Breaking Data Silos
the Gen3 platform for creating data commons

Presented by
Chris Meyer, PhD

Center for Translational Data Science,
University of Chicago
Presentation Outline

● The Data Silo Problem
  ○ What Are Data Silos?
  ○ Why Do They Exist?

● How to Open Data Silos and Prevent Their Creation
  ○ The Data Commons Paradigm: Making Data FAIR
  ○ The Gen3 Platform and How It Breaks Data Silos

● Demonstration of the Gen3 Platform
  ○ Exploration of Windmill, Gen3’s Web-based Data Portal UI
  ○ Introduction to the Gen3 Workspace: Using the Gen3 SDK in a Jupyter Notebook
The Data Silo Problem

the barriers to sharing and re-analysis of data
What are Data Silos?

- A **data silo** is an isolated data management system that is incapable of interoperating with other similar systems.
- Data stored in silos are hidden or inaccessible to analysts outside of the contributing department or organization.

Photo credit: Ina Kratzsch
Common Causes of Data Silos

- **Silo mentality**: refusal to open data up to competitors, even to other departments within an institution, or lack of effort due to different goals.

- **Lacking Data Model or Quality**: data are not well-described in the system, which prevents new users from understanding how to use it properly.

- **System architecture incompatibility**: data are stored on a system that does not provide open APIs, requires a special login, or has other technical limitations and incompatibilities.
The Promise of Data Re-use

Why should the scientific community be concerned about the sharing and re-analysis of data?

- The scientific method promotes the concept of study reproducibility.
- There are undiscovered insights in the data from a single study that re-analysis using new methods could reveal.
- Combining data from multiple, smaller studies in a cross-project or meta-analysis could reveal insights that would not be discoverable through analysis of individual studies (ML across TCGA is a great example).
Breaking Data Silos with Gen3

*open-source software for creating data commons*
A **data commons** is cyberinfrastructure that co-locates data storage, data management, and computing infrastructure with commonly used tools for analyzing and sharing data to create an interoperable resource for the research community.

The Case for Data Commons

- Data commons provide a secure platform for integrated data management and analysis to the entire scientific research community.
- The goal of a data commons is to deliver new datasets, new analytical methods and pipelines, and new apps for exploring and analyzing data through collaboration.
- New analyses can be performed and results can be hosted and shared all within the secure data commons environment to promote reproducibility and accelerate discoveries.
What is the “Gen3” platform?

- **Gen3** is an open-source software stack for creating data commons.
- The software is comprised of a set of microservices that provide the basic functions for creating and operating a data commons:
  - **User authentication and authorization** to secure data access and analysis (*Fence*, auth service).
  - **Metadata import and indexing** using permanent digital IDs (UUIDs) (*Sheepdog*, metadata service).
  - **Data file import and indexing** using permanent digital IDs (GUIDs) (*Indexd*, file service).
  - **Query of metadata and files** using graphQL (*Peregrine*, query service).
  - **A web-based user interface** for data management, exploration, and analysis (*Windmill*, data portal service).
- **Gen3** is the third generation of this technology, which runs microservices in containers and utilizes cloud automation (*Kubernetes*).
- **Gen3** is cloud-agnostic, so files in a data commons are assigned a unique, permanent ID, but can be stored in and moved between any cloud location (*Amazon S3*, *Google GCP*, private FTP, intranet servers, etc.).
Gen3 uses a graph-like relational data model to describe the metadata associated with data files and any other information required to understand and replicate the scientific study, e.g.:

- Sample storage / processing info
- Patient demographics / medical history
- Environmental / wearable sensor data
- Omics data and associated metadata
- Processing pipelines and parameters
- Associated authors / publications

The data model evolves and typically follows a widely accepted vocabulary in the field (e.g., ICD codes, OMOP, NCIt).
Gen3 provides a lightweight, minimal set of core framework services for operating data commons, and it does so by exposing open APIs that apps can be developed over.
An **API** (application programming interface) is a communication protocol between client and server for building client-side apps.

- APIs have various “endpoints” (URLs) for performing their various functions.
- A client sends a request to an API endpoint to retrieve specific types of information or to perform a specific function.
- An example is the Facebook API, which allows 3rd parties to develop apps over their APIs that request information about user profiles, friends, and events.
- Gen3 provides APIs relevant to biomedical, environmental and translational data science.

**Examples of requests sent to Gen3 APIs include:**

- List the data projects you have access to and download credentials for data access (auth service).
- Submit metadata to a project in the data commons (metadata service).
- Create a synthetic cohort by querying patient metadata across all projects (query service).
- Upload genome sequences from new tumor samples (file service) and link them to their patients’ medical history metadata (metadata service).
Why use Gen3? It promotes FAIR data

- Gen3 was designed to make it easy for research communities to create FAIR data commons at minimal cost and technical barrier.
- The FAIR principles promote the idea that data should be:
  - **Findable**: Gen3’s metadata and query services index data with unique and persistent identifiers and expose it to queries that run across data resources.
  - **Accessible**: Gen3’s authentication and authorization service provides secure access to data and the portal service provides a web-based user interface for exploring data projects and launching analysis workspaces.
  - **Interoperable**: Gen3 was designed for interoperability, providing open APIs that communicate with clients using common protocols and formats.
  - **Reusable**: Gen3 enforces the use of a data model, which requires data contributors to adopt a common vocabulary when providing patient medical history, data files and their associated metadata.

https://www.go-fair.org/fair-principles/
Demonstration of Gen3

use cases for Gen3 data commons
Windmill provides a UI for all the API functions a commons user would want:

- User login and credentials management UI.
- A project-based data upload / download UI.
- An interactive data exploration and cohort building application.
- An interactive data dictionary viewer.
- An interactive query building interface (GraphiQL) with autocomplete and built-in documentation.
- An integrated workspace with pre-built VM images that support JupyterHub and R Studio.
The Gen3 SDK: Developing Tools Over Gen3 APIs

- **The Gen3 SDK** is an open-source software development kit (SDK) that provides tools in the Python and R programming languages for interacting with Gen3 APIs.
- For example, the Python SDK uses the `requests` package to hit Gen3 APIs.
  - Gen3Submission is a class of functions for exporting/importing/querying data using the Sheepdog and Peregrine APIs.
  - Gen3Auth is a class of functions for use authentication and getting pre-signed URLs for data file upload/download.
- The code lives in public GitHub repos:
  - [https://github.com/uc-cdis/gen3sdk-python/](https://github.com/uc-cdis/gen3sdk-python/)
  - [https://github.com/uc-cdis/gen3sdk-R](https://github.com/uc-cdis/gen3sdk-R)
Now I will demonstrate use of the Gen3 SDK using the BloodPAC Data Commons, which is a data commons for liquid biopsy data.

The BPACDC is a Cancer Moonshot project launched in 2016 and is a collaborative effort by industry, government, and academic partners and stands for “Blood Profiling Atlas in Cancer” (bloodpac.org).

In this demonstration I will:

1. Login to the data portal (data.bloodpac.org).
2. Spin-up a Workspace VM within the secure Virtual Private Cloud (data.bloodpac.org/workspace)
3. Launch a Jupyter Notebook to interactively run code over the BloodPAC APIs for a basic exploratory data analysis.
4. Demonstrate similar functions in the Windmill data portal UI.
The Gen3 Community

democratizing translational data science
The University of Chicago operates several production data commons for clients like the NCI, NIH, VA, and others.

- Go to stats.gen3.org to see some examples.

Gen3 is entirely open-source and available to the community.

- Code and documentation for all microservices are in public GitHub repositories: https://github.com/uc-cdis/
- Data commons operators outside of UChicago’s development team are adapting and customizing Gen3 for their own purposes.
- These 3rd party developers can clone the Gen3 repo and make changes or submit pull requests (PRs) that our developers can review and implement.
Learn More

- [github.com/uc-cdis](https://github.com/uc-cdis)
- [gen3.org](https://gen3.org)
- Gen3 Community on Slack (email us to join)
- [support@datacommons.io / cgmeyer@uchicago.edu](mailto:support@datacommons.io / cgmeyer@uchicago.edu)
- [ctds.uchicago.edu](https://ctds.uchicago.edu)
- Paid Summer 2020 Internships and full-time positions are
Questions?