

Training models on atlas-scale single-cell datasets

scverse Conference | September 12, 2024

Maximilian Lombardo

Sr. Product Applications
Scientist, CZI



Ryan Williams

SOMA Software
Engineer, TileDB



Spencer Seale

Solutions Architect, TileDB

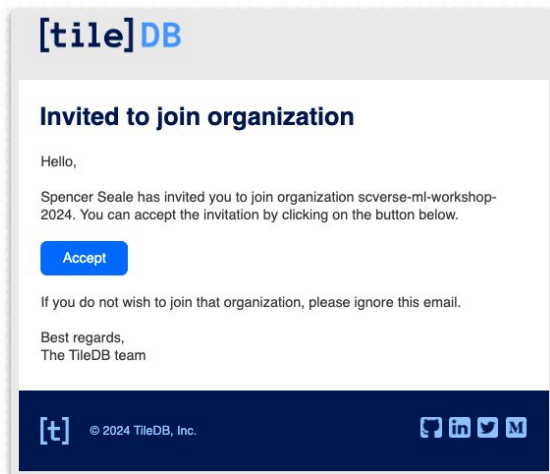


The workshop in a nutshell

WE WILL COVER

- What is CellxGene Census?
- What are TileDB and TileDB-SOMA?
- Understanding the Census and its data.
- Using the Census to:
 - Gain a high-level understanding of its contents.
 - Access and slicing data efficiently.
 - Performing computations efficiently.
 - Export data to single-cell toolkit.
- Utilizing Census PyTorch loaders for scalable modelling.
 - Training a classifier for Cell Type Annotation (immune cells)
- Cell Type Annotation via similarity search (vector search) for immune cells

This will be a hands-on workshop



If you *didn't* receive an invitation to join TileDB Cloud via email, **email spencer.seale@tiledb.com now.**

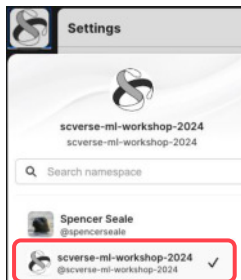
If you *did* receive an invitation, **accept it, and follow the instructions to create a TileDB Cloud account.**

Alternatively, you can passively along, or view the notebook at [TileDB-Inc/scverse-ml-workshop-2024](https://github.com/TileDB-Inc/scverse-ml-workshop-2024) on GitHub.

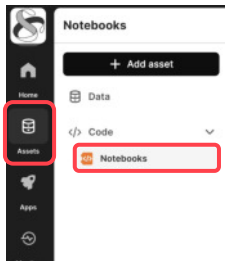
Questions? Message us in Zulip!

1 Login at cloud.tiledb.com

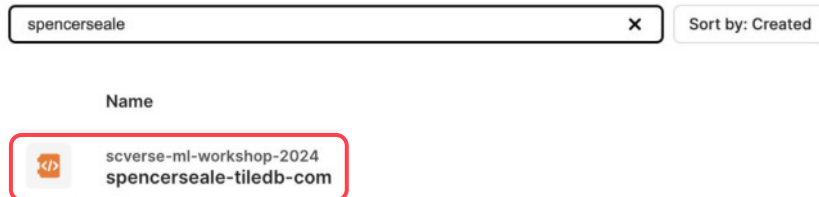
2 Switch to workshop Namespace (top-left)



