

voi.

voi.com

How to increase developer productivity with dbt + Snowflake

Cities made
for living



Agenda

Introduction	01
What is dbt?	02
How we work with dbt + Snowflake	03





TABLE OF CONTENTS

A better future is possible.

This report aims to substantiate our company's vision statement and further articulate the end goal of the micromobility movement: reinventing public transport for a future of shared mobility.

This report is interactive

02 Problem: Car dependency

06 Vision: Freeing up space for living

- 6 Cities have a unique opportunity
- 9 A glimpse of the streets of the future
- 11 The repurposing of streets has already begun

20 Solution: Less is more with shared mobility

- 21 Moving people to the top of the sustainable mobility diet
- 22 Digitalisation is transforming mobility and enabling sharing
- 25 Car freedom through e-commerce

27 Scaling systems: Building out the public transport network

- 28 Bringing new mobility modes to the public transport network
- 31 The success factors for integrating with shared micromobility
- 32 Cases and learnings from different European cities

36 Transparency: Being the change you want to see

- 38 Tracking our negative environmental impact
- 39 Working towards having a fully circular service
- 40 Design and repairs to extend the vehicle lifespan
- 45 The climate impact of each trip is decreasing
- 47 A diverse and growing team of changemakers



Sveavägen. Voi was born in 2018 on Sveavägen, Stockholm, out of a group of young peoples' vision for helping build cities made for living. We use the latest technology to enable access to safe, sustainable and reliable micromobility for everyone.

VISION: FREEING UP SPACE FOR LIVING

Envision the future of your city, without old habits. What if we went back to the drawing board, what would it look like?



