

# Basin: the MITOS Data Platform



Yu Cheng June 11, 2024

## Warm Up Poll

-0





## Yu Cheng 鄭嵎

Sustainability Data Scientist MIT Office of Sustainability

S 🖪 🕄 🖓 🖬

 Five years working as data scientist/consultant in a few startups.

-0

- Tackling climate change with a unique blend of backgrounds
- Joined MIT Sep. 2023



#### **Brief intro of MITOS**

Get to know the Office of Sustainability

#### 0-----0

4

74

3

#### **Data Challenges**

Observations and learnings from existing data practice

#### 0------

#### Basin: walkthrough

Introducing 'Basin', the MITOS data platform, featuring data lineage, cataloging, testing and scheduling.

#### 0-----0

#### Looking Ahead

Outlining forthcoming goals and potential enhancements to 'Basin'.

0-----0

## **Brief Intro of MITOS**

0-----0

## Overview

 Founded in 2013 under Executive Vice President & Treasurer's Office (EVPT) to integrate sustainability across all levels of our campus operations.

• Organizational lead for campus goals on Fast Forward

E38 Third floor, above the MIT Welcome Center

### MITOS Team 2024





campus climate action

#### How do we solve for sustainability at MIT?



## **Scales of Impact**



# Projects & Events

- Annual Sustainability Connect conference
- Food waste fighters, Choose-to-reuse etc.
- Offsite Renewable Energy Project development
- Co-lead Fast-Forward Workstreams and the Decarbonization working group
- Sustainability Datapool

## Data Challenges

0\_\_\_\_0

## Sustainability Datapool

-0



## Challenges

• Excel files One-off projects Multiple versions and locations of data artifacts Limited data quality control, and documentation Datahub, Datapool, and Dashboards are not in sync.



# Case Study: Scope 3 Emission

"tableau\_all\_scope.xlsx" in multiple folders
Lacking documentation how to update
Category Mapping
Emission Factors are not consistent.

## Basin: walkthrough

0\_\_\_\_0



# A river basin consists of many streams.

# Objectives

A centralized repository to track all pipelines
Transparency in data lineage and quality
Consistency between reported and shared data
Modular and easy to build on top.



# Dagster

An open-source data orchestrator
Declarative pipeline using Python
Scheduling, monitoring, and observability
Easy integration with other tools and solutions

## **Asset Definition**

```
@asset(
    io_manager_key="postgres_replace",
    compute_kind="python",
    group_name="raw",
def expense_emission_mapper(dhub: ResourceParam[DataHubResource]):
    """This asset ingest the expense_type_to_emissions.json from the Data Hub"""
    project_id = dhub.get_project_id("Scope3 Business Travel")
    logger.info(f"Found project id: {project id}!")
    download_links = dhub.search_files_from_project(project_id, "expense type to emissions.json")
    if download links is None:
        logger.info("No download links found!")
        return pd.DataFrame()
    response = requests.get(download_links[0], timeout=10)
    if response.status code = 200:
        payload = json.loads(response.text)
    mapper = {v: key for key, value in payload.items() for v in value}
    df = pd.DataFrame(list(mapper.items()), columns=["expense_type", "emission_category"])
    return df
```