3 Go to "Assets" & "Notebooks" (left)

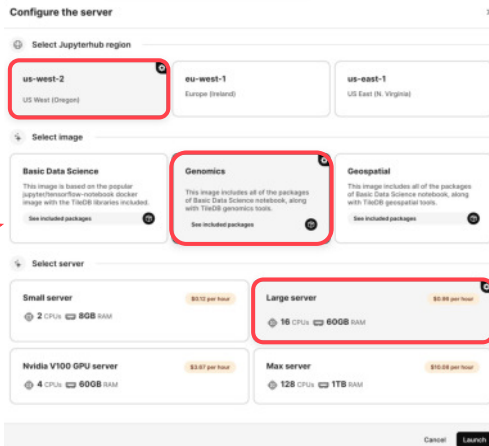


4 Search by your email username and select your notebook (top-right)

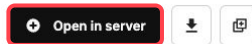


5 Launch your notebook with the configurations:

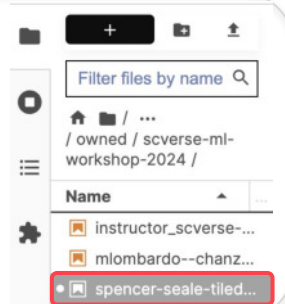
- **us-west-2**
- **Genomics**
- **Large server**



6 Display server



7 Select notebook



CELLxGENE Discover Census

<https://cellxgene.cziscience.com/>

The screenshot displays the Cell Ranger web application interface. At the top, there is a navigation bar with links for 'Collections', 'Datasets', 'Data Explorer', and 'Help & Documentation'. Below this, the main content area is divided into several sections. On the left, there are two panels: 'Standard Categories' and 'Author Categories', both with expandable menus. The 'Standard Categories' panel shows a list of categories including 'anxiety', 'cell_type', 'development_stage', 'disease', 'self_reported_ethnicity', and 'sex'. The 'Author Categories' panel shows a list of categories including 'cell_embedding_class', 'time_annotation', and 'Continuum'. The central part of the interface features a large t-SNE plot showing a distribution of cell clusters. The plot is color-coded by cluster, with a legend on the left showing 'Standard Categories' and 'Author Categories'. The top of the plot area displays 'sample: 101115 of 101115 cells' and '0 cells, 0 cells'. The right sidebar contains a 'Genes' section with a search bar and a 'Gene Sets' section with a 'Create new' button. The bottom of the plot area shows a 'Tabula Sapiens' dataset with a 'Tabula Sapiens - Blood' view.

[illegible]

Cardiac Muscle Cell hsa04151

Search all types of Enzymes

Marker Genes

Gene	Marker Score (Q)	Expression Score (Q)	% of Cells
ITIN	8.63	4.66	88.8
ITIN2	3.57	5.03	96.1
ITIN1	3.33	2.95	95.8

UNIQUE CELLS

90M

Human 55M
Mouse 30M

DATASETS

1486

DISEASES
132

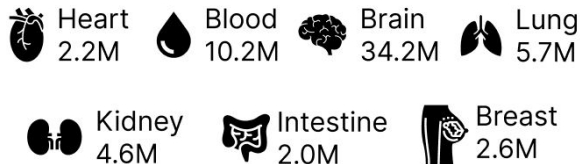
CELL TYPES

907

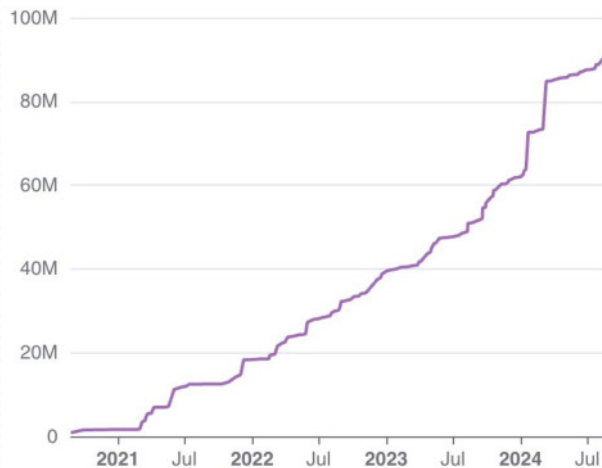
ASSAYS

- 10x RNA-seq
- sci-RNA-seq
- microwell-seq
- Drop-seq
- Seq-Well
- Smart-seq2 and v4
- BD Rhapsody
- GEXSCOPE
- MARS-seq
- Visium 10x

