a Catalyst query and the steps we took to integrate MPC into a healthcare ecosystem. We took a modular approach that enabled participating institutions to fine-tune their involvement based on their own goals, security posture, and computing resource availability.

We reached out to seven CAPriCORN sites with the aim of conducting a new MPC study while lowering the barrier of entry from both a data-use perspective and an integration perspective. In a Catalyst query, no information is revealed to anyone except that which can be deduced from the outputs of the study. For interested sites, we held a series of virtual demonstrations and question and answer questions with key CAPriCORN committees (Steering and Informatics) to describe the study design and overall data flows. Figure 1 shows how Catalyst shifts our trust model from that of a conventional data federation to one where data partners work collaboratively to answer queries securely without the assistance of an honest broker.

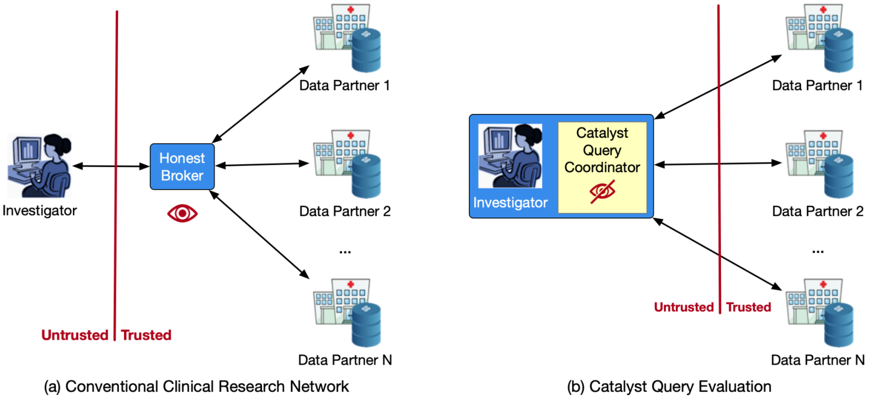


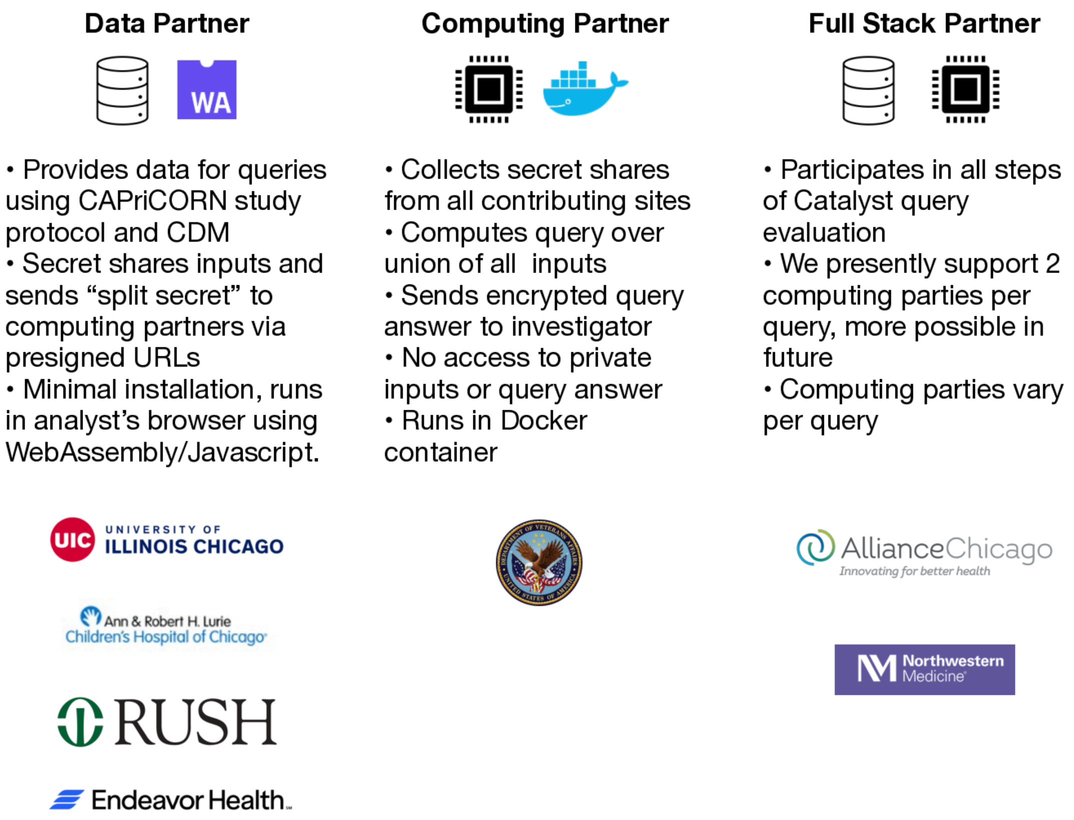
Figure 1: Query workflow with and without Catalyst

To achieve this, Catalyst uses cryptographic protocols – secure multiparty computation (MPC) -- to run a query over the union of the private inputs of our data partners. Secure multiparty computation is a maturing cryptography technology that allows multiple parties to perform joint computations with inputs from every participant. Its security ensures that no one's private inputs are disclosed to any other participants, yet the protocol is able to compute over these private inputs and deliver the computation output. The MPC protocol used for this trial is based on an optimized version of the classical Yao's garbled circuit[1] protocol. We deployed this system over six data partners by building atop the common data model (CDM) and IRB framework already established by CAPriCORN. In our deployment, we demonstrated how this system efficiently and correctly evaluates public health surveillance queries.

