`git push` your data stack with Airbyte, Airflow, and dbt
Hello!

Evan Tahler
Staff Engineer
@evantahler
evantahler
/in/evantahler

Marcos Marx
User Success Engineer
marcosmarxm
/in/marcosmarxmillnitz

Airbyte
Open-Source Data Integration

20,000 users
7,500 Slack members
6,500 GitHub stars
Sample Code!


The goal of this talk is to share an example of how we can use today’s tools to automate your data stack and gain the benefits of Continuous Integration and Continuous Deployment (CI/CD)
What is Airbyte?
**Extract**

General-purpose routines to pull selected data from a source.

**Load**

General-purpose routines to push raw data where it is going to be consumed.

**Transform**

Business logic specific to your organization to serve an analytics or operational use case with SQL / dbt / ...
The Modern Data Stack
The Modern Data Stack
The Modern Data Engineering Workflow
Data Teams == Engineering Teams

Modern Data Teams are Engineering Teams. We make data products consumed by others. To be successful, reliable, and fast, modern engineering workflows are needed.

- Version Control
- Tests
- Continuous Deployment
Data Teams == Engineering Teams

Modern Data Teams are Engineering Teams. We make data products consumed by others. To be successful, reliable, and fast, modern engineering workflows are needed.

- **Version Control**
  - Track and test changes before they are deployed
  - Rollback when problems occur
  - Manage approvals & reviews

- **Tests**
  - ... with the confidence that nothing broke

- **Continuous Deployment**
  - ... deployed quickly and automatically, perhaps to multiple environments
Version Controlled Airbyte: The Octavia CLI

This is Octavia ->
$ octavia --help
Usage: octavia [OPTIONS] COMMAND [ARGS]...

Options:
    --airbyte-url TEXT     The URL of your Airbyte instance.
    --workspace-id TEXT    The id of the workspace on which you want octavia-cli to work. Defaults to the first one found on your Airbyte instance.
    --help                 Show this message and exit.

Commands:
    apply Create or update Airbyte remote resources according local...
    delete [NOT IMPLEMENTED] Delete resources
    generate Generate a YAML template for a source, destination or a...
    import [NOT IMPLEMENTED] Import an existing resources from the...
    init Initialize required directories for the project.
    list List existing Airbyte resources.
generate YAML files that can be version controlled in Git

leverage any scripting or templating tool to generate configurations dynamically for creation of data sources, destinations, and connections

deploy configurations on multiple Airbyte instances, in order to manage several Airbyte environments

integrate into a CI workflow to automate the deployment of data integration configurations

- This allows for the same configuration to be run against multiple environments - running the same ELT process against staging or test data by changing environment variables

- Secrets can be in the system/shell Environment, or a local dotfile

```
\`octavia` Released in April, 2022!
```
Coming Soon for `octavia`

- Deterministic Source and Destination IDs for idempotent recovery
- Deterministic Connection IDs to orchestrate with other applications

For now, we have some custom code to make this work - change_resource_id.py
Demo Time!

www.github.com/airbytehq/airflow-summit-airbyte-2022
Our Demo Stack 100% Open Source, of course

- Airflow (Orchestrator)
- Airbyte (Extract & Load)
- dbt (Transform)
Our Demo Stack 100% Open Source, of course
DEMO PART 1

In which everything works great
Demo: Docker Compose

- For easy developer setup
- For easy CI setup

Just run `./tools/start.sh` and everything will be configured and the Airflow DAG will run
Demo:
Orchestrate w/ Airflow
Demo:

Extract & Load w/ Airbyte
Demo: Extract & Load w/ Airbyte
Demo: Extract & Load w/ Airbyte

Don't worry, it's fake data from the `faker` package
Demo: Transform w/ dbt
Demo: Transform w/ dbt

Don’t worry, it’s fake data from the `faker` package
Demo Notes:

Future Work:

1. Custom code to change Airbyte Connection ID: `change_resoource_id.py`
2. Custom docker image for Airflow w/ dbt included
3. Using a hosted Postgres database in CI so we can inspect the data

Pull Requests Welcome!
DEMO PART 2

In which Evan breaks the build
THE TWELVE FACTORS

I. Codebase
One codebase tracked in revision control, many deploys

II. Dependencies
Explicitly declare and isolate dependencies

III. Config
Store config in the environment

IV. Backing services
Treat backing services as attached resources

V. Build, release, run
Strictly separate build and run stages

VI. Processes
Execute the app as one or more stateless processes

VII. Port binding
Export services via port binding

VIII. Concurrency
Scale out via the process model

IX. Disposability
Maximize robustness with fast startup and graceful shutdown

X. Dev/prod parity
Keep development, staging, and production as similar as possible

XI. Logs
Treat logs as event streams

XII. Admin processes
Run admin/management tasks as one-off processes

CI/CD Ethos (since 2012)

- All Code in git
- Git is used to deploy
- All configuration is done via environment variables
- ... that’s it!

A historical aside while we wait for tests to run...
Demo: A small change...
Failing PRs are great.

- The data product and our users are protected
- Clear and immediate feedback
- A place for the team to communicate about what to do next
- Less bugs and a better night’s sleep
What’s next?

*Fix the PR 😞... But then what?*

- **Automatic Deployment from the `main` branch**
- **Health-checks on the production system (both infrastructure and data)**
- **Better Tests:**
  - We have an end-to-end Integration Test
  - Unit Tests - Checking specific known rows
  - Acceptance Tests - Great Expectations
  - Linting - SQLFluff

*The entry point for all of the above is another Github Action that checks PRs and each deployment*
`git push` your data stack with Airbyte, Airflow, and dbt

Airflow Summit 2022
THE END
Template Slides Below
1. Section 1
2. Section 2
3. Section 3
4. Section 4
5. Section 5
Section title
Section title
Section title
Summary

- `#troubleshooting` to Troubleshooting
- `#troubleshooting` to Troubleshooting
- `#troubleshooting` to Troubleshooting
- `#troubleshooting` to Troubleshooting
- `#troubleshooting` to Troubleshooting
- `#troubleshooting` to Troubleshooting
Slide Main Title Sub section title

- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
- #troubleshooting to Troubleshooting
### What is Airbyte?

Airbyte is an open-source and hosted ELT solution that replicates data from a long-tail of APIs, databases, and files to data.

<table>
<thead>
<tr>
<th>Mission</th>
<th>Power organizations' data movement.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary persona</td>
<td>Data engineers.</td>
</tr>
<tr>
<td>Tag line</td>
<td>The data integration platform for any connection</td>
</tr>
</tbody>
</table>

**Narrative**

**Mission**

Power organizations' data movement.
#advice channels
#advice-data-orchestration
#advice-data-ingestion
#advice-data-architecture
#advice-data-transformation
#advice-data-quality
#advice-data-warehousing
#advice-reverse-etl
#advice-data-privacy
#advice-data-visualization
Slide Main Title  Sub section title

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet lorem ipsum dolor sit amet
Slide Main Title Sub section title

Bloc title
- Conferences
- Sponsored events
- Meetup program with ambassadors
- Community calls

Bloc title
- Conferences
- Sponsored events
- Meetup program with ambassadors
- Community calls

Bloc title
- Conferences
- Sponsored events
- Meetup program with ambassadors
- Community calls
Slide Main Title  Sub section title

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate
Slide Main Title Sub section title

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate

Abhi Vaidyanatha
Senior dev advocate
“Quote lorem ipsum dolor sit lorem ipsum dolor sit amet lorem ipsum”