TOP TISSUES



CONSTANT RATE OF DATA INGESTION



All data at CZI CELLxGENE Discover is curated for data reuse and integration





<https://cellxgene.cziscience.com/>

GENE EXPRESSION

Normalized data

Raw counts

FEATURE METADATA

 Name   Type  ID



SCHEMA



UNIVERSALLY AVAILABLE CELL METADATA

 Species

 Tissue

 Age

 Sex

 Ethnicity

 Assay

 Cell type

 Suspension

 Donor

 Disease

<https://github.com/chanzuckerberg/single-cell-curation/blob/main/schema/5.1.0/schema.md>

What is Census?

Built from >800 datasets Census is a **data object + API**. It gives efficient access to the largest aggregation of standardized single-cell data ready for analysis and modelling at scale.

<https://chanzuckerberg.github.io/cellxgene-census/>

Single-cell RNA
datasets

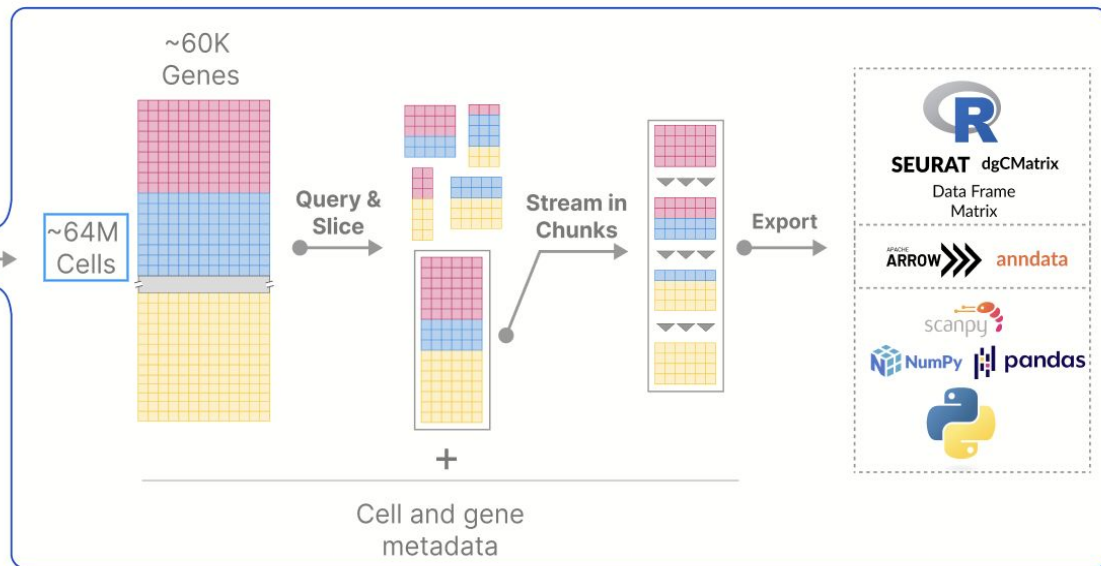


800+
H5ADs { : : }



Census

Powered by [tile]DB



Census stack

Census is built atop **TileDB-SOMA**, which is built atop **SOMA** and **TileDB**

<https://github.com/chanzuckerberg/cellxgene-census>

<https://github.com/single-cell-data/TileDB-SOMA>

<https://github.com/single-cell-data/SOMA>

<https://github.com/TileDB-Inc/TileDB>

For this workshop we will use **TileDB-Cloud** to skip the installation process

<https://cloud.tiledb.com/>

Coming to Census soon

Newly released **Census Similarity Search**
Now available to use!

▶ [Similarity Search Documentation](#)

SOMA (data model) – **TileDB** (database)

What is SOMA?

