

dbt Intro

Alexander Matrunich
Data analyst
HomeBuddy.com

PGUG.EE
December 2025
meetup

About speaker

Degree in Sociology (Pskov Volny Institute) ->

Surveys and opinion polls, analytics with SPSS ->

Interest to open source ->

Linux, R ->

From surveys to pure data analytics ->

Contracts with the UN FAO in Rome (R, Oracle) ->

Data Analyst, Taxify/Bolt (Redshift, Looker, R) ->

Data Analyst, HomeBuddy (Snowflake, dbt, Superset, Thoughtspot)

About speaker

Degree in Sociology (Pskov Volny Institute) ->

Surveys and opinion polls, analytics with SPSS ->

Interest to open source ->

Linux, R ->

From surveys to pure data analytics ->

Contracts with the UN FAO in Rome (R, Oracle) ->

Data Analyst, Taxify/Bolt (Redshift, Looker, R) ->

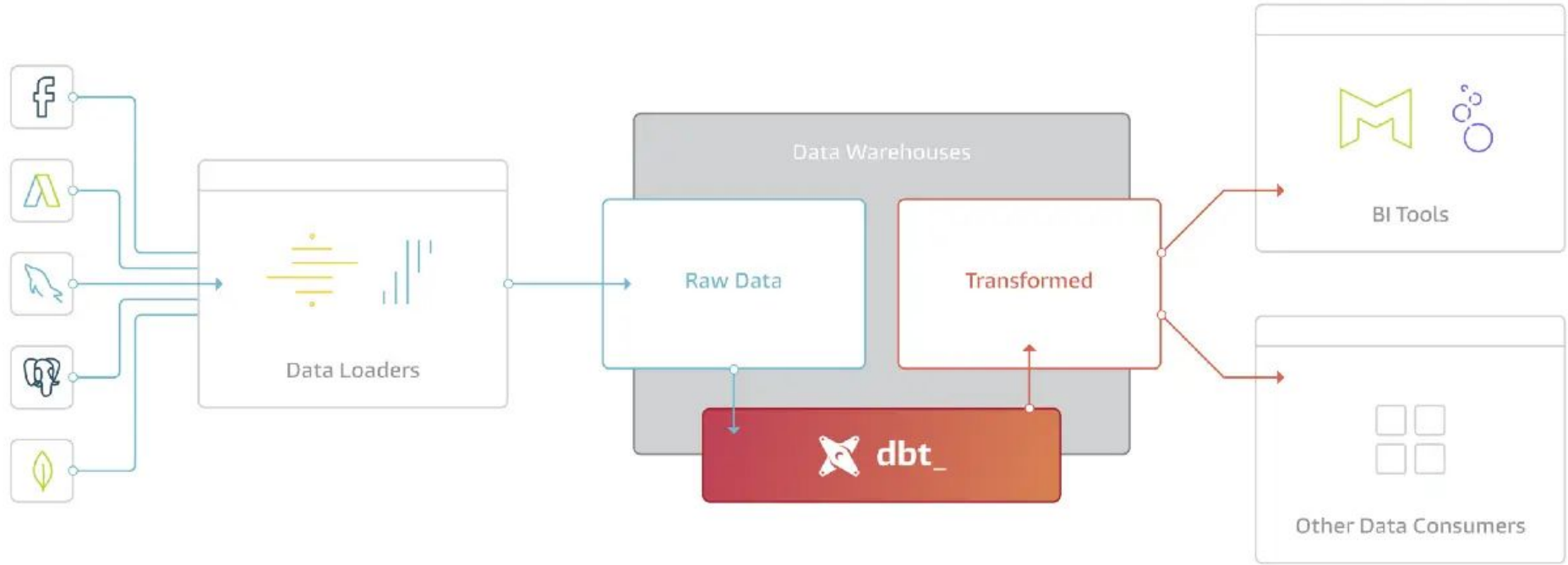
Data Analyst, HomeBuddy (Snowflake, dbt, Superset, Thoughtspot)