## **Data Lineage and Observability**

| const  | Construction_asset_job 📽 Job in orchestrator C 💿 Latest run: Mar 2, 8:53 PM 🖽 View 8 assets |                      |                               |   |                            |  |     |  |  |                           |
|--------|---|----------------------|-------------------------------|---|----------------------------|--|-----|--|--|---------------------------|
| Overvi | Overview Runs   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      | ant low unstaning late const  | ruction evenes)                                   |                            |  |     |  |  |                           |
|        | ¥ Fitter  | Type an asset subs   | set (ex: ++staging/stg_const  | ruction_expense)                                  |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   | orchestrator         |                               | ×   | staging ×                  |  | ×   |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               | ☐ dof_maintenance_cost                            |                            | 🖩 stg_dof_maintenance_cost                     |     |  | final  | •                         |
|        |   |                      |                               | This asset ingest the maintenance cost data fro   |                            | Adjusted fiscal year maintenance cost and GHG  | 3 e |  | orchestrator                                 |                           |
|        |   |                      |                               | Materialized May 29, 1:26 PM                      |                            | Never materialized                             |     |  | (5 Unevoced (4)                              |                           |
|        |   |                      |                               | n 🔁 Python  |                            |  | dbt |  | Ste onlynded (4)                             |                           |
|        |   |                      |                               |   |                            |  |     |  | <pre>m construction_expense_emission</pre>   | Ed dhub construction cost |
|        |   |                      |                               |   |                            |  |     |  | Construction expense and GHG emission broken | , anab_construction_cost  |
|        |   |                      |                               |   |                            |  |     |  | Checks                                       |                           |
|        |   |                      |                               | emission_factor_naics                             |                            |  |     |  |  |                           |
|        |   |                      |                               | Ingest and combine NAICS and USEEIO emission      |                            |  |     |  | X db   | 9                         |
|        |   |                      |                               | Materialized May 29, 1:27 PM                      |                            | (1) Unsynced (3)                               |     |  |  |                           |
|        |   |                      |                               | n Python  |                            |  |     |  |  |                           |
|        |   |                      |                               |   | $  \cdot \rangle \nearrow$ | Adjusted fiscal year construction cost and GHG | i e |  |  |                           |
|        |   |                      |                               |   |                            | Materialized Mar 2, 8:54                       | PM  |  |  |                           |
|        |   |                      |                               | This asset ingest the construction expense data f |                            | ( <b>X</b>                                     | dbt |  |  |                           |
|        |   |                      |                               | Materialized May 29 1:26 PM                       |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               | Python  |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   | 🖩 emission_fa        | actor_useeio_v2               |   | 1:7:1                      |  |     |  |  |                           |
|        |   | This asset ingest th | he USEElOv2.0.1 emission fact |   |                            |  |     |  |  |                           |
|        |   | Materialized         | May 29, 1:26 PM               |   |                            |  |     |  |  |                           |
|        |   |                      | n Python                      |   |                            | 1  |     |  |  |                           |
|        |   |                      |                               |   | 17::::                     |  |     |  |  |                           |
|        |   |                      |                               | 🖂 annual cpi index                                |                            | ☑ stg_purchased_goods_invo:                    | ice |  |  |                           |
|        |   |                      |                               | Get annual CPI index from python cpi library      |                            |  |     |  |  |                           |
|        |   |                      |                               | Materialized May 29. 1:25 PM                      |                            |  |     |  |  |                           |
|        |   |                      |                               |   |                            |  |     |  |  |                           |
|        |   |                      |                               | Python  |                            |  |     |  |  |                           |
| 🔵 v    | iew as Asse   | Graph                |                               |   |                            |  |     |  |  |                           |

## **Run History**

-0

| Asset | s  | Provide the second definitions Contract of the second definitions and the second definitions and the second definitions and the second definition of the second definition |   |     |  |
|-------|--|---|---|-----|--|
|       | <b>T</b> Filter Filter asset keys  |   | 0:02 C + Materialize selected                       | • • |  |
|       | Asset name   | Code location / Asset group   | Status  |     |  |
|       | all_scope_summary     this asset ingest the all_scope summary data from                      | □ orchestrator<br>匾 raw   | Materialized • Feb 1, 11:56 AM                      | •   |  |
|       | annual_cpi_index     cet annual CPI index from python cpi library                            | <ul><li>□ orchestrator</li><li>□ raw</li></ul>  | <ul> <li>Materialized - Jan 27, 7:30 AM</li> </ul>  | •   |  |
|       | ☐ commuting_emission_factors_EPA<br>This asset ingest the Commuting Emission Factors         | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | Materialized - Apr 16, 8:34 PM                      | ~   |  |
|       | ☐ commuting_survey_2018 Provide the Commuting Survey 2018 dat                                | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | Materialized - Apr 16, 8:36 PM                      | ~   |  |
|       | ☐ commuting_survey_2021 This asset ingests the Commuting Survey 2021 dat                     | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | Materialized - Apr 16, 8:34 PM                      | •   |  |
|       | ☐ commuting_survey_2023 This asset ingests the Commuting Survey 2023 dat                     | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | Materialized - Apr 16, 8:35 PM                      | •   |  |
|       | ☐ commuting_survey_modes This asset ingests the mode breakdown across yea                    | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | <ul> <li>Materialized - May 3, 11:02 AM</li> </ul>  | ~   |  |
|       | <ul> <li>☐ construction_expense</li> <li>Provide the construction expense data fr</li> </ul> | <ul> <li>□ orchestrator</li> <li>□ raw</li> </ul>   | <ul> <li>Materialized - Feb 21, 12:04 PM</li> </ul> | ~   |  |