(*Stack Of Matrices, Annotated*)

Data Model

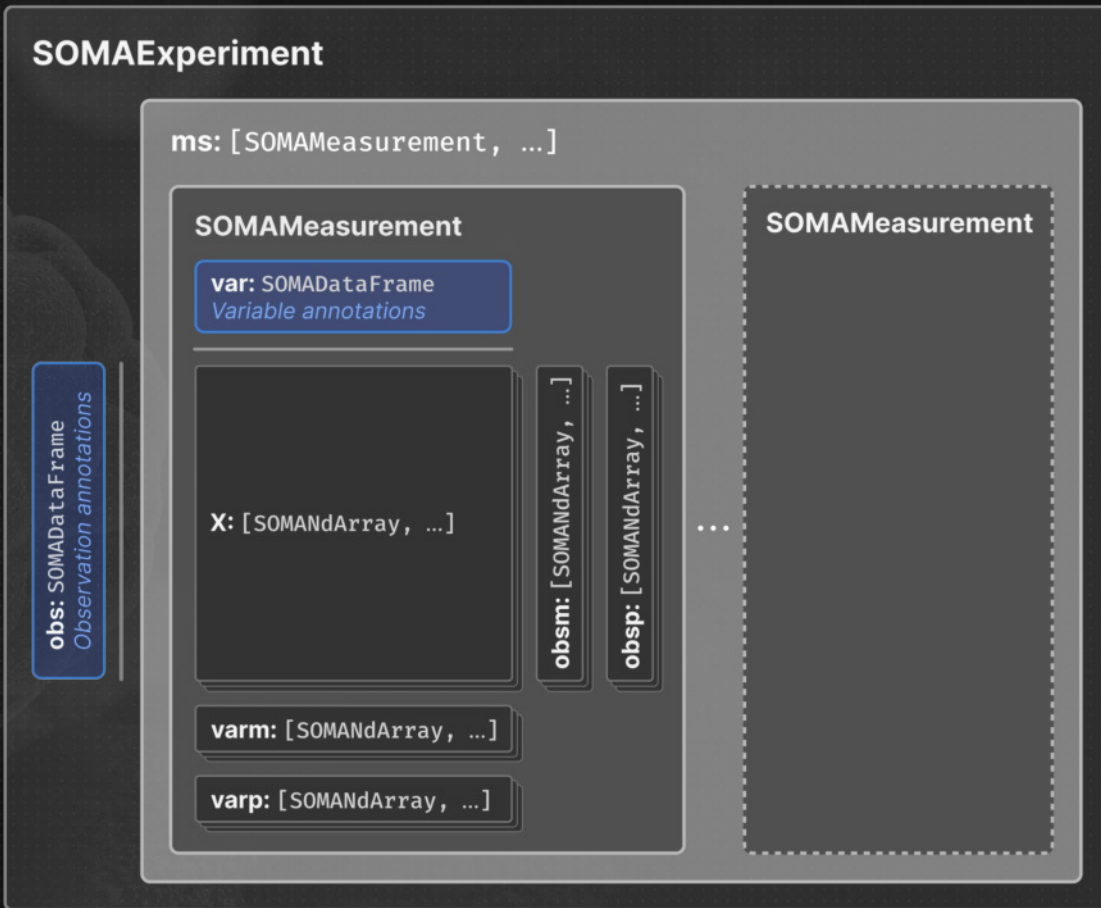
A language-agnostic data model for representing collections of annotated matrices and derived results on disk.

API Specification

A language-agnostic API specification for interacting with the data model.



SOMA Data Model



Goals for SOMA



Open source

- ✓ specification & reference implementations



Language-independent

- ✓ support both R & Python
- ✓ option to add more languages in the future

Interoperable with popular single-cell analysis frameworks

- ✓ AnnData/ScanPy (Python)
- ✓ Seurat (R)
- ✓ Bioconductor (R)