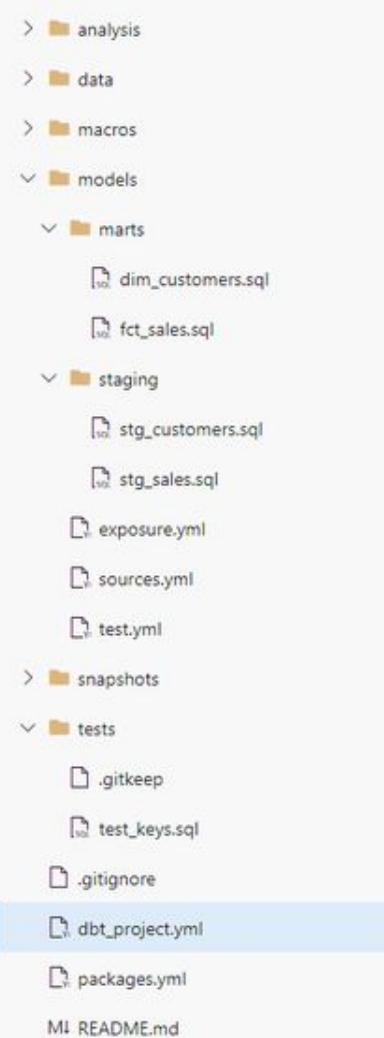
Other interests:

Paragliding (no engine/propeller!)

Family history research

dbt and the modern BI stack





A dbt project

A dbt model

```
1  {{
2    config(
3      unique_key="BK_ORDER_NUMBER||'-'||BK_ORDER_LINE_NUMBER",
4      file_format='delta'
5    )
6  }}
7
8  with stg_sales as (
9    select * from {{ref('stg_sales')}}
10 ),
11
12 sales as (
13   SELECT
14     BK_ORDER_NUMBER,
15     BK_ORDER_LINE_NUMBER,
16     to_timestamp(OrderDateKey, 'yyyyMMdd') as ORDER_DATE,
17     M_ORDER_QUANTITY,
18     SHA1(ResellerKey) as A_HASHED_KEY_BK
19   FROM stg_sales
20 )
21
22 select * from sales
23
24 {% if is_incremental() %}
25
26   -- this filter will only be applied on an incremental run
27   where ORDER_DATE >= (select date_add(max(ORDER_DATE), -3) from {{ this }})
28
29 {% endif %}
```

A dbt model

```
1  {{
2    config(
3      unique_key="BK_ORDER_NUMBER||'-'||BK_ORDER_LINE_NUMBER",
4      file_format='delta'
5    )
6  }}
7
8  with stg_sales as (
9    select * from {{ref('stg_sales')}}
10 ),
11
12 sales as (
13   SELECT
14     BK_ORDER_NUMBER,
15     BK_ORDER_LINE_NUMBER,
16     to_timestamp(OrderDateKey, 'yyyyMMdd') as ORDER_DATE,
17     M_ORDER_QUANTITY,
18     SHA1(ResellerKey) as A_HASHED_KEY_BK
19   FROM stg_sales
20 )
21
22 select * from sales
23
24 {% if is_incremental() %}
25
26   -- this filter will only be applied on an incremental run
27   where ORDER_DATE >= (select date_add(max(ORDER_DATE), -3) from {{ this }})
28
29 {% endif %}
```

```
1  version: 2
2
3  models:
4    - name: stg_customers
5      columns:
6        - name: BK_CUSTOMER_CODE
7          tests:
8            - unique
9            - not_null
10   - name: dim_customers
11     columns:
12       - name: BK_CUSTOMER_CODE
13         tests:
14           - unique
15           - not_null
```