## **Processing Stages and Scheduling**

|             |  | · · · · · · · · · · · · · · · · · · ·                 | · · · · · · · · · · · · · ·               |                             |                             |                                |
|-------------|--|---|---|-----------------------------|-----------------------------|--------------------------------|
|             |  | staging   | •   |                             |                             |                                |
| ra          | w o  |   |   |                             | dhub_                       | sync                           |
|             | · · · · · · · · · · · · · · · · · · ·  |   | final                                     |                             | •                           |                                |
|             |  |   | · · · · · · · · · · · · · · · · · · ·     |                             |                             |                                |
|             |  |   |   |                             |                             |                                |
|             |  |   |   |                             |                             |                                |
| =           |  | ponto Donloumont                                      |   |                             |                             |                                |
|             | Overview Rulis As  | sets Deployment                                       |   |                             |                             |                                |
|             |  | ssets Deployment                                      |   |                             |                             |                                |
| Ove         | erview   | ssets Deployment                                      |   |                             |                             |                                |
| Ove         | erview   | ssets Deployment                                      |   |                             |                             |                                |
| Ove         | erview<br>vity Jobs Schedules Senso  | ors Auto-materialize • Reso                           | urces Backfills                           |                             |                             |                                |
| Ove         | erview<br>vity Jobs Schedules Senso  | ors Auto-materialize • Resou                          | urces Backfills                           |                             |                             |                                |
| Ove<br>Acti | vity Jobs Schedules Senso<br>Filter  | ors Auto-materialize • Resou                          | urces Backfills                           |                             |                             | Actions ~                      |
| Ove<br>Acti | vity Jobs Schedules Senso<br>Filter Q Filter by schedule na  | ors Auto-materialize • Resou<br>Schedules<br>Schedule | urces Backfills                           | 1 Last tick                 | Last run                    | Actions ~                      |
| Ove<br>Acti | vity Jobs Schedules Senso<br>Filter Q Filter by schedule na<br>Schedule name<br>orchestrator   | ors Auto-materialize  Resou                           | urces Backfills                           | J Last tick                 | Last run                    | Actions ~<br>Actions<br>2      |
| Ove<br>Acti | vity Jobs Schedules Senso<br>Filter Q Filter by schedule na<br>Schedule name<br>orchestrator<br>business_asset_job_schedule  | e At 12:00 AM UTC, o                                  | urces Backfills<br>Running                | g Last tick                 | Last run<br>None            | Actions ~<br>Actions<br>2      |
| Ove<br>Acti | vity       Jobs       Schedules       Sensor         Filter       Q. Filter by schedule name         Schedule name         orchestrator         business_asset_job_schedule         P business_asset_job   | e At 12:00 AM UTC, or                                 | urces Backfills Running n day 1 of the mo | Last tick                   | Last run<br>None            | Actions ~<br>Actions<br>2      |
| Ove<br>Acti | erview         vity       Jobs         Schedules       Sensor         Filter       Q. Filter by schedule name         Schedule name       Schedule name         orchestrator       business_asset_job_schedule         Base business_asset_job       materialize_dbt_models_schedule | e At 12:00 AM UTC, or                                 | urces Backfills Running n day 1 of the mo | J Last tick<br>None<br>None | Last run Last run None None | Actions ~<br>Actions<br>2<br>– |

# dbt (Data Build Tool)



Open-source tool for SQL-based data modeling

- Versioned controlled SQL queries, reusable macros.
- Support various SQL flavors, i.e. BigQuery, Redshift, Oracle, and Postgres
- Automatically generation of documentation and data lineage.
- Easy integration with orchestrator like Dagster and Airflow

## **SQL-based Data Modeling**

```
-- set static variables using jinja2 syntax
{% set reply_rate = 0.33 %}
\{\% \text{ set work week} = 50 \%\}
{% set remote_rate = 0.21 %}
WITH distance AS (
    SELECT
        "drove alone" * {{ var('car speed') }} * commute time average hours AS drove alone,
        "carpooled(2-6)"
        * {{ var('car speed') }}
        * {{ var('car_share_ratio') }}
        * commute time average hours AS carpooled,
        "vanpooled(7+)"
        * {{ var('car speed') }}
        * {{ var('van share ratio') }}
        * commute_time_average_hours AS vanpooled,
        shuttle * {{ var('bus_speed') }} * commute_time_average_hours AS shuttle,
        "public transportation"
        * {{ var('t_ratio') }}
        * {{ var('t_speed') }}
        * commute time average hours AS subway,
```

### Documentation

Search for models...

stg\_commuting\_survey\_2018 view

#### Overview

stg\_commuting\_survey\_2018
 stg\_commuting\_survey\_2021
 stg\_commuting\_survey\_2023
 stg\_construction\_expense
 stg\_cost\_object\_rollup
 stg\_dof\_maintenance\_cost

stg\_travel\_spending

E

|           | Project                         | T Database | Details | Description    | Columns       | Referenced By    | Depends On | Code |
|-----------|---------------------------------|------------|---------|----------------|---------------|------------------|------------|------|
|           | Ŧ                               | Group      |         | Priori         |               |                  |            |      |
| Т         | ables and Views<br><b>B</b> dbt |            | Es      | timated GHG en | nissions fron | n Commuting Surv | vey 2018.  |      |
| "Eª final |                                 | Colum      | ns      |                |               |                  |            |      |
|           | E raw                           |            |         |                |               |                  |            |      |
|           | T staging                       |            |         |                |               |                  |            |      |

| COLUMN | ТҮРЕ | DESCRIPTION                                  | CONSTRAINTS | TESTS | MORE? |
|--------|------|--|-------------|-------|-------|
| mode   |      | Commute mode                                 |             |       | >     |
| share  |      | Share of people using the mode in the year   |             |       | >     |
| miles  |      | Daily commute miles by mode                  |             |       | >     |
| mtco2  |      | Anual equivalent CO2 emission in metric tons |             |       | >     |
|        |      |  |             |       |       |

MIT Office of Sustainability

## Data Warehouse

Use a RDS managed Postgres instance
Processing stages managed by "schema"
Isolated local development and production environments

 All dashboards serve from the Postgres warehouse.



# IS&T DataHub

MIT's "data lake" powered by AWS
 Web UI to interact with projects and files

Project-based data access management
API support to handle files programmatically.

## CI/CD

 Use Github Actions to run tests, build image and documentation, push to ECR, and deploy.

Use Docker to ensure portability

 Infrastructure as code: With the AWS Cloud Formation template to setup cloud resources in minutes.

## Demo

#### • Dagster

\_\_\_\_O

 $\cap$ 

- Dbt documentation
- Github Repo

## Looking Ahead

0\_\_\_\_0

## Benefits

• Version controlled pipelines Automated build/deploy process Transparent data lineage and run history Enhanced data quality control Accelerated development cycle Integrated with MIT Data Hub • Single source of truth

## Improvement Ideas

- Enhance data validation and quality checks: explore tools like Great Expectations and Pandera
- Add more unit tests
- Explore scalable structures, i.e. Redshift and ECS.
- Improve on Infrastructure as code
- Implement more pipelines

## Looking for Collaborators

#### <u>Github Repo</u>

- This project is open-source. As the sole developer in the office, I need help to improve the code base.
- The data-warehouse/data-platform approach is a great first step for small teams striving for a data-driven culture.
  Join me <sup>(C)</sup>