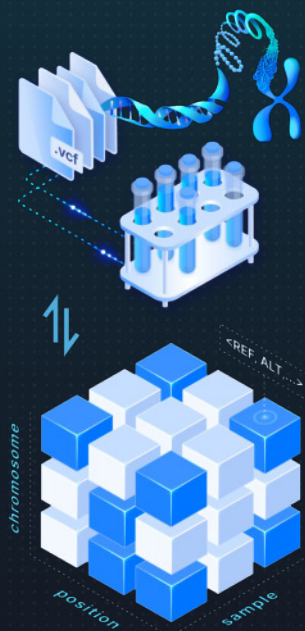
Designed for atlas-scale datasets

- ✓ support for cloud object stores
- ✓ out-of-core reads
- ✓ efficient querying

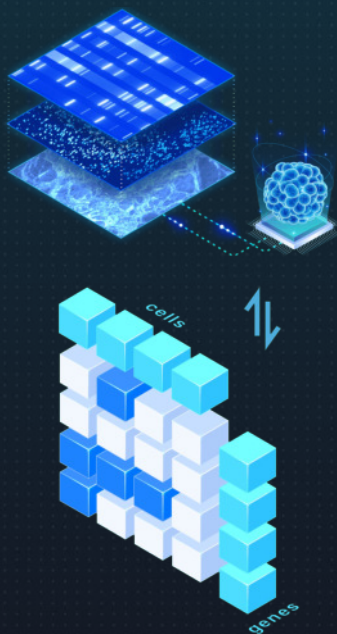


TileDB is a multi-modal database that morphs for any application

Population Genomics



Single-cell Biology



Biomedical Imaging



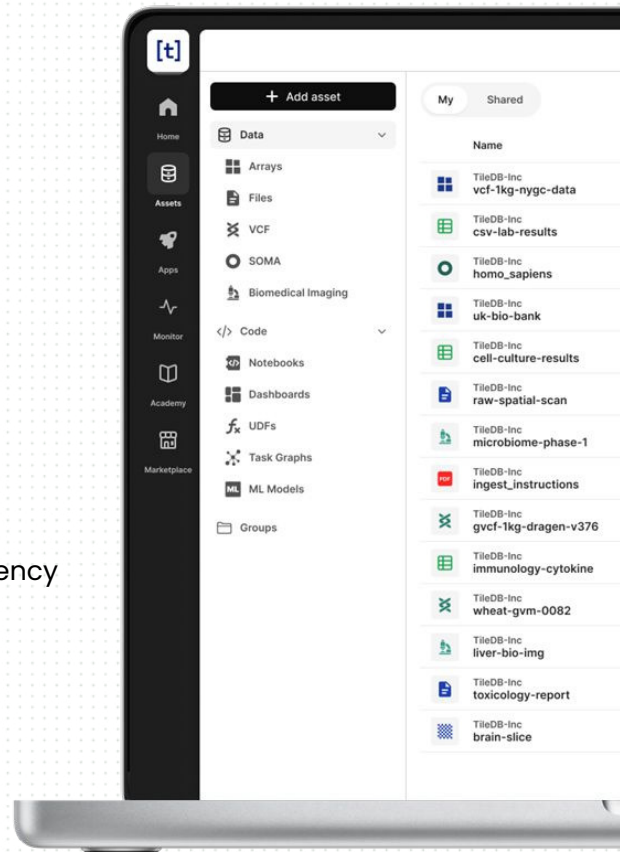
Why TileDB?