+6

MILLION USERS

+100

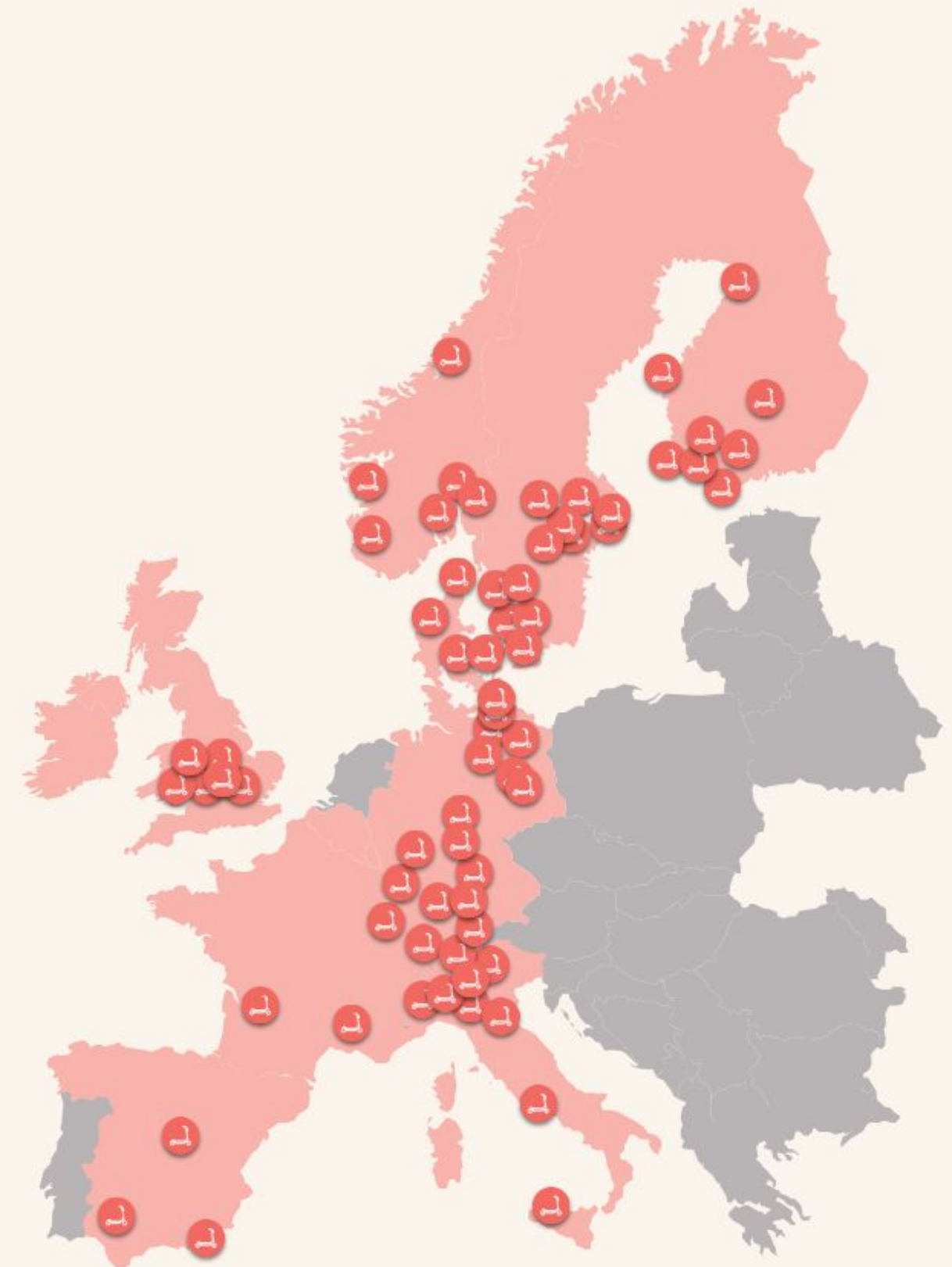
MILLION JOYFUL RIDES

+80

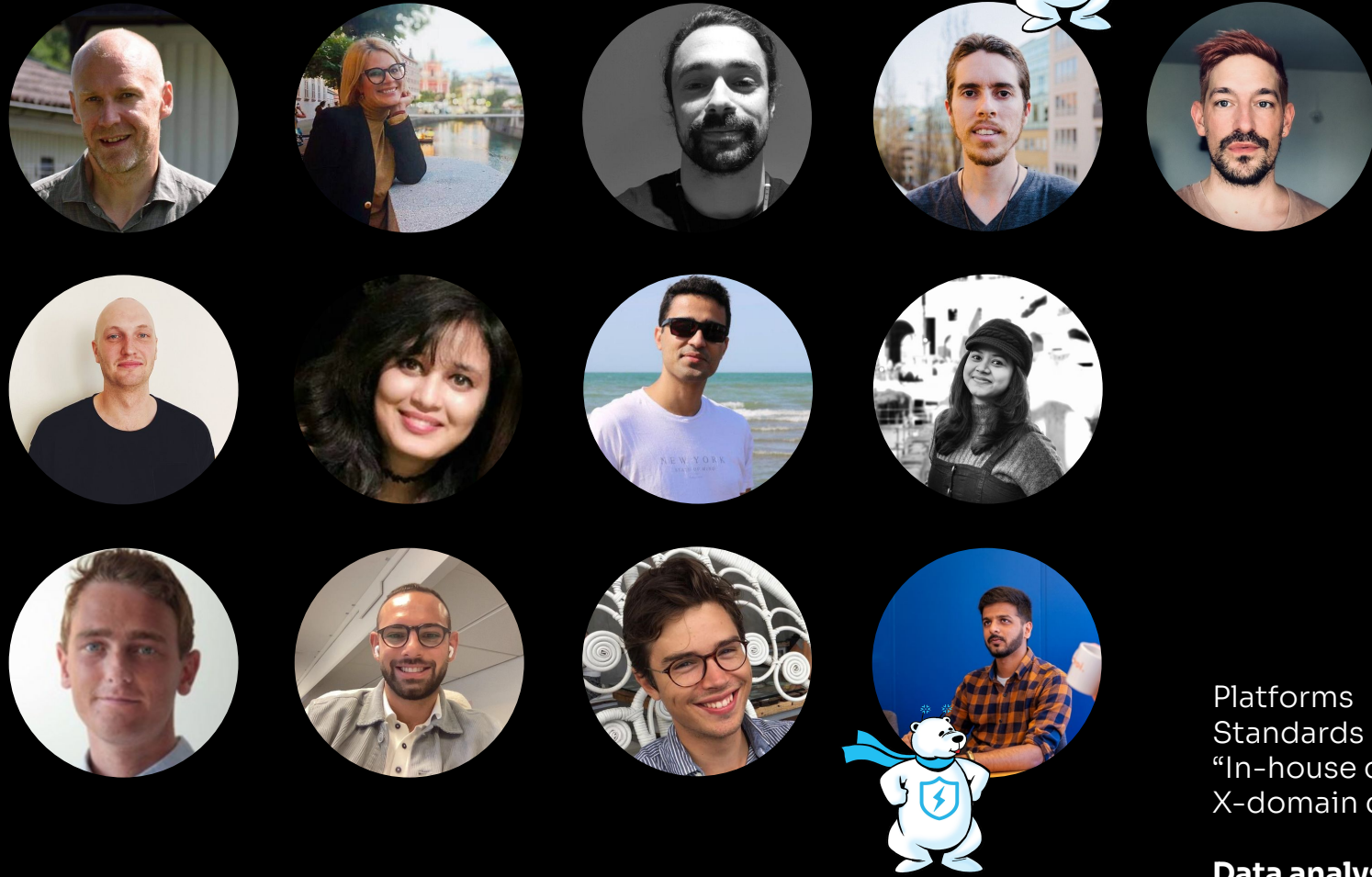
CITIES ACROSS 11 COUNTRIES

+10000

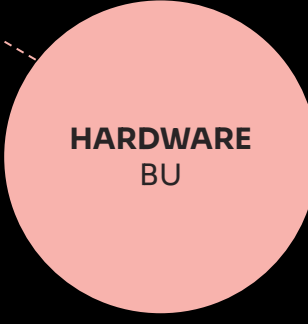
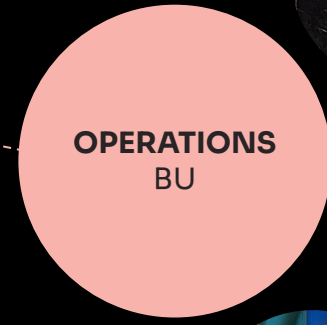
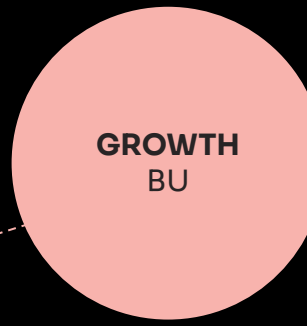
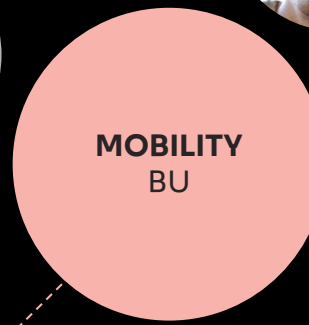
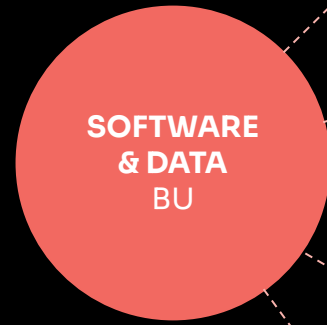
EMPLOYEES



DATA PLATFORM & DATA ANALYTICS



ACCELERATED MACHINE LEARNING



Platforms
Standards
"In-house consulting"
X-domain data products

Data analysts & data scientists
Data engineers & ML engineers

Domain expertise
Domain data products

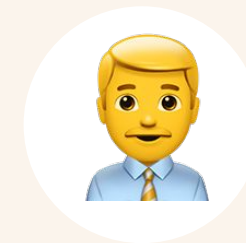
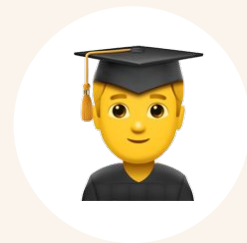
Data analysts & data scientists
Data engineers & ML engineers