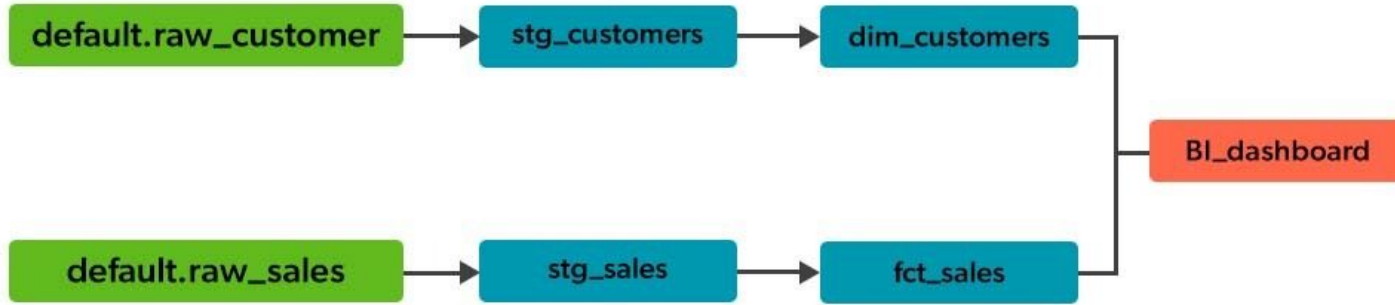
```
$ cat -n models/orders_rollup.sql
 1  with
 2    orders as ( select * from {{ref('orders')}} )
 3    , line_items as ( select * from {{ref('line_items')}} )
 4
 5  select
 6    id
 7    , sum(line_items.purchase_price)
 8
 9  from orders
10 left join line_items
11   on orders.id = line_items.order_id
12
13 group by 1

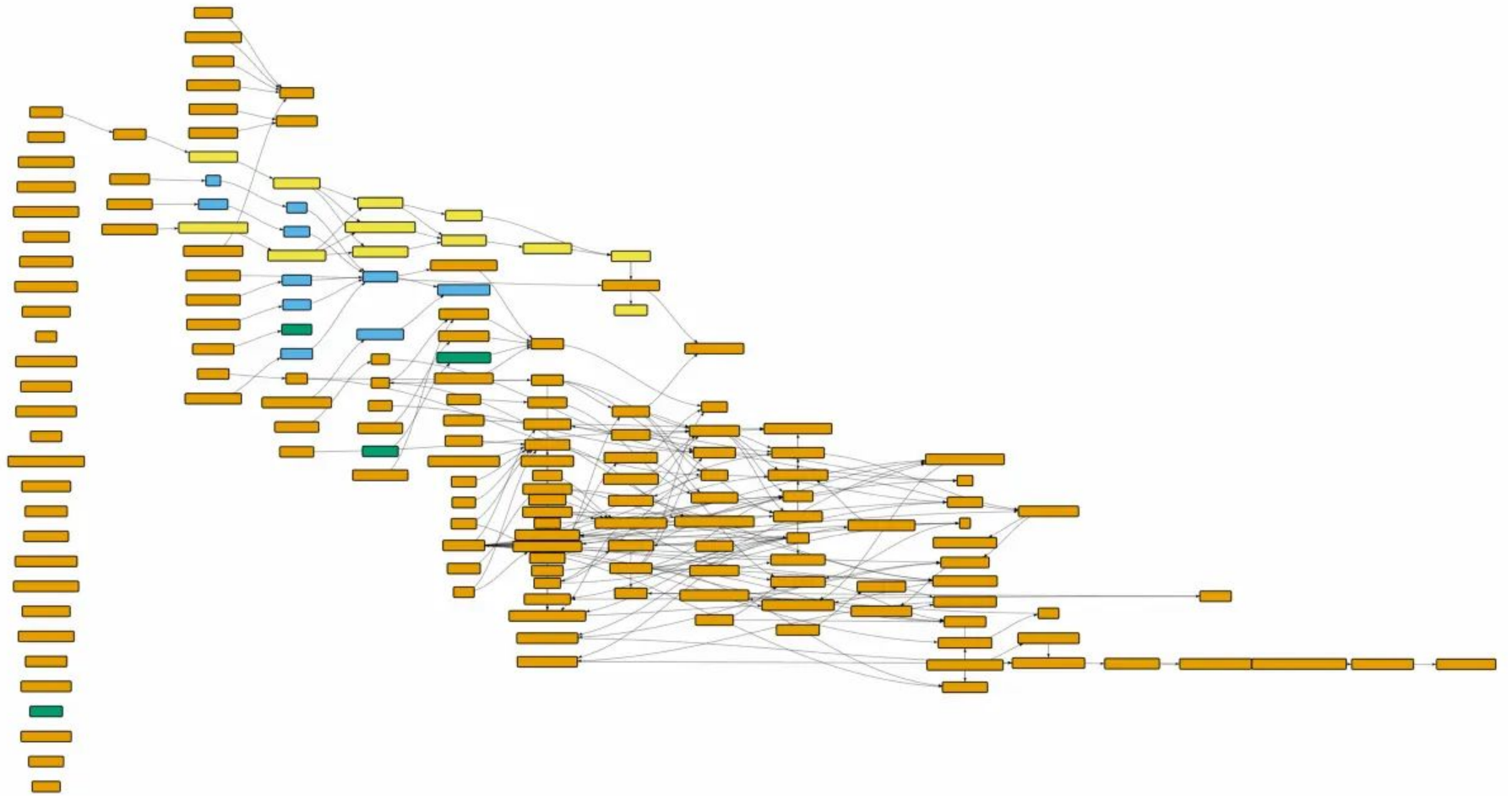
$ dbt run
Found 3 models, 0 tests, 0 archives, 0 analyses, 10 macros, 0 operations
```

```
09:44:39 | Concurrency: 4 threads (target='postgres')
09:44:39 |
09:44:39 | 1 of 3 START view model ecom.orders..... [RUN]
09:44:39 | 2 of 3 START view model ecom.line_items..... [RUN]
09:44:39 | 1 of 3 OK created view model ecom.orders..... [CREATE VIEW in 0.07s]
09:44:39 | 2 of 3 OK created view model ecom.line_items..... [CREATE VIEW in 0.07s]
09:44:39 | 3 of 3 START view model ecom.orders_rollup..... [RUN]
09:44:39 | 3 of 3 OK created view model ecom.orders_rollup..... [CREATE VIEW in 0.05s]
09:44:39 |
09:44:39 | Finished running 3 view models in 0.43s.
```