KEY TECHNICAL FEATURES

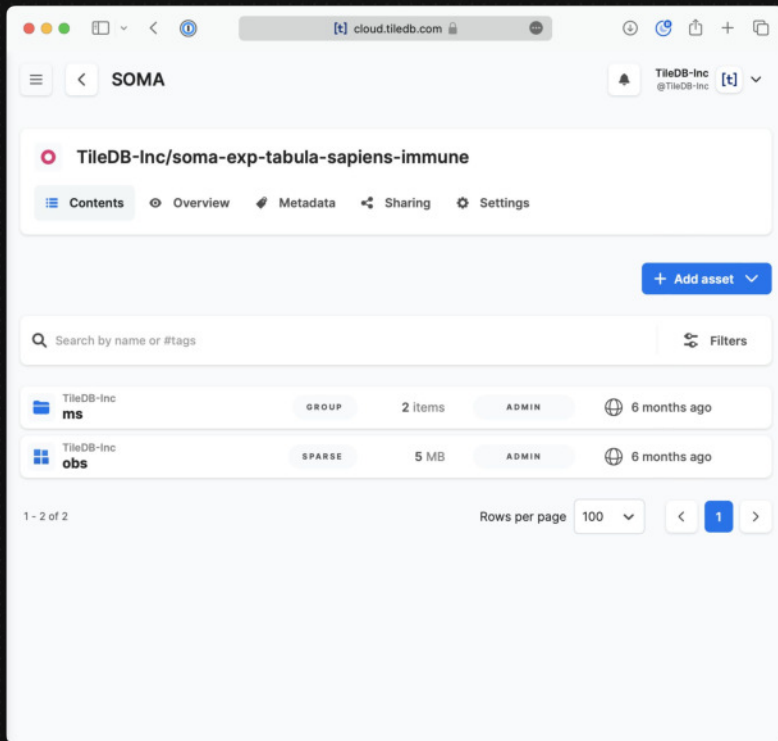
- ✓ **Universal Format** Sparse and dense ND arrays, key-value data, data frames
- ✓ **Flexible Indexing** Supports ints, floats, dates, and strings for versatile queries
- ✓ **Scalable Design** Uses tiling for selective memory loading during queries
- ✓ **Columnar** Enables efficient compression and selective attribute queries
- ✓ **Cloud-Native** Seamlessly works with local and cloud-based storage
- ✓ **High Performance** Fully parallelized I/O operations, multi-reader/writer concurrency
- ✓ **Cross-Platform** APIs for Python, R, Java, and many other languages

SUMMARY

TileDB offers **unparalleled flexibility** and **scalability**, enabling researchers to **handle diverse and growing datasets** with ease.



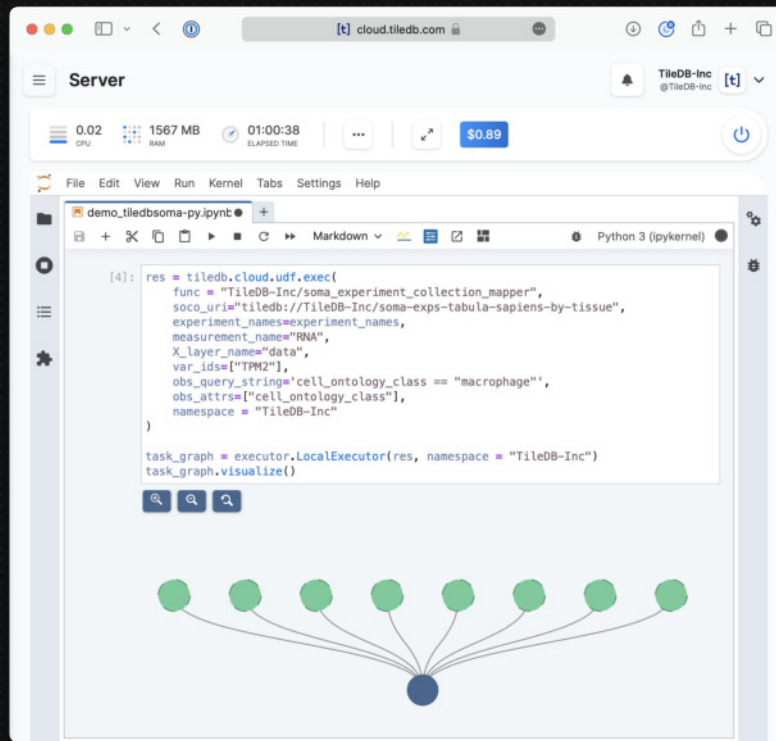
Scaling SOMA with TileDB Cloud



The screenshot shows the TileDB Cloud interface for a workspace named "SOMA". The top navigation bar includes a hamburger menu, the workspace name "SOMA", and a user profile for "TileDB-Inc". Below the navigation bar, there is a section for the workspace "TileDB-Inc/soma-exp-tabula-sapiens-immune" with tabs for "Contents", "Overview", "Metadata", "Sharing", and "Settings". A blue "Add asset" button is visible. A search bar with the placeholder "Search by name or #tags" and a "Filters" button are present. Below the search bar, there is a table listing assets:

Asset Name	Group	Items	Admin	Created
TileDB-Inc ms	GROUP	2 items	ADMIN	6 months ago
TileDB-Inc obs	SPARSE	5 MB	ADMIN	6 months ago

At the bottom, there is a pagination bar showing "1 - 2 of 2" and a "Rows per page" dropdown set to "100".



The screenshot shows the TileDB Cloud interface for a workspace named "Server". The top navigation bar includes a hamburger menu, the workspace name "Server", and a user profile for "TileDB-Inc". Below the navigation bar, there is a section for the workspace "Server" with a status bar showing "0.02 CPU", "1567 MB RAM", "01:00:38 ELAPSED TIME", and a cost of "\$0.89". A power button is visible. Below the status bar, there is a Jupyter Notebook interface with a file explorer showing "demo_tiledbsoma-py.ipynb". The notebook code is as follows:

```
[4]: res = tiledb.cloud.udf.exec(
    func = "TileDB-Inc/soma_experiment_collection_mapper",
    soco_uri="tiledb://TileDB-Inc/soma-exps-tabula-sapiens-by-tissue",
    experiment_names=experiment_names,
    measurement_name="RNA",
    X_layer_name="data",
    var_ids=["TPM2"],
    obs_query_string='cell_ontology_class == "macrophage"',
    obs_attrs=["cell_ontology_class"],
    namespace = "TileDB-Inc"
)

task_graph = executor.LocalExecutor(res, namespace = "TileDB-Inc")
task_graph.visualize()
```

Below the code, there is a visualization of a task graph showing a central blue node connected to seven green nodes.



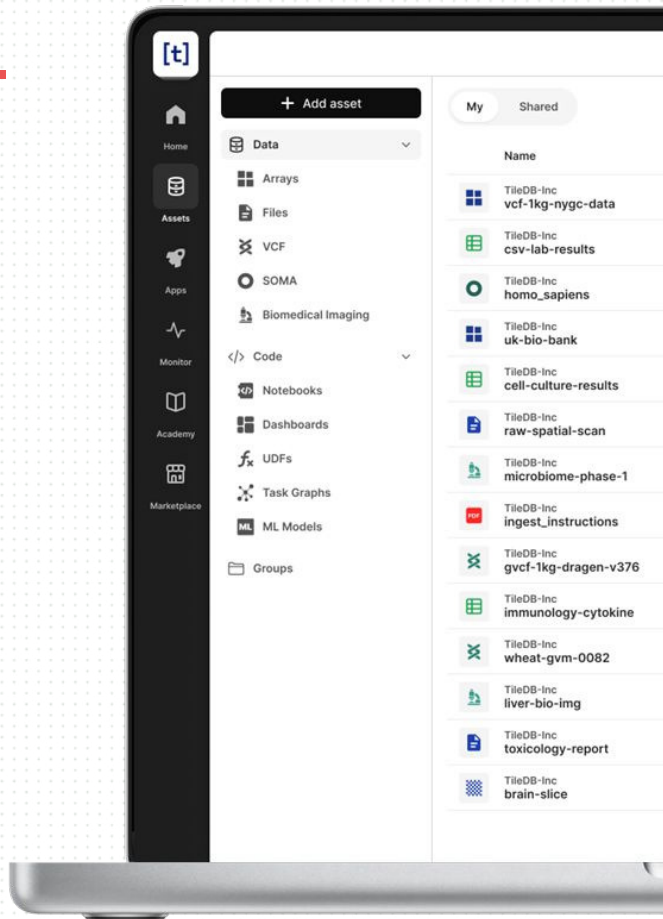
Why TileDB Cloud?