BUSINESS CONTROL

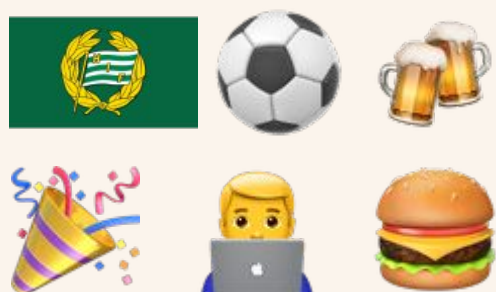
HARDWARE BU

OPERATIONS BU

Kevin “KevTheDev, Kevin nr 2, tall Kevin, Swedish Kevin” Neville




Personal



Educational

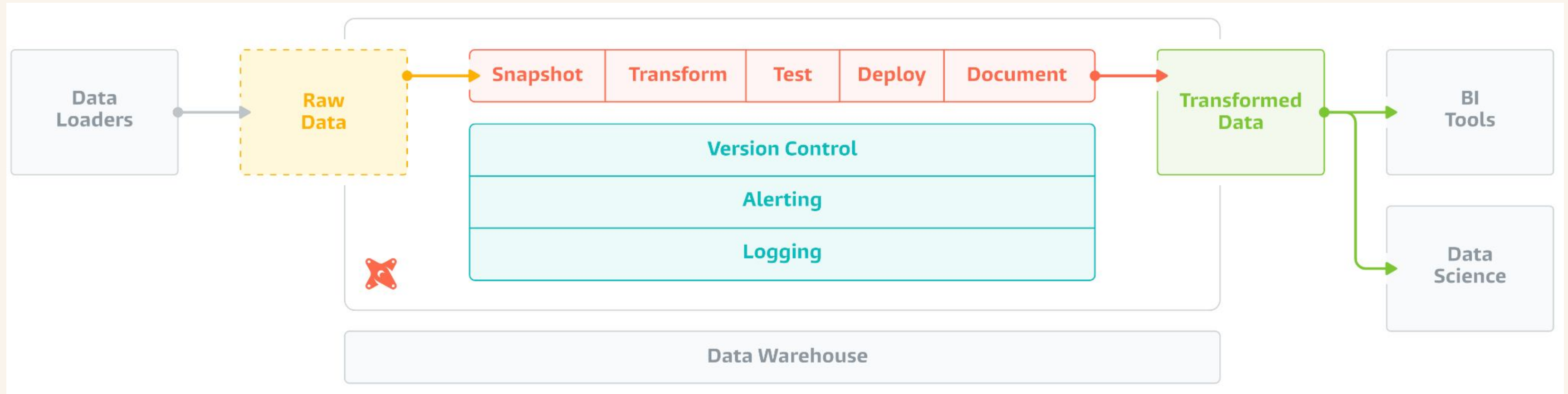
- Master of Science in Statistics and machine learning

Professional

- Voi, Staff data engineer
- Klarna, Software engineer + various data roles
- 

Introduction to dbt





[Introduction to dbt \(data build tool\) from Fishtown Analytics](#)


```

1  {{
2    config(
3      schema='vehicle',
4      materialized='incremental',
5      cluster_by=['date(received_at)'],
6      meta={'owner': 'kevin.neville@voiapp.io', 'team_owner': 'data-platform'},
7      tags=['hourly']
8    )
9  }}
10
11 {% if target.name == 'prod' %}
12   {% set start_date = "'2022-02-15'" %}
13 {% else %}
14   {% set start_date = "dateadd('day', -1, current_timestamp)" %}
15 {%- endif %}
16
17 select
18   vehicle_id
19 from {{ ref('vehicles') }}
20 where vehicle_id is not null and received_at > {{ start_date }}
21 {% if is_incremental() %}
22   and received_at > (select max(received_at) from {{ this }} )
23 {% endif %}

```

```

1  select
2    vehicle_id
3  from model.vehicle.vehicle
4  where vehicle_id is not null and received_at > dateadd('day', -1, current_timestamp)

```

```

1  select
2    vehicle_id,
3  from model.vehicle.vehicle
4  where vehicle_id is not null and received_at >= '2022-02-15'
5     and received_at > (select max(received_at) from model.vehicle.vehicle_presentation)

```

```

1  version: 2
2
3  models:
4    - name: vehicle_notification
5      description: 'Raw notifications from the vehicles'
6
7      columns:
8        - name: vehicle_id
9          description: 'Imei of the vehicle'
10         tests:
11           - not_null

```


dbt Core:

Open-source data transformation

- Open Source: Apache 2.0
- Includes core SQL compilation logic, Jinja templating, database adapters
- Interface via the CLI

dbt Cloud:

A fully-managed dbt experience

- Managed: SaaS
- Full IDE to develop and test your dbt code
- Orchestrate your jobs
- Logging and Alerting
- Integrated documentation
- User auth / SSO

[Introduction to dbt \(data build tool\) from Fishtown Analytics](#)

- Project configs >
- Adapter-specific configs >
- Postgres configurations
- BigQuery configurations
- Redshift configurations
- Snowflake configurations**
- SingleStore configurations
- Apache Spark configurations
- Materialize configurations
- Firebolt configurations
- Teradata configurations
- ClickHouse configurations
- MindsDB configurations
- Microsoft SQL Server configurations
- Microsoft Azure Synapse DWH configurations
- Greenplum configurations
- Apache Impala configurations
- Resource configs and properties >
- Commands >
- Jinja Reference >
- Profiles (CLI only) >
- dbt Artifacts >

Adapter-specific configs > Snowflake configurations

Snowflake configurations

Transient tables

Snowflake supports the creation of **transient tables**. Snowflake does not preserve a history for these tables, which can result in a measurable reduction of your Snowflake storage costs. Note however that transient tables do not participate in Time Travel. Weigh these tradeoffs when deciding whether or not to configure your dbt models as `transient`. **By default, all Snowflake tables created by dbt are transient.**

Configuring transient tables in dbt_project.yml

A whole folder (or package) can be configured to be transient (or not) by adding a line to the `dbt_project.yml` file. This config works just like all of the **model configs** defined in `dbt_project.yml`.

```
dbt_project.yml
name: my_project
...
models:
  +transient: false
  my_project:
    ...
```

Configuring transience for a specific model

A specific model can be configured to be transient by setting the `transient` model config to `true`.

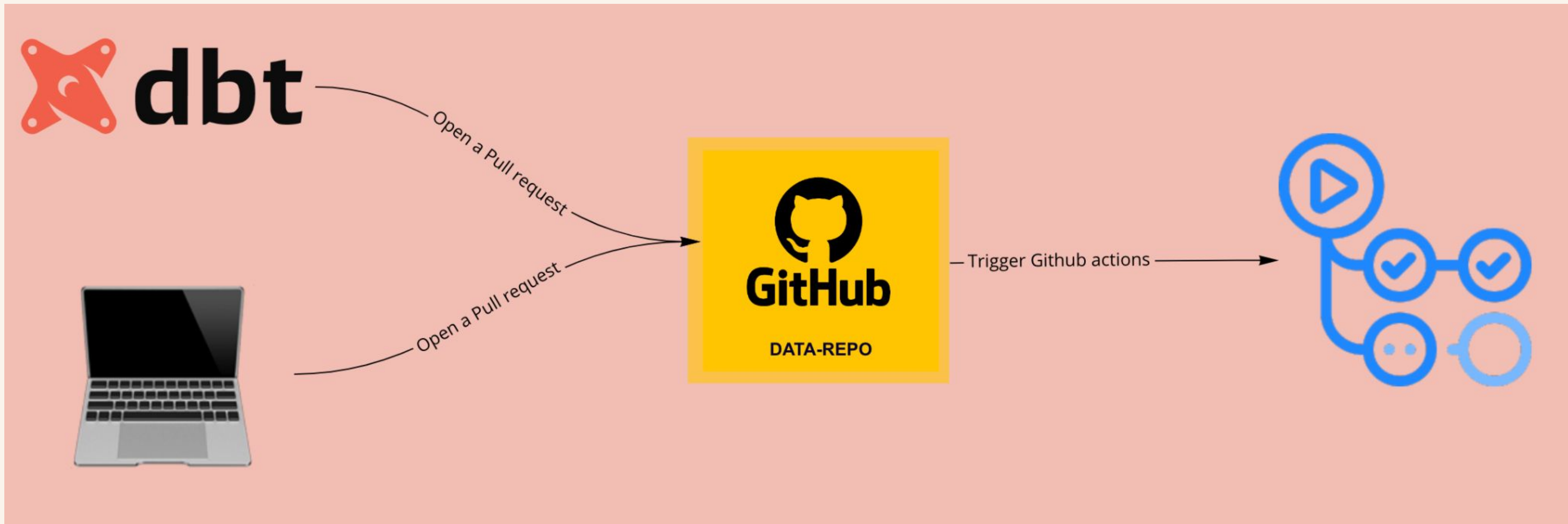
```
my_table.sql
{{ config(materialized='table', transient=true) }}
```

- Transient tables
 - Configuring transient tables in dbt_project.yml
 - Configuring transience for a specific model
- Query tags
- Merge behavior (incremental models)
- Configuring table clustering
 - Using cluster_by
 - Automatic clustering
- Configuring virtual warehouses
- Copying grants
- Secure views

dbt + Snowflake at Voi



Hybrid usage between cloud and CLI



Automatic checks

- Check meta fields of each model

```
Check ownership meta keys on models..... Failed
transform/snowflake-dbt/models/model/vehicle/vehicle_notification.sql: does not have some of the meta keys defined:
- owner
- team_owner
```

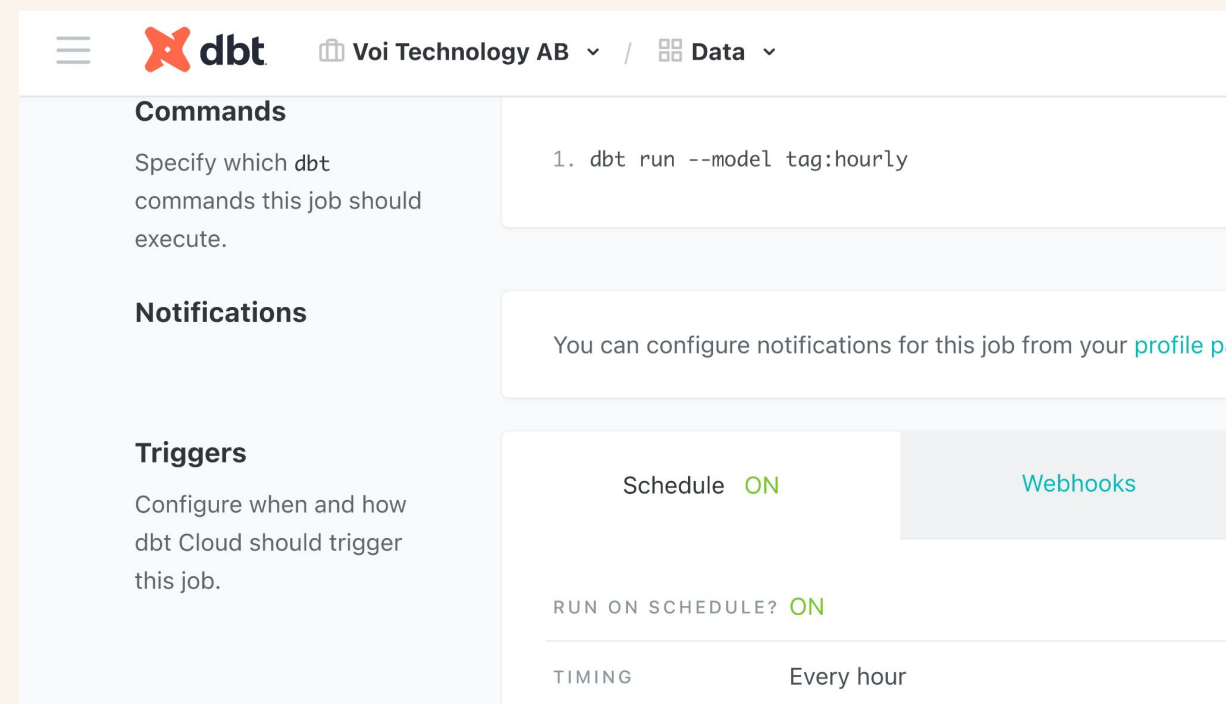
- Lint sql using sqlfluff

```
== [transform/snowflake-dbt/models/model/vehicle/vehicle_notification.sql] FAIL
L: 19 | P: 1 | L010 | Keywords must be lower case.
All Finished 📦 🎉!
```

- Run a test with the new changes in a dev database
- Code review

Easily deploy jobs using tags

- Hourly
- Daily
- Weekly
- ...



```

1  {{
2      config(
3          schema='vehicle',
4          materialized='incremental',
5          cluster_by=['date(received_at)'],
6          meta={'owner': 'kevin.neville@voiapp.io', 'team_owner': 'data-platform'},
7          tags=['hourly']
8      )
9  }}
10
11  {% if target.name == 'prod' %}
12      {% set start_date = "'2022-02-15'" %}
13  {% else %}
14      {% set start_date = "dateadd('day', -1, current_timestamp)" %}
15  {%- endif %}
16
17  select
18      vehicle_id
19  from {{ ref('vehicles') }}
20  where vehicle_id is not null and received_at > {{ start_date }}
21  {% if is_incremental() %}
22      and received_at > (select max(received_at) from {{ this }} )
23  {% endif %}

```

Data Quality Checks

Click [here](#) to learn how to interpret it.

Freshness

- **Dataset freshness:** **pass** (checked at: 2022-06-26, 19:41:01 UTC)
- **Total upstream sources:** 54 (checked at: 2022-06-26, 19:16:50 UTC)
 - **Freshness check passed:** **46**
 - **Freshness check failed:** **0**
 - **Sources with no freshness check:** 8 (source.voi_snowflake.public.poi_data_osm, source.voi_snowflake.stripe_portugal.charges, source.voi_snowflake.dbexp.vehicle_models, source.voi_snowflake.dbexp.vehicle, source.voi_snowflake.stripe_portugal.events, source.voi_snowflake.dbexp.account, source.voi_snowflake.dbexp.account_history, source.voi_snowflake.bigquery_archive.vehicle_v1)

Documentation

- **Documentation coverage:** **High** (checked at: 2022-06-26, 20:27:57 UTC)
- **Ratio of columns that have description:** **102 / 102**

Tests

- **Ratio of tests passing:** **15 / 15** (checked at: 2022-06-26, 19:41:01 UTC)
- **Failed tests:** **0**
- **Ratio of columns that have test:** **13 / 102**




```

1  {% macro set_query_tag() -%}
2  {% set new_query_tag = model.name %}
3  {% if new_query_tag %}
4  {% set original_query_tag = get_current_query_tag() %}
5  {{ log("Setting query_tag to '" ~ complete_tag ~ "'. Will reset to '" ~ original_query_tag ~ "' after materialization.") }}
6  {% do run_query("alter session set query_tag = 'dbt_model: {}'.format(new_query_tag)) %}
7  {{ return(original_query_tag)}}
8  {% endif %}
9  {{ return(none)}}
10 {% endmacro %}

```

Query History

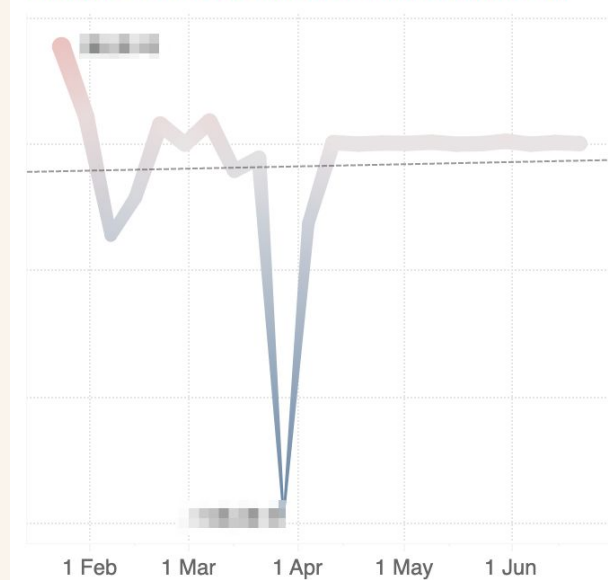
56 Queries

Status All User All Filters 1 Columns

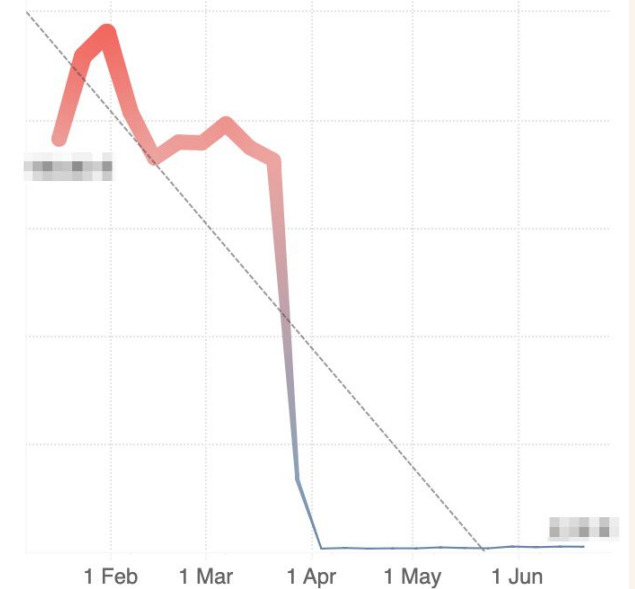
SQL TEXT	QUERY ID	STARTED	SESSION ID	QUERY TAG
commit;	01a53832-3201-338b-00...	6/27/2022, 6:02 AM	24657605191238	dbt_model: vehicle_notification
insert into model.vehicle.vehicle_notifcat...	01a53831-3201-338b-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
begin;	01a53831-3201-3293-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
describe table "MODEL"."VEHICLE"."VEHICLE_...	01a53831-3201-3293-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
describe table model.vehicle.vehicle_notif...	01a53831-3201-3293-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
describe table model.vehicle.vehicle_notif...	01a53831-3201-338b-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
create or replace temporary table model.ve...	01a53831-3201-3293-00...	6/27/2022, 6:01 AM	24657605191238	dbt_model: vehicle_notification
commit;	01a537f6-3201-3293-00...	6/27/2022, 5:02 AM	24657605177890	dbt_model: vehicle_notification
insert into model.vehicle.vehicle_notifcat...	01a537f5-3201-338b-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification
begin;	01a537f5-3201-338b-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification
describe table "MODEL"."VEHICLE"."VEHICLE_...	01a537f5-3201-338b-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification
describe table model.vehicle.vehicle_notif...	01a537f5-3201-3293-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification
describe table model.vehicle.vehicle_notif...	01a537f5-3201-338b-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification
create or replace temporary table model.ve...	01a537f5-3201-3293-000...	6/27/2022, 5:01 AM	24657605177890	dbt_model: vehicle_notification

daily cost changes

weekly resolution does not show data from current week



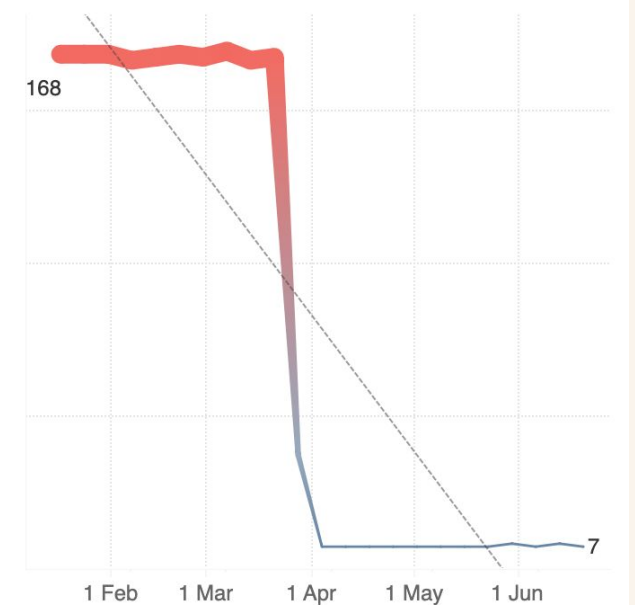
daily costs



average execution time (s)



number daily executions

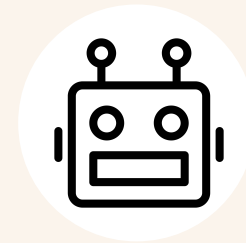


And more!



Tableau exposures

Connect BI tools to lineage graph.



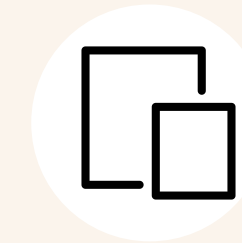
Custom tests and macros

Custom dbt test and macros to improve productivity.



Using different databases in the same dbt project

By default, all dbt models are built in the schema specified in your target.



Zero-copy cloning for dev environments

Use zero copy cloning to test new models in dev environment.

Summary



- Introduced to dbt on Snowflake
 - CLI vs. dbt cloud
 - Dbt integrated with Snowflake
- Voi usage of dbt
 - Improved productivity
 - Governance of platform
 - And more!

Thank you!