Partner Roles

Each participating site elected one of three roles. We describe each below and show each partner in the role with which they participated in this study. We encouraged our partners to participate at the level in which they are most comfortable at the time of the secure computation.

[1] Yao, Andrew Chi-Chih (1986). "How to generate and exchange secrets". 27th Annual Symposium on Foundations of Computer Science (SFCS 1986). pp. 162–167.



Data Partner

- Provides data for queries using CAPriCORN study protocol and CDM
- Secret shares inputs and sends “split secret” to computing partners via presigned URLs
- Minimal installation, runs in analyst’s browser using WebAssembly/Javascript.

Computing Partner

- Collects secret shares from all contributing sites
- Computes query over union of all inputs
- Sends encrypted query answer to investigator
- No access to private inputs or query answer
- Runs in Docker container

Full Stack Partner

- Participates in all steps of Catalyst query evaluation
- We presently support 2 computing parties per query, more possible in future
- Computing parties vary per query

WA
UIC UNIVERSITY OF ILLINOIS CHICAGO
Ann & Robert H. Lurie Children's Hospital of Chicago
RUSH
Endeavor Health.
AllianceChicago
Innovating for better health
Northwestern Medicine

Figure 2: Role descriptions and participating institutions

Data partners contribute data that will be searched over by one or more Catalyst queries. They do not participate in the secure query evaluation over the union of their data. In these sites, data analysts run SQL queries provided by Catalyst and upload the results to a browser-based portal. This establishes a simple, HIPAA-compliant, auditable pipeline with which they enable their EHRs to be used in a way where only the query results are shared. The portal uses client-side encryption to generate secret shares of their private inputs that Catalyst distributes to the sites that will complete the study under MPC. We will describe the Catalyst Portal and our protocol for distributing secret shares in Section 3.1. Participating as a data partner was attractive to many sites because it required minimal setup and was easy to use. Our data partners for this study were: the University of Illinois at Chicago, Lurie Children’s Hospital, Rush University Medical Center, and Endeavor Health.

A *computing partner* collects secret shares from the data partners and full-stack partners and runs the study queries under MPC. Catalyst evaluates queries using EMP Toolkit’s semi-honest 2-party computation protocols, but this framework is also extensible to other protocols. It leverages the VaultDB query processing backend; this system has also been used to run oblivious queries in maliciously secure *n*-party computation and zero-knowledge proofs. In our pilot, the Veteran Health Administration of the U.S. Department of Veterans Affairs (VA) served as a computing party. We deployed Catalyst using a Docker container within their Red Hat Enterprise Linux ecosystem. It is connected to a full-stack partner to run the provided MPC protocols.

Full-stack partners perform both of the roles described above. These sites usually had two team members working on each task. The first would complete the data contribution protocol using the Catalyst portal and this was usually performed by a data analyst. Members of the IT team assisted with the computing phase task owing to the need to initiate connections, manage ports, et cetera. One could imagine efforts in future versions of Catalyst to integrate these responsibilities more tightly. Northwestern Medicine (NM) and the Alliance of Chicago (AC) were full-stack partners in this pilot. We evaluated the system with our computing parties as NM-AC and NM-VA. When a full-stack site was not involved in participating in a particular secure computation, it would assume the role of data partner for that query.

3. Query Evaluation Workflow

The phases of the Catalyst processing pipeline begin with each data contributor secret-sharing their data and distributing these shares to the MPC compute nodes. We describe this data distribution in Section 3.1, and then provide an overview of our secure computation phase in Section 3.2.

3.1 Secret Sharing and Distributing the Data

Each data (and full-stack) partner starts with an analyst locally running specified queries (e.g., SQL) on their site's database, which outputs the query results to a CSV file. The analyst then visits the Catalyst Portal to secret-share these query results (representing this site's private inputs). This process guarantees that no non-colluding computing party has enough information with which to reveal these private, site-level inputs.

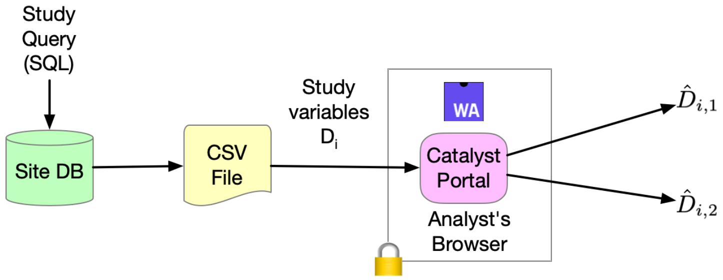


Figure 3: Secret sharing life cycle for data partner i who provides D_i for query evaluation.

Figure 3 outlines our secret sharing phase. In this example, data provider *i* provides their records D_i. We designate encrypted data with a hat over its name, e.g., $\hat{D}_{i,2}$.

We now describe our HIPAA-compliant mechanism for distributing secret shares for use in Catalyst query evaluation. Recall that the portal encrypts the data client-side so that each site's inputs are never readable to anyone except the data partner creating them. Hence, their values do not leave their site of origin. Figure 4 diagrams our workflow for this.

