Cloud-based data ingest for the Human Cell Atlas

June 27th
BOSC
Portland

Daniel Vaughan and Brian O’Connor
The Human Cell Atlas

To create a comprehensive reference maps of all human cells, as a basis for both understanding human health and diagnosing, monitoring, and treating disease.
The Human Cell Atlas

- Hundreds of labs
- Billions of cells
- Petabytes of data

Image Credit: EnCor Biotechnology Inc and Gerry Shaw
The Team

62 scientists and engineers

4 different locations

3 time zones

~30 video conferences a month
The Data Coordination Platform
Ingest

UI

Programmatic API

Metadata Standards

Validation
Data Store
Secondary Analysis Pipelines
Standardized Results
Data Access

Search, download, and compute on all data

Web Browser
Command Line
Multiple Clouds
Ingest Infrastructure Objectives

- Painless submissions
- High-quality, FAIR data
- Scalable, flexible infrastructure
Metadata Model Example
JSON all the way down
Spreadsheet Generation

spreadsheet_builder.py

Ingest Schema API

Get latest submittable type schemas

Generates full empty spreadsheet

External layout config e.g. 10x.yaml

File given to submitter
Ingest Services
Kubernetes - Efficient Resource Usage

Kubernetes Comic: https://cloud.google.com/kubernetes-engine/kubernetes-comic/
Kubernetes - Resilience & Flexible Scaling

Kubernetes Comic: https://cloud.google.com/kubernetes-engine/kubernetes-comic/
Push-Button multi-cloud deployment

- Terraform
- HELM
- openstack
- Amazon Web Services
- Google Cloud Platform
- Microsoft Azure
NeMO - Neuroscience Multi-Omic Archive

- NIH-funded repository for BRAIN Initiative (and other) data at IGS/UMB.
- Tasked with ingestion/archival of brain-related omic data.
- Adapting HCA ingest/storage infrastructure for this purpose.
- Have a working NeMO purple box and S3-only (for now) blue box.
- Currently collaborating with HCA team on:
  - Security for restricted access/embargoed datasets
  - Neuroscience-specific metadata schema extensions
  - Support for ingesting existing analysis results directly
    - i.e., versus adding them post-ingest via green box
The HCA Data Coordination Platform

an open source, cloud-based system for ingesting, storing, analyzing, discovering, and exploring single-cell data

June 27th
BOSC
Portland
Daniel Vaughan and Brian O’Connor
What can the DCP do for you?

Provide access to HCA data in order to...

- Make biological discoveries
- Develop and benchmark analysis methods
- Create tools and applications
- Facilitate collaborations

*Provide key Open Source, cloud-based infrastructure components that are reusable for other projects and efforts*
DCP Design and Components
The Data Coordination Platform
The Data Coordination Platform
Blue Box aka HCA Data Store provides:

- Multi-cloud sync
- Uniform cloud access
- Indexing and search across metadata
- Signed URLs, direct URLs via checkout
- Data and metadata organized into bundles
Data Store - Data Bundles

**Data Bundles**, groupings of related files, are a fundamental part of the Data Store.

- Files and data bundles are versioned.
- Files can be metadata or data.
- Client allows retrieval of full bundles or individual files (latest or specific version).
Data Store - API

- **Async copy a bundle to cloud or local store**
  - POST /bundles/(uuid)/checkout: Check out a bundle to DSS-managed or user-managed cloud object storage destination
  - GET /bundles/checkout/(checkout_job_id): Check the status of a checkout request.

- **Searching and retrieving bundles and files**
  - POST /search: Find bundles by searching their metadata with an Elasticsearch query
  - GET /bundles/(uuid): Retrieve a bundle given a UUID and optionally a version.
  - GET /files/(uuid): Retrieve a file given a UUID and optionally a version.
  - HEAD /files/(uuid): Retrieve a file's metadata given an UUID and optionally a version.

- **Requires a login credential**

- **Get notified about new data**
  - GET /subscriptions: Retrieve a user's event subscriptions.
  - PUT /subscriptions: Create a event subscription.
  - DELETE /subscriptions/(uuid): Delete an event subscription.
  - GET /subscriptions/(uuid): Retrieve an event subscription given a UUID.
Data Store - CLI

- CLI: [https://pypi.python.org/pypi/hca](https://pypi.python.org/pypi/hca)
  - $> pip install hca
  - $> hca dss -h
The Data Coordination Platform
A system for automated analysis of HCA data using best practices

- WDL-based workflows, registered on Dockstore, executed in the Cromwell as a service on the Google Cloud environment
Contributing Pipelines

**Unity (coming soon):**
- A place to benchmark pipelines.
  - Same reports, same data, cloud backend.
  - A place to compare against production pipelines.
- Encourage standards used in the green box.
- Service to build communities around pipelines.
The Data Coordination Platform
Discover

Single-cell data open and accessible for all

Start Exploring
The Data Browser indexer subscribes to events in the Dats Store
We ETL the metadata into a schema capable of driving facets via ElasticSearch

Want other portals to leverage this code
Finding Data via the Data Browser

- Example: Select all bone marrow datasets
- Download manifest of search results
- Use in your favorite environment
Handoff Mechanism Connects the Data Browser to Portals

Data Browser

Send your search to Portal

Tertiary Analysis Portal

I support receiving specimen IDs or file IDs for gene x cell matrix files

Here’s a list of specimen IDs to render for the user....

Want to connect many different portals
Infrastructure Use by HCA Researchers
Preview Datasets Available Now!

Human Cell Atlas Preview Datasets

The first single-cell sequencing datasets from the Human Cell Atlas are now available to the research community.

<table>
<thead>
<tr>
<th>Census of Immune Cells</th>
<th>Ischaemic Sensitivity of Human Tissue</th>
<th>Melanoma Infiltration of Stromal and Immune Cells</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Species</strong></td>
<td><strong>Species</strong></td>
<td><strong>Species</strong></td>
</tr>
<tr>
<td>Homo sapiens</td>
<td>Homo sapiens</td>
<td>Mus musculus</td>
</tr>
<tr>
<td><strong>Organ</strong></td>
<td><strong>Organ</strong></td>
<td><strong>Lymph node</strong></td>
</tr>
<tr>
<td>Umbilical cord blood and bone marrow</td>
<td>Spleen</td>
<td></td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td><strong>Method</strong></td>
<td><strong>Method</strong></td>
</tr>
<tr>
<td>10x</td>
<td>10x</td>
<td>Smart-seq2</td>
</tr>
<tr>
<td><strong>Cell count</strong></td>
<td><strong>Cell count</strong></td>
<td><strong>Cell count</strong></td>
</tr>
<tr>
<td>~530,000 cells</td>
<td>~2,000 cells</td>
<td>6,639 cells</td>
</tr>
<tr>
<td><strong>File size</strong></td>
<td><strong>File size</strong></td>
<td><strong>File size</strong></td>
</tr>
<tr>
<td>1.3 TB</td>
<td>14 GB</td>
<td>380 GB</td>
</tr>
</tbody>
</table>

https://preview.data.humancellatlas.org
Query HCA Metadata

*Search metadata to find and access data files*

**Metadata search**
- Organ
- Species
- Sex
- Tissue type
- etc..

**Find and access:**
- Expression matrix
- Metadata
- Raw files (fastq, tiff)
- BAM
- QC
Accessing HCA Data

Interfaces: command line, API, and web GUI (summer)
Search: browse or use search API
Access: copy data or use in place
Exploring Example Code

Indexed Metadata

The Investigate a Bundle notebook walked through the contents of a bundle (and you probably should step through it before looking at this notebook). There we started with a bundle uuid, but what if we wanted to find that bundle uuid based on some facts we knew about the bundle?

Recall the bundle we were looking at:

```python
In [ ]:
import hca

dss_client = hca.dss.DSSClient()
bundle_uuid = 'ec7e4476-abe9-4d76-82d6-8a28b43c46a6'
bundle = dss_client.get_bundle(uuid=bundle_uuid, replica="aws")
```

And recall that bundle contained some json files, one of which was biomaterial.json. If you look at the contents of the bundle, you can see that that file has an "indexed" field, which means that the contents of that file can be searched. So let’s look at the contents of biomaterial.json, again:

```python
In [ ]:
biomaterial_json_uuid = [f['uid'] for f in bundle['bundle']['files'] if f['name'] == 'biomaterial.json'][0]
dss_client.get_file(uuid=biomaterial_json_uuid, replica="aws")
```

There are a number of things in there that we could search for, but let’s say we want to find bundles associated with that biomaterial_id, BT_S2_T:

```python
In [ ]:
es_query = {
    'query': {
        'term': {
            'files.biomaterial_json.biomaterials.content.biomaterial_core.biomaterial_id': 'BT_S2_T'
        }
    }
}
result = dss_client.post_search(replica="aws", es_query=es_query)
print(result['total_hits'])
result
```

DCP Component Reuse by Other Projects
A Data Biosphere for Biomedical Research

We, the authors listed below, are privileged to be part of the growing global community bringing data and life science together. Our groups have been working together in overlapping combinations during the past two years to drive the creation of data commons to support flagship scientific initiatives. This document lays out our evolving vision for the next steps in that journey. Our hope is that others will join the effort to build momentum for an open, compatible, and secure approach to data within the larger research community. We welcome your feedback, and look forward to continuing this journey together.
The Data Biosphere

Scientific Portals
- Portal
- Portal
- Portal
- Portal

Other applications
- App
- App
- App

Index & Search
Workspaces
Analytical Engines
Tool Repositories

Digital Object Catalog

Data Assets
Data Assets
Data Assets
Data Assets

(Services below this line are managed by cloud vendors)

Cloud 1
Cloud 2
Cloud 3
Cloud 4
HCA Component Reuse

Blue Box - HCA Data Store

Orange Box - Data Browser

Green Box - Cromwell Service

NHLBI Data
STAGE

NIH Data Commons

AnVIL

And more to come...
What is next for the DCP?

DCP meeting in Hinxton, working to integrate, improve and plan for the future

Continue collecting data, improving services, and gathering feedback

Early data submissions, user testing, improving scaling and sustainability of DCP

Pilot launch of DCP

2018

2019
How to get involved?

1) Try the Preview HCA system and APIs, see:
   a) https://preview.data.humancellatlas.org (data)
   b) https://dss.data.humancellatlas.org (api)
   c) https://github.com/HumanCellAtlas (code)

2) Use/extend/improve our software components, see:
   a) https://github.com/HumanCellAtlas

3) Email for help or feedback data-help@humancellatlas.org
Thank You!