Commercial product for pharma, biotechs, and research institutions

- ✓ stores all types of multi-omics data as arrays and groups
- ✓ multiple ingestors
- ✓ provides a holistic catalog
- ✓ decentralizes data ownership
- ✓ centralizes governance and sharing of notebooks and dataset
- ✓ provides a common and scalable compute infrastructure and APIs
- ✓ allows definition and documentation of “data products” with the concept of shareable virtual “groups”

► BENEFIT

Significantly reduced data engineering and infrastructure hassles



Let's dive in!

Training and inference

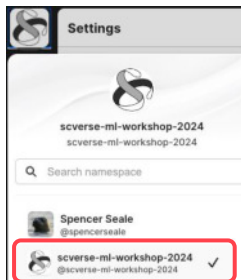
[tile]DB
DESIGNED FOR DISCOVERY™

+

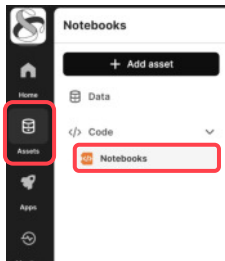
Chan
Zuckerberg
Initiative 

1 Login at cloud.tiledb.com

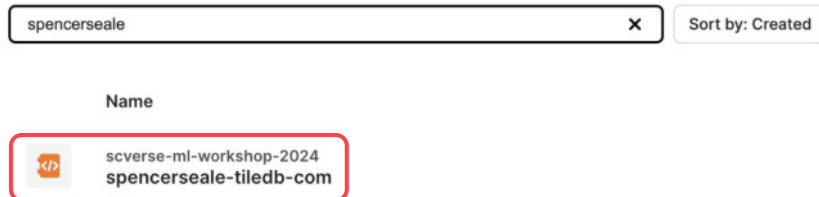
2 Switch to workshop Namespace (top-left)



3 Go to "Assets" & "Notebooks" (left)

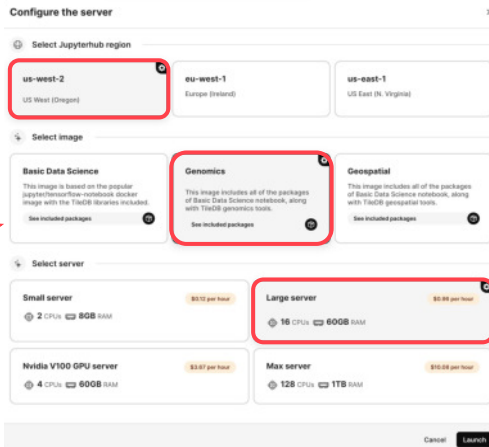


4 Search by your email username and select your notebook (top-right)

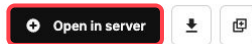


5 Launch your notebook with the configurations:

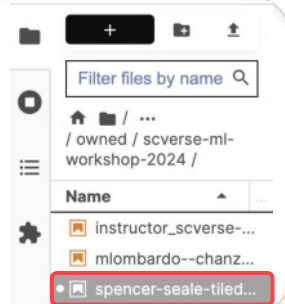
- **us-west-2**
- **Genomics**
- **Large server**



6 Display server

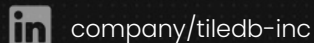


7 Select notebook



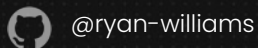
Thank you!

TileDB

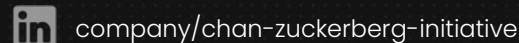
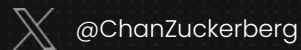


Ryan Williams, TileDB

ryan.williams@tiledb.com



CZI-wide



Maximilian Lombardo, CZI

mlombardo@chanzuckerberg.com



Similarity search UX
survey:

