Building Reusable and Trustworthy pipelines
Outline

1. Context
2. Design Requirements
3. Proposed Solution
4. Example Code
Context
Hello 👋!

- Data engineer @ SnapTravel
- SnapTravel
  - M-commerce startup
  - Data team: 8, Data Sources: 86
- Data infrastructure, Data engineering, Analytics engineering
- 🚖 + 🌊 + 🌍 + 💪 stack
Purpose

Share 🎒 BI pipelines
💡 Community with lessons learnt
🎧 feedback
How are my company?

- gross_revenue
- contribution_margin
- number_of_active_users
- retention_rate
- conversion_rate
Hows my airflow repo 📊?

- number_prs_merged
- number_prs_closed_without_merge
- number_prs_opened
- number_of_commits
Extract

Transform

Load

Report

Fetch Data

Process data to generate metrics

Load to a DB

Visualize in Superset
Let us consider

- The pipeline failed in production
- Shift focus on to issues, comments
- Gitlab released a new version of API
- I want to analyze other apache projects too
- Github produced similar insights and their numbers didn't match mine
Been there done that?
Classify the problems

- Toil
- Cannot scale Data Analytics
- Data Discovery
- Data Trust
- Throw over the boundary
- Ambiguous ownership
What can we do to solve this?
..build tools, infrastructure, frameworks and services
– Maxime Beauchemin

I just published “The Rise of the Data Engineer”

The Rise of the Data Engineer
I joined Facebook in 2011 as a business intelligence engineer. By the time I left in 2013, I was a data engineer.

medium.com

6:44 PM · Jan 20, 2017

64  45 people are Tweeting about this
Design
Requirements
NO PATIENCE REQUIRED
FREE SAME-DAY DELIVERY

NEW IN YOUR AREA
Single Source of Truth

- Standardization
- Data Lineage
- Empower non-technical folks
Easy to consume

- Airflow + Other OSS
- Ideally `pip install awesome-elt-tool`
- Low barrier to entry for data analytics
- Operational creep
Promote data integrity

- Test the raw data supply
- Automated analytics testing
Proposed Solution
Conceptually
ETL vs ELT

- Load once and transform
- Reduced complexity
- Reduce cost
- Speed of delivery
Validate your source data
- expect_column_to_exist
- expect_table_row_count_to_be_between
- expect_table_row_count_to_equal
- expect_multicolumn_values_to_be_unique
- expect_column_values_to_not_be_null
- expect_column_values_to_be_null
- expect_column_fancy_statistic_to_be
Why?

- Profiling
- Data Docs <-> Tests
- Send notifications automatically
Extract – Load
Singer – What?
tap-github --config tap_config.json | target-postgres --config target_config.json >> state.json
Singer – Why?

- Standardized communication
- Incremental out of the box
- Documentation
- See your data in under 10 mins
It's a long list
Transform
DBT – What?

Data Loaders → Raw Data → Snapshot → Transform → Test → Deploy → Document → Transformed Data → BI Tools

Data Warehouse

Version Control
Alerting
Logging

Data Science
Building dependencies between dbt models

```sql
select
    orders.order_id,
    orders.customer_id,
    customers.first_name as customer_first_name,
    customers.last_name as customer_last_name,
    orders.amount,
    orders.ordered_at

```
This model represents one record per month, per account (months have been filled in to include any period). This model classifies each month as one of: new, reactivation, upgrade, downgrade, or churn.

<table>
<thead>
<tr>
<th>COLUMN</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>TEXT</td>
<td></td>
</tr>
<tr>
<td>date_month</td>
<td>TIMESTAMP_NTZ</td>
<td></td>
</tr>
<tr>
<td>customer_id</td>
<td>NUMBER</td>
<td></td>
</tr>
</tbody>
</table>
DBT – Why?

- Modular code

```sql
{% set payment_methods = ['bank_transfer', 'credit_card', 'gift_card'] %}

select
    order_id,
    {% for payment_method in payment_methods %}
    sum(case when payment_method = '{{payment_method}}' then amount end) as {{payment_method}}
    {% endfor %}
    sum(amount) as total_amount
from app_data.payments
group by 1
```
DBT – Why?

- Modular code
- Testing is 1st Class
DBT – Why?

- Modular code
- Testing is 1st Class
- Data documentation is 1st Class

```json

description: This table contains clickstream events from the marketing website

columns:
  - name: event_id
description: This is a unique identifier for the event
tests:
  - unique
  - not_null
```
Great adoption
All together
Meltano

- Open Source, GitLab
- Self Hosted

```bash
pip3 install meltano
meltano init airflow-analytics-project
meltano add extractor tap-github
meltano add loader target-postgres
meltano add transformer dbt
meltano add transform tap-github
# add env variables
meltano elt tap-gitlab target-postgres --transform=run --job_id=gitlab-to-postgres
meltano add orchestrator airflow
```
Let's look at the code
version: 1
send_anonymous_usage_stats: true
project_id: 
plugins:
  extractors:
    - name: tap-github
      namespace: tap_github
      pip_url: 'git+https://github.com/nehiljain/tap-github.git'
      executable: tap-github
      capabilities:
        - discover
        - properties
      settings:
        - name: access_token
          env: TAP_GITHUB_ACCESS_TOKEN
        - name: repository
          env: TAP_GITHUB_REPOSITORY
    loaders:
      - name: target-postgres
        pip_url: 'git+https://github.com/meltano/target-postgres.git'
  transforms:
    - name: tap-github
      pip_url: 'https://github.com/nehiljain/dbt-tap-github.git'
      orchestrators:
        - name: airflow
          pip_url: wget==2.2.1 apache-airflow==1.10.2
      transformers:
        - name: dbt
          pip_url: dbt==0.16.1
      files:
        - name: airflow
          pip_url: 'git+https://github.com/meltano/files-airflow.git'
      schedules:
        - name: gitlab-to-postgres
          extractor: tap-github
          loader: target-postgres
          transform: skip
          interval: 'hourly'
          start_date: 2020-07-05 18:58:28.155924
A templated approach
version: 1
send_anonymous_usage_stats: true
project_id: 
plugins:
  extractors:
  - name: tap-github
    namespace: tap.github
    pip_url: 'git+https://github.com/nehiljain/tap-github.git'
    executable: tap-github
    capabilities:
    - discover
    - properties
    settings:
    - name: access_token
      env: TAP_GITHUB_ACCESS_TOKEN
    - name: repository
      env: TAP_GITHUB_REPOSITORY
  loaders:
  - name: target-postgres
    pip_url: 'git+https://github.com/meltano/target-postgres.git'
  transforms:
  - name: tap-github
    pip_url: 'https://github.com/nehiljain/dbt-tap-github.git'
  orchestrators:
  - name: airflow
    pip_url: wtforms==2.2.1 apache-airflow==1.10.2
  transformers:
  - name: dbt
    pip_url: dbt==0.16.1
  files:
  - name: airflow
    pip_url: 'git+https://gitlab.com/meltano/files-airflow.git'
  schedules:
  - name: gitlab-to-postgres
    extractor: tap-github
    loader: target-postgres
    transform: skip
    interval: 'hourly'
    start_date: 2020-07-05 18:58:28.155924
version: 1
send_anonymous_usage_stats: true
project_id: 

plugins:

extractors:
  - name: tap-github
    namespace: tap.github
    pip_url: 'git+https://github.com/nehiljain/tap-github.git'
    executable: tap-github
    capabilities:
      - discover
      - properties
    settings:
      - name: access_token
        env: TAP_GITHUB_ACCESS_TOKEN
      - name: repository
        env: TAP_GITHUB_REPOSITORY

loaders:
  - name: target-postgres
    pip_url: 'git+https://github.com/meltano/target-postgres.git'

transforms:
  - name: tap-github
    pip_url: 'https://github.com/nehiljain/dbt-tap-github.git'
  
orchestrators:
  - name: airflow
    pip_url: wtforms==2.2.1 apache-airflow==1.10.2

transformers:
  - name: dbt
    pip_url: dbt==0.16.1

files:
  - name: airflow
    pip_url: 'git+https://github.com/meltano/files-airflow.git'

schedules:
  - name: gitlab-to-postgres
    extractor: tap-github
    loader: target-postgres
    transform: skip
    interval: 'hourly'
    start_date: 2020-07-05 18:58:28.155924
version: 1
send_anonymous_usage_stats: true
project_id: 
plugins:
  extractors:
    - name: tap-github
      namespace: tap.github
      pip_url: 'git+https://github.com/nehiljain/tap-github.git'
      executable: tap-github
      capabilities:
        - discover
        - properties
      settings:
        - name: access_token
          env: TAP_GITHUB_ACCESS_TOKEN
        - name: repository
          env: TAP_GITHUB_REPOSITORY
  loaders:
    - name: target-postgres
      pip_url: 'git+https://github.com/meltano/target-postgres.git'
  transformers:
    - name: tap-github
      pip_url: 'https://github.com/nehiljain/dbt-tap-github.git'
orchestrators:
  - name: airflow
    pip_url: wtforms==2.2.1 apache-airflow==1.10.2
transformers:
  - name: dbt
    pip_url: dbt==0.16.1
files:
  - name: airflow
    pip_url: 'git+https://github.com/meltano/files-airflow.git'
schedules:
  - name: gitlab-to-postgres
    extractor: tap-github
    loader: target-postgres
    transform: skip
    interval: 'hourly'
    start_date: 2020-07-05 18:58:28.15924
version: 1
send_anonymous_usage_stats: true
project_id: 
plugins:
  extractors:
    - name: tap-github
      namespace: tap.github
      pip_url: 'git+https://github.com/nehiljain/tap-github.git'
      executable: tap-github
      capabilities:
        - discover
        - properties
      settings:
        - name: access_token
          env: TAP_GITHUB_ACCESS_TOKEN
        - name: repository
          env: TAP_GITHUB_REPOSITORY
  loaders:
    - name: target-postgres
      pip_url: 'git+https://github.com/meltano/target-postgres.git'
  transforms:
    - name: tap-github
      pip_url: 'https://github.com/nehiljain/dbt-tap-github.git'
  orchestrators:
    - name: airflow
      pip_url: wtforms==2.2.1 apache-airflow==1.10.2
  transformers:
    - name: dbt
      pip_url: dbt==0.16.1
  files:
    - name: airflow
      pip_url: 'git+https://github.com/meltano/files-airflow.git'
  schedules:
    - name: gitlab-to-postgres
      extractor: tap-github
      loader: target-postgres
      transform: skip
      interval: 'hourly'
      start_date: 2020-07-05 18:58:28.155924
Sit back & Relax
Some challenges out there

- Visualisation/BI layer
- Analytics code coverage
- Singer community
Key Takeaways

- Standardized tooling
- ELT >> ETL
- GE + Singer + DBT orchestrated by Airflow
Thanks
Resources

- Meltano Project
- Advanced Data Engineering Patterns with Apache Airflow by Maxime Beauchemin
- The Rise of the Data Engineer
- The Future of Data Engineering
- Downfall of the data engineer
Resources

- Singer | Open Source ETL
- Why we are building an open-source platform for ELT pipelines - Meltano
- Dbt Docs