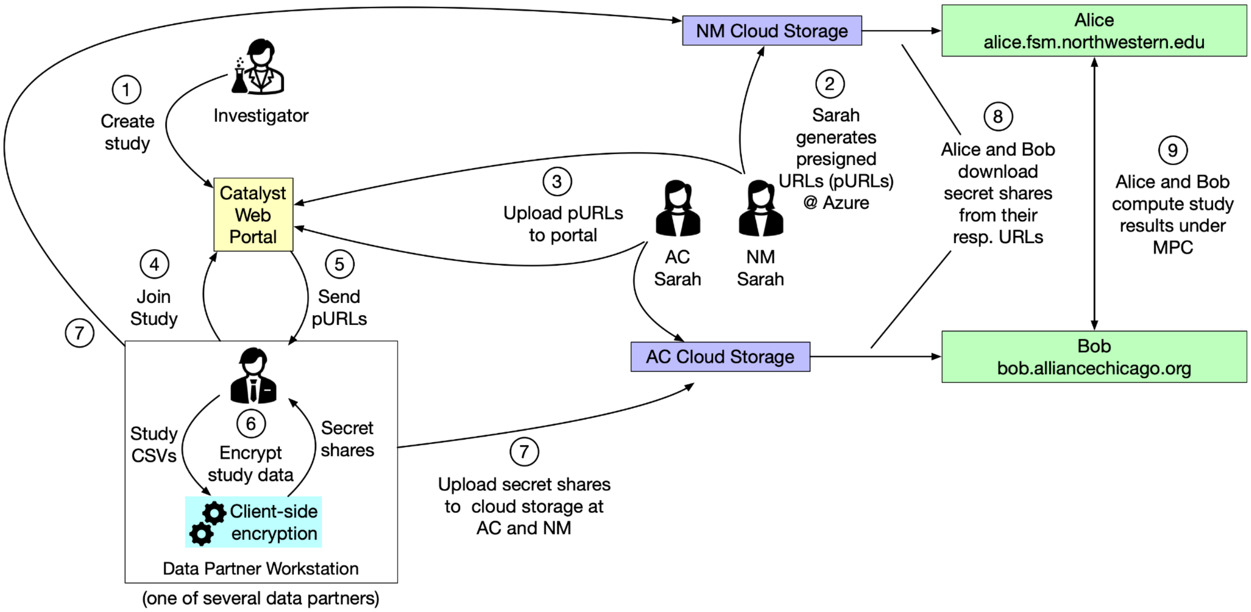
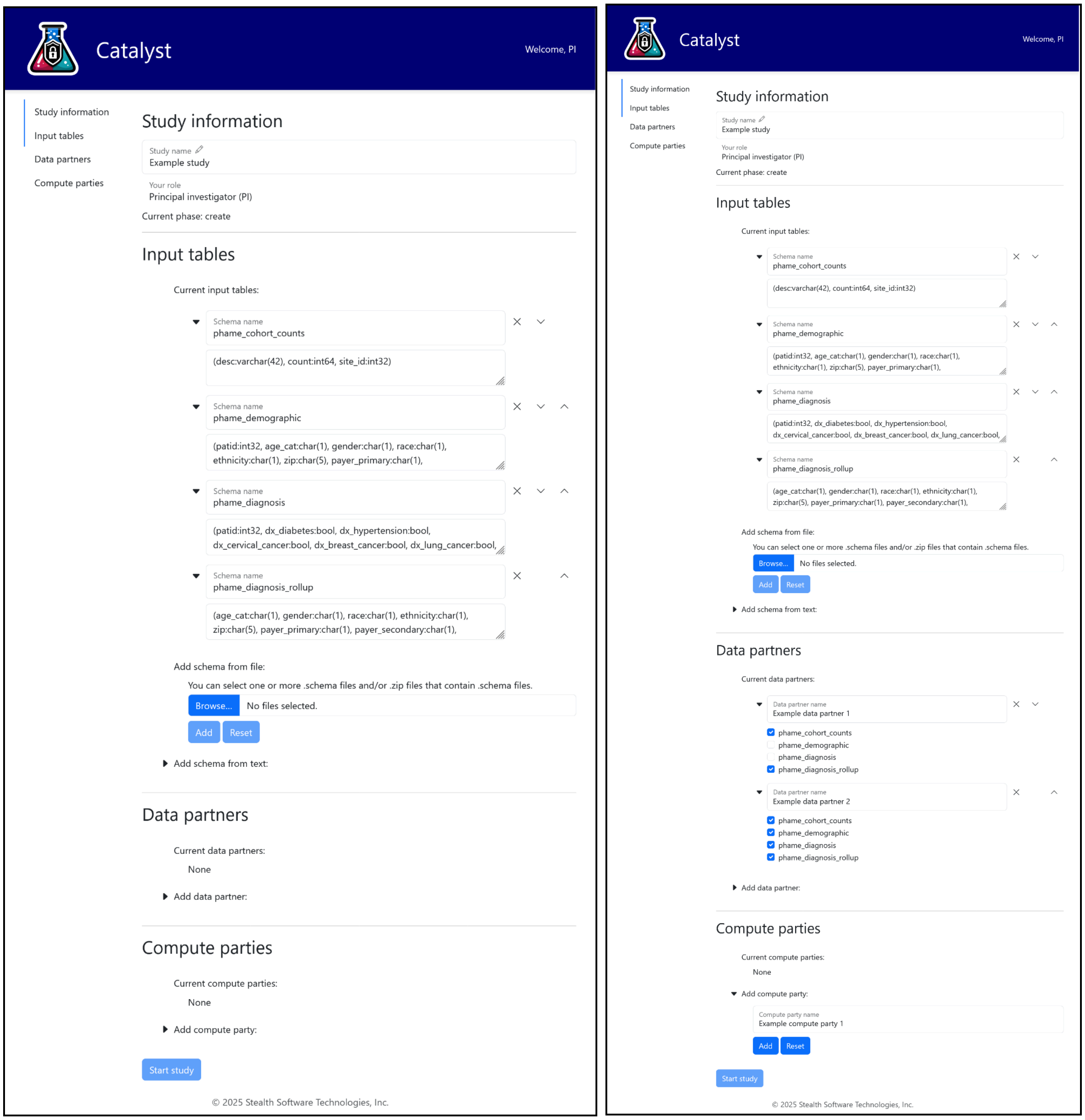


Figure 4: Workflow for distributing secret shares. Shown with computing parties AC and NM.

When an investigator creates a study in the Catalyst portal, this software generates pre-signed URLs to upload their encrypted shares to write-only storage buckets in the cloud. To create a study, the investigator navigates to the study creation page on the portal and fills out the important bits of the study, the schema of the data tables, and finally indicates the various data and compute partners. A full-stack contributor can be simply expressed by including them in both sections. Each data partner will also have checkboxes marking which tables each one will contribute. We show two screenshots of adding the schema and partners in Figure 5.



Catalyst

Welcome, PI

Study information
Input tables
Data partners
Compute parties

Study information

Study name
Example study

Your role
Principal investigator (PI)

Current phase: create

Input tables

Current input tables:

Schema name
phame_cohort_counts

(desc:varchar(42), count:int64, site_id:int32)

Schema name
phame_demographic

(patid:int32, age_cat:char(1), gender:char(1), race:char(1), ethnicity:char(1), zip:char(5), payer_primary:char(1),

Schema name
phame_diagnosis

(patid:int32, dx_diabetes:bool, dx_hypertension:bool, dx_cervical_cancer:bool, dx_breast_cancer:bool, dx_lung_cancer:bool,

Schema name
phame_diagnosis_rollup

(age_cat:char(1), gender:char(1), race:char(1), ethnicity:char(1), zip:char(5), payer_primary:char(1), payer_secondary:char(1),

Add schema from file:

You can select one or more .schema files and/or .zip files that contain .schema files.

Browse... No files selected.

Add Reset

Add schema from text:

Data partners

Current data partners:

None

Add data partner:

Compute parties

Current compute parties:

None

Add compute party:

Start study

© 2025 Stealth Software Technologies, Inc.

Catalyst

Welcome, PI

Study information
Input tables
Data partners
Compute parties

Study information

Study name
Example study

Your role
Principal investigator (PI)

Current phase: create

Input tables

Current input tables:

Schema name
phame_cohort_counts

(desc:varchar(42), count:int64, site_id:int32)

Schema name
phame_demographic

(patid:int32, age_cat:char(1), gender:char(1), race:char(1), ethnicity:char(1), zip:char(5), payer_primary:char(1),

Schema name
phame_diagnosis

(patid:int32, dx_diabetes:bool, dx_hypertension:bool, dx_cervical_cancer:bool, dx_breast_cancer:bool, dx_lung_cancer:bool,

Schema name
phame_diagnosis_rollup

(age_cat:char(1), gender:char(1), race:char(1), ethnicity:char(1), zip:char(5), payer_primary:char(1), payer_secondary:char(1),

Add schema from file:

You can select one or more .schema files and/or .zip files that contain .schema files.

Browse... No files selected.

Add Reset

Add schema from text:

Data partners

Current data partners:

Data partner name
Example data partner 1

☑ phame_cohort_counts
☐ phame_demographic
☐ phame_diagnosis
☑ phame_diagnosis_rollup

Data partner name
Example data partner 2

☑ phame_cohort_counts
☑ phame_demographic
☑ phame_diagnosis
☑ phame_diagnosis_rollup

Add data partner:

Compute parties

Current compute parties:

None

Add compute party:

Compute party name
Example compute party 1

Add Reset

Start study

© 2025 Stealth Software Technologies, Inc.

Figure 5: Creating a study in the Catalyst portal

The portal provisions these storage buckets with a delegate at each computing site that we call Sarah. We show this in Steps 1-3 in Figure 4. We implemented this in Microsoft Azure using PHI-approved storage. Although not required to use these resources for de-identified data, we opted to use the most secure options available.

When a data partner uploads their data with the portal, it sends a pair of pre-signed URLs for each table (or CSV file) they are uploading from the resources that the Sarahs provisioned. The portal locally generates the shares, $(D_{i,1}, D_{i,2})$ in Figure 3, once per table, and uploads the shares to their corresponding storage buckets. We show this in Step 7 of Figure 4. In our example, Site *i* might upload all of its $D_{i,1}$ shares to NM and its $D_{i,2}$ shares to AC. Hence, neither computing party can see the other's shares, thus ensuring that accidental share recovery cannot occur.

After all data partners have uploaded their data to the portal, the two computing parties being employed for this study are ready to start the MPC phase. From within their private Docker containers, both sites download their respective secret shares and then participate in an MPC protocol to securely aggregate across all inputs and compute the study output (see next section).

3.2 Secure Query Evaluation

In this step, the computing partners evaluate the query using a secure computation (MPC) protocol, which is invoked by each site by running a locally deployed Docker image. The result of the computation is secret-shared amongst the compute partners, and they send these secret shares to the investigator or MRAIA, CAPriCORN's honest broker for conventional query evaluation. Figure 6 demonstrates this process. This computation simulates the presence of a completely trustworthy, incorruptible, honest broker by having the computing parties evaluate garbled circuits over the secret-shared inputs. In our running example, AC and NM pass encrypted messages to one another via an SSH tunnel between the two sites.

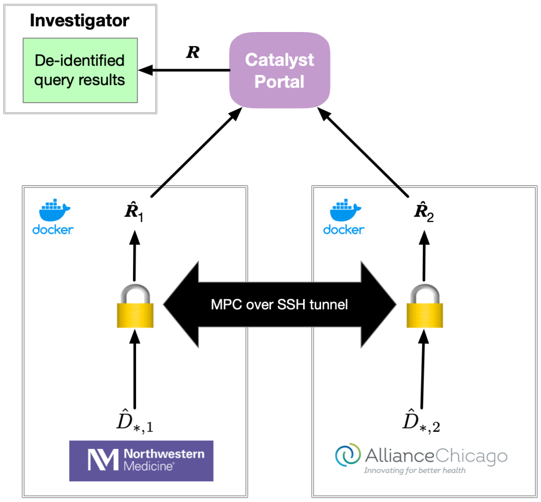


Figure 6: Computing partners evaluate study protocol over MPC and send results to the investigator.

As mentioned, each compute partner performs the secure computation protocol by running a virtual machine packaged in a Docker container. We deployed this on each computing party's infrastructure. Over the years, we have done this in a few ways, including 1) running the Docker container in local server space; 2) installing it on an analyst's workstation; and 3) serving it as an Azure Container Instance in a virtual private cloud. The security posture we used was also tailored to the best practices of each partner, where some placed it within a DMZ (demilitarized zone in network), and others were more comfortable running it behind the firewall but with rules that only admit connections from trusted sources (IP whitelisting).

The compute nodes work together to jointly compute the query answer, which is de-identified and further protected by insisting that any results with cell counts below a certain threshold (a threshold of 11 was used in our studies) are automatically removed. This produces the encrypted query result, *R*. Both compute nodes send their respective encrypted query result, $\hat{R}_1$ and $\hat{R}_2$, as shown in Figure 6.

The study Investigator can receive the experiment results either through the portal or by directly downloading the result shares from each site and reassembling them locally. In our experiments, we downloaded the shares to a study team member for reassembly, as shown In Figure 6, where the Investigator uses the output shares $\hat{R}_1$ and $\hat{R}_2$ to reassemble the query result *R*. Optionally, a data steward may verify that *R* satisfies cell size requirements and other criteria for release before returning them to the study Investigator.

4. Clinical Study Design

To demonstrate the clinical utility of MPC, we built upon our prior pilot study that was based on a query originally proposed by the Population Health Analytics Metrics and Evaluation (PHAME) center of the University of Illinois Chicago (Investigators Sanjib Basu, Sage Kim, and Wayne Giles). Namely, the original PHAME study (e.g., https://phame.uic.edu) aimed to test whether de-identified data of patients who had been diagnosed with various diseases -- diabetes, hypertension, breast cancer, cervical cancer, lung cancer, colorectal cancer, and prostate cancer -- could be aggregated at the zip code level and then used to compute the prevalence rates of these chronic diseases; and then to further estimate sub-group (age, sex, and race/ethnicity) rates by zip code. Our site lead for UIC, Dr. Howard Gordon, led the adaptation of the original PHAME query for use within the MPC pilot.

We chose to build from this existing study because we were able to leverage queries that were expertly validated against this real-world data. The investigators had previously used their domain knowledge to confirm sets of diagnosis codes, covariates of interest, and cohort inclusion criteria that would generate scientifically useful results for population health researchers. Recall that once the data are secret-shared, we have no way of cleaning or debugging it to prepare for query evaluation. Hence, working with Dr. Gordon and his team to extend their expertise to our secure queries was important for making this pilot possible.

Because these chronic health conditions are widespread in patient populations, we followed the original study's two-tiered approach for generating aggregated statistics, by considering two analyses ("queries") of interest:

(a) Population-level statistics: Estimate prevalence rates of various diseases by aggregating data across an entire patient population.
(b) Sampled prevalence-of-disease statistics: Sample amongst patients at each site to estimate the prevalence of disease.

For approach (b): Because the overall case count of our conditions of interest is estimated at over 10 million, we compute over a sample of 20% of cases, as in the original study. Our sampling was stratified by gender and age group. Our queries included all individuals over 18 years of age (when the query was run) and encounters between 2015 and 2021. All data partners took part in both of these queries.

4.1 Estimation of Disease Rates

Our data partners contributed their records to this study on one of two tracks:

1) Row-level partner: each record pertains to one patient.
2) Aggregate-only partner: each such partner site contributes (secret shares of) a single table that covers all patients in their sample.

Each team selected their track based on their comfort level for participating in the study. Contributing factors in this decision included the availability of a data use agreement, the availability of similar data handling precedents for each site in the past, and personnel availability for escorting the data through their local release process. A participant's choice to be a row-level data partner versus an aggregate-only partner is orthogonal to their role selection (data partner versus full-stack, as described in Section 3.1).

Row-level partner. For each patient in our sample, we added one Boolean flag indicating if they received a diagnosis code during the study period of each of the following: diabetes, hypertension, and cervical, breast, lung, and colorectal cancer.

The study's input SQL queries only generated a row for each patient selected in our stratified sample that had at least one diagnosis column set to true. Admittedly, this leaks the cardinality of patients who have at least one disease. We consulted with data stewards and other professionals in HIPAA's expert determination criteria and determined that this was consistent with the standard and practices of clinical research.

For each patient, we recorded their demographic information paired with their insurance status. We coded demographic data with a single character to make our computation more efficient and decoded it with a dictionary later. In this study, AllianceChicago, Northwestern Medicine, University of Illinois at Chicago, and Rush University Medical Center participated as row-level data partners. Our full workflow is in Figure 7, and we will review this shortly.

Aggregate-only partner. For data partners that preferred to contribute pre-aggregated data to our study, they provided a table that summarized their relevant chronic disease burden by pre-joining the tables described above. Endeavor Health and Lurie Children's Hospital participated in this study as aggregate-only data partners. All sites contributed a table for this, and we used this to evaluate different execution paths for running this query.

Input Data Sizes

	Input Rows		
Data Partner	**Rollup**	**Demographics**	**Diagnosis**
AllianceChicago	38,016	203,832	21,967
Rush University	45,736	423,852	75,686
UIC at Chicago	32,027	284,142	33,673
Endeavor Health	35,015	N/A	
Northwestern Medicine	83,118	1,275,051	170,440
Lurie Children’s Hospital	4,208	N/A	

Table 2: Row counts from each data partner in Catalyst pilot

Taken together, our MPC inputs had the following cardinalities:

Table	Total Row Count
Rollup	238,120
Demographic	2,186,877
Diagnosis	301,766

Table 3: Aggregate row counts for pilot

Note that the number of rows for `phame_diagnosis` is significantly lower than the ones for `phame_demographic`. The SQL statements that the analysts ran in their local data marts to generate study inputs only created `phame_diagnosis` rows for patients with at least one diagnosis of interest. In the following sections, we present and discuss results.

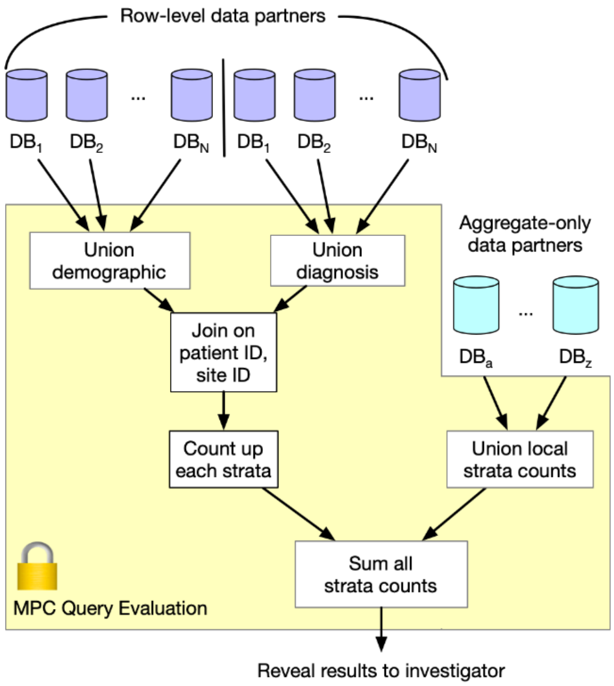
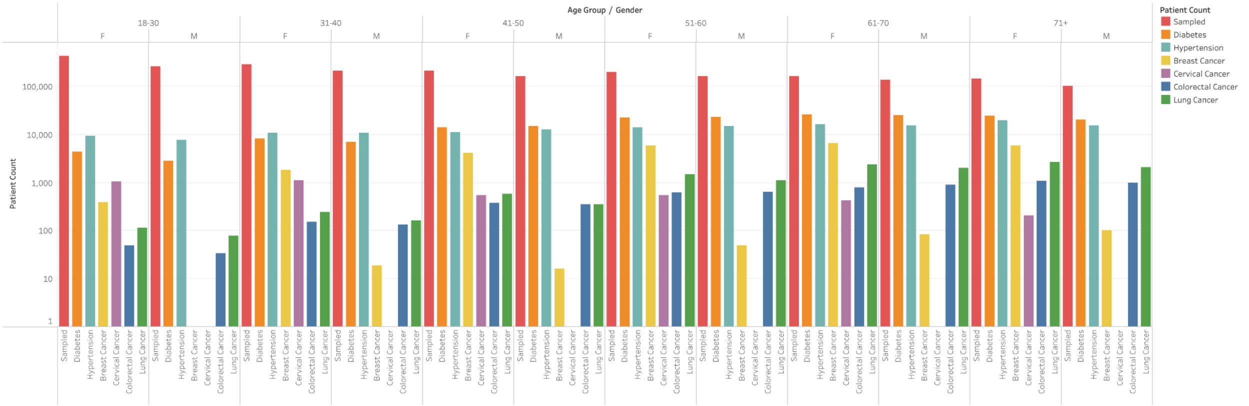


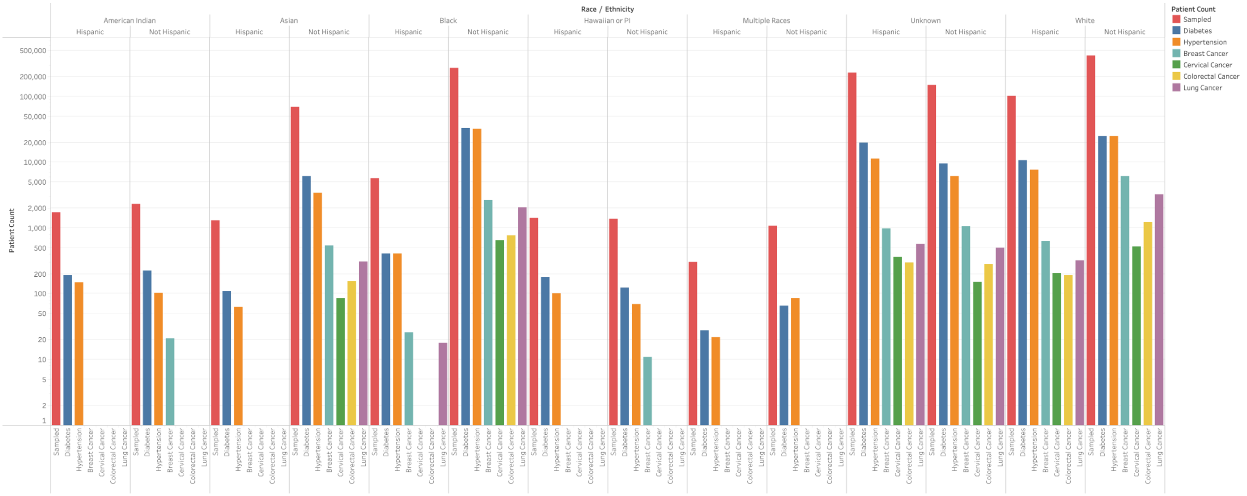
Figure 7: Query workflow for estimation of disease rates.

5. Results

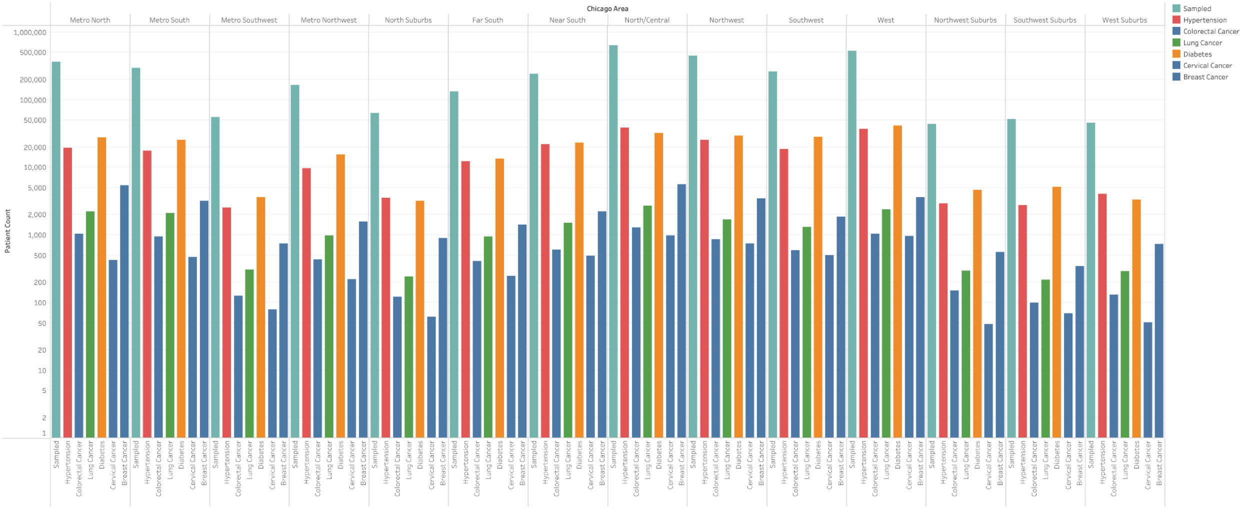
5.1 Age and Gender



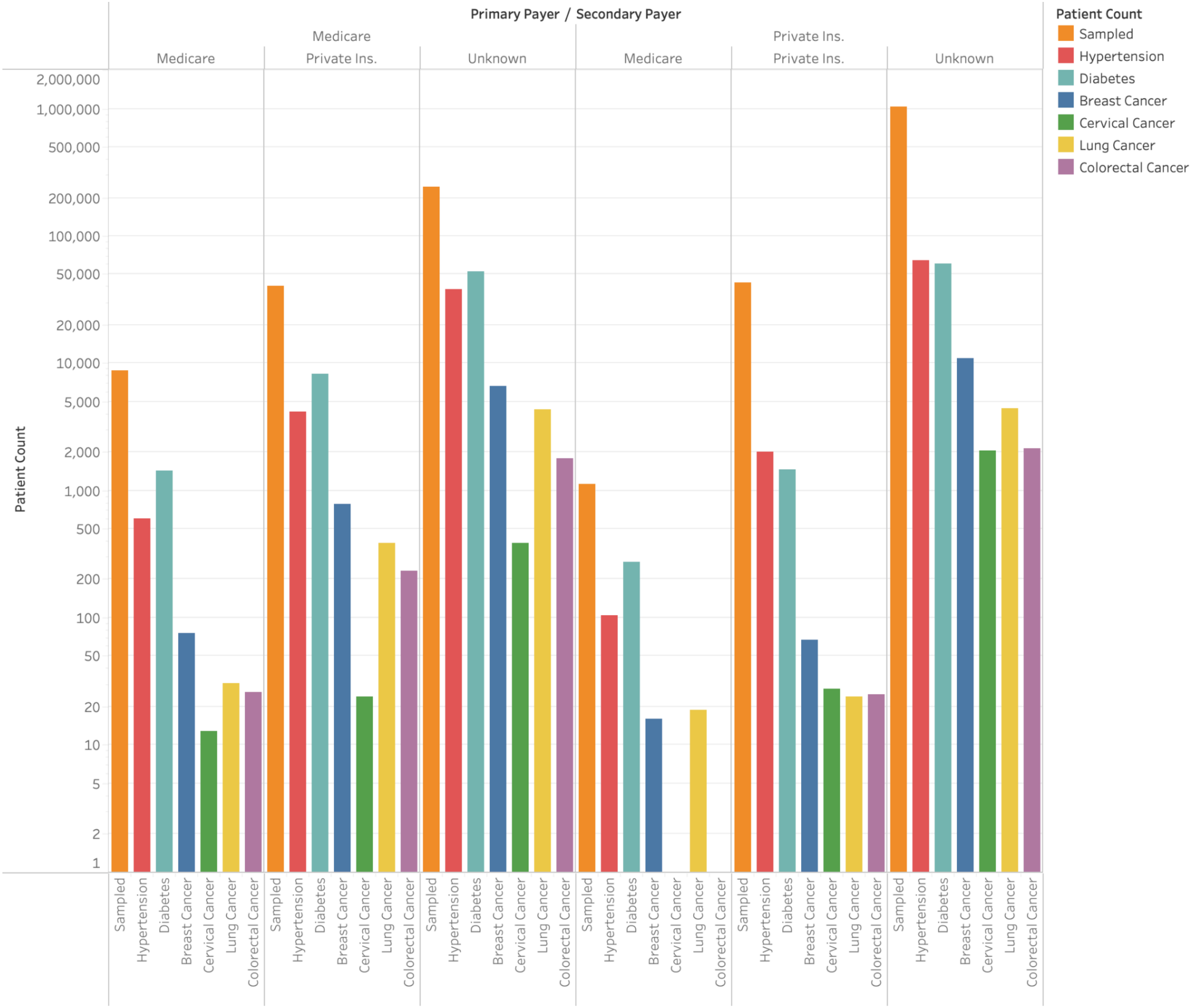
5.2 Race and Ethnicity



5.3 Geographic Distribution


Chicago Area
Metro North
Metro South
Metro Southwest
Metro Northwest
North Suburbs
Far South
Near South
North/Central
Northwest
Southwest
West
Northwest Suburbs
Southwest Suburbs
West Suburbs
Patient Count
1,000,000
500,000
200,000
100,000
50,000
20,000
10,000
5,000
2,000
1,000
500
200
100
50
20
10
5
2
1
Sampled
Hypertension
Colorectal Cancer
Lung Cancer
Diabetes
Cervical Cancer
Breast Cancer
🟩 Sampled
🟥 Hypertension
🟦 Colorectal Cancer
🟩 Lung Cancer
🟧 Diabetes
🟦 Cervical Cancer
🟦 Breast Cancer

5.4 Payer



6. Discussion

In this report, we describe the design, implementation, and results for a real-world, large-scale deployment of secure multi-party computation (MPC) across six healthcare institutions covering over 12 million individuals. To our knowledge this is the first and largest deployment of MPC within healthcare. Our learnings were often as non-technical (e.g., organizational or administrative) as they were technical. Key learnings included:

- **Educating stakeholders on MPC eased acceptance and adoption**: Time spent educating decision makers and local technology leaders helped to build comfort with MPC technology and engage key stakeholders throughout the back-and-forth of deployment.

- **Variation in local software availability**: Medical software varied in terms of which versions were available at each site. This required writing additional javascript plugins for compatibility with older browsers.
- **Value in the use of cloud resources**: The compute partner sites all had cloud-based resources, and by making our service run in the cloud, we avoided many of the on-prem idiosyncrasies.
- **Manual steps in the workflow are more helpful than harmful**: Creating csv snapshots can be done by users rather than an automatic database shim. This allowed users to better understand what is going into the system and also helped catch otherwise hard-to-detect malformed data. It also enabled users to have more transparency into the underlying system and made it easier to establish trust in comparison to needing direct access to a production database.

One weakness in our pipeline that this exercise exposed is identifying the right mechanism with which to release results to the Investigator. In our configuration, they ran a binary program that takes in the secret shares from the computing parties and the schema (column definition) of the output table, XORs together the shares, and outputs the results to CSV. We considered building an Investigator dashboard, similar to that of the contributors, where the shares are recombined in the browser. This would depend on the operational policies around how this system would be deployed in a study.

In practice, many users are not on HIPAA-compliant/PHI-approved machines when they are working. This makes sense because minimizing where we handle the data reduces the surface area for a potential data breach. On the other hand, major cloud vendors offer PHI-ready counterparts to their standard offerings (e.g., storage and compute). In future work, we anticipate developing a simple Function as a Service (FaaS) solution, such as AWS's Lambda or Azure Functions, to perform this workflow and output the results to an appropriate in-cloud storage location that the Investigator accesses from within the portal.

6. Conclusion

The Catalyst project demonstrated that MPC works in real-world settings and is provably adoptable in healthcare. Computationally, MPC can scale to clinically relevant population health queries over a large, diverse metropolitan area.

Significant challenges still remain. Because of the “black-box” nature of MPC, future work to exhibit the correctness of resulting analyses is needed. This may involve executing MPC simultaneously with an “in-the-clear” query on less sensitive data, and comparing results to ensure consistency, so that when we use MPC for more sensitive data, we can be assured it is producing the correct results. An additional challenge is to streamline the deployment of MPC, which would reduce the amount of local expertise and resources that are required to participate, thus allowing for more widespread adoption. For example, removing some of the more time-consuming or human-dependent steps.

Encouragingly, our Catalyst deployment, with all the components of development, testing, education, adaptation, and end-to-end execution, was completed within a year. As one of the sites in our study observed: “Catalyst makes the data sharing process easy to understand and follow. It makes us feel more comfortable with sharing protected health information.”

With Catalyst installed at several sites, our future work is focused on exploring additional use cases for high-value, high-sensitivity data/insights; e.g., benchmarking, rare or sensitive conditions, or work with inaccessible data whether it be health data, student data, income data, or population data.