DATA EXCHANGE DEMO

Share data while retaining control and confidentiality of your data
Gains and difficulties of sharing confidential data

Access to non-public data.

Potential new research and collaborations.

More work to manage confidential data.

Possible to gain new insights.

Risks on privacy and security.

Additional work without direct return on investments (ROI).

Gain is usually with the data requester, burden is with the data provider.
Willingness to share data

Return on Investment (ROI) is determined by the balance between the effort it takes to share data, and the gain received by sharing data.

Trust is determined by the balance between the risks (due to privacy or competition), and the control (due to verification and security) of sharing and usage of data.

**ROI**

<table>
<thead>
<tr>
<th>Gains</th>
<th>Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

+ 

**Trust**

<table>
<thead>
<tr>
<th>Control</th>
<th>Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Return on investment

Trust
Type of Data Owners

- Data aggregators
  - Health care (Palga, NZa)
  - Social-economic (CBS, municipalities)

- Hospitals + medical institutions
  - Hospital (AMC, vuMC, St. Antonius)
  - Insurance companies (Zilveren Kruis)

- Onderzoekers + universiteiten
  - Universities (Twente, Wageningen, Groningen)
  - Researchers

- Bedrijven
  - Friesland-Campina, Elsevier

Privacy sensitive

Competitive data
Methods to Ease Data Sharing

**Agreements**
- Stipulation of what can/cannot be done
- Signing of contract or NDA
- Dispute resolution process

**Registration**
- Authentication
- Verification of credential
- Reputation score
- Policy framework
- Audit trails

**Pseudonymization**
- Filtering (on records)
- Pruning (on properties)
- Aggregation (combining records)
- Make coarse grained buckets
- Slight alteration of data
- One-way hashing
- One-time identifiers

**Data Vault**
- Data source retains control
- Delegate permissions
- No central data lake
- Data marketplace

**Secure Containers**
- Bring algorithm to data
- Trusted third party
- Share output instead of data

**Secure Computing**
- Secure multi-party computation
- Homomorphic encryption
- Garbled Circuits
- Zero-knowledge proof
Example: Find the average income

Run #1
- 21 people
- Algorithm verified
- Outcome guaranteed not to be traceable to individual people

Run #2
- 22 people (same 21 and 1 other)
- Algorithm verified
- Outcome guaranteed not to be traceable to individual people

Even if individual runs are fine, combining two runs may reveal confidential data
## Data Exchange

<table>
<thead>
<tr>
<th>VISION</th>
<th>Realize a platform where data can easily be shared, while retaining control and confidentiality of the data</th>
</tr>
</thead>
<tbody>
<tr>
<td>TARGET GROUP</td>
<td>Data providers with confidential data. E.g. • Companies; • Academic hospitals. Researchers who like to use data from other organizations for a specific purpose.</td>
</tr>
<tr>
<td>NEEDS</td>
<td>Data providers like to share data, while • retain control who can use the data for what purpose; • adhere to legal limitations of processing data. Data consumers (researchers) don’t want to be limited to public datasets.</td>
</tr>
<tr>
<td>PRODUCT</td>
<td>Proof of concept (demonstration). Secure environment at trusted third party. Performs calculations on data on behalf of a researcher, with explicit consent from the data provider.</td>
</tr>
<tr>
<td>BUSINESS GOALS</td>
<td>Facilitate open science Researchers make more use of data sources. Provide a easy-to-use and trusted solution for both parties, data providers and researchers</td>
</tr>
</tbody>
</table>
Collaborating without direct Sharing Data

- **Data Provider**
- **Researcher (Algorithm Provider)**
- **Trusted Third Party**

Data → Trusted Third Party → Secure container

- **Curation of result**
- Code + Data

Result → Result
Data provider shares data with trusted third party;  
Researcher shares algorithm with trusted third party;

Researcher makes request to data provider;

Data provider verifies requester and algorithm;  
... and selects data set(s);

Trusted third party creates secure container;  
... mounts algorithm and data set;  
... executes algorithm;

Data provider verifies output and algorithm behaviour;

Once released, the researcher receives the output.
## Permission Models

### One-off permission

The data provider permits a researcher to **run** a specific algorithm **once** on a specific dataset.

The permission can be revoked at any time. Example use cases:
- the data provider trust the researcher to always write benevolent code
- the researcher wants to tweak the algorithm, and run it on a sample dataset every time.

### Trust a researcher

The data provider permits a researcher to **run any algorithm on a specific dataset**.

The permission can be revoked at any time. Example use cases:
- the data provider trust the researcher to always write benevolent code
- the researcher wants to tweak the algorithm, and run it on a sample dataset every time.

### Run on a data stream

The data provider permits a researcher to **run a specific algorithm on any data set in a selected folder**. Every time a new dataset is added to the folder, the algorithm is automatically run.

The permission can be revoked at any time, but is also automatically revoked as soon as a change to the shared algorithm is detected.

Currently supported permission models
Technical Implementation of the prototype

External integrations

- Secure container
- Secure container
- Secure container
- Secure container

Data Exchange

- Data provider account
- Data Exchange account
- Researcher account

Data & Algorithm Storage

Secure container

WebDAV file copy

Internal Components

- Frontend (Sapper)
  - Backend (Django)
    - Database (PostgreSQL)
      - Message Queue (RabbitMQ)
        - Backend Listener (Django)
          - Tasker (Scala)
            - File Manager (Scala)
Data is shared with the Data exchange
Algorithm is shared with the Data exchange by researcher
Researcher makes a request to the data provider
Data provider reviews request and selects dataset
Trusted Third Party runs algorithm on dataset
Data provider reviews output
<table>
<thead>
<tr>
<th>Data Owner</th>
<th>Algorithm Name</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:freek.dijkstra@surfsara.nl">freek.dijkstra@surfsara.nl</a></td>
<td>calculate_sum.py</td>
<td>30717</td>
</tr>
</tbody>
</table>

**Permission Type**
- one time permission

**Permission Information**
The selected algorithm will be ran on the selected dataset of the data owner exactly once.

**Algorithm Dependencies**
- sys

**Algorithm Length**
- Lines: 22, Words: 44, Characters: 522

**Choose dataset**
- random_numbers.txt

Researcher can see released output
Data provider can at any time withdraw permissions
Related Projects

ODISSEI Secure Supercomputer (OSSC)
- In production
- Processes CBS micro-data on Cartesius
- Does pseudonymization as well

AMdEX
- Collaboration of interested parties
- Initiated by Amsterdam Economic Board
- Goal is to build an infrastructure for multiple Data Marketplaces
COLLABORATION WITHOUT SHARING DATA

Freek Dijkstra
Freek.Dijkstra@surfsara.nl
www.surf.nl

Driving innovation together

This presentation is available under the creative commons attribution 4.0 license