Completed successfully

DAG (Directed Acyclic Graph)





Jinja, a templating language

```
select * from {{ref('really_big_table')}}
{%raw%}
{% if incremental and target.schema == 'prod' %}
    where timestamp >= (select max(timestamp) from {{this}})
{% else %}
    where timestamp >= dateadd(day, -3, current_date)
{% endif %}
```

[📖 README](#) [👤 Contributing](#) [📄 Apache-2.0 license](#)

- [Installation Instructions](#)
- [Generic Tests](#)
 - [equal_rowcount \(source\)](#)
 - [fewer_rows_than \(source\)](#)
 - [equality \(source\)](#)
 - [expression_is_true \(source\)](#)
 - [recency \(source\)](#)
 - [at_least_one \(source\)](#)
 - [not_constant \(source\)](#)
 - [not_empty_string \(source\)](#)
 - [cardinality_equality \(source\)](#)
 - [not_null_proportion \(source\)](#)
 - [not_accepted_values \(source\)](#)
 - [relationships_where \(source\)](#)
 - [mutually_exclusive_ranges \(source\)](#)
 - [sequential_values \(source\)](#)
 - [unique_combination_of_columns \(source\)](#)
 - [accepted_range \(source\)](#)
 - [Grouping in tests](#)

- [Macros](#)
- [Introspective macros](#)
 - [get_column_values \(source\)](#)
 - [get_filtered_columns_in_relation \(source\)](#)
 - [get_relations_by_pattern \(source\)](#)
 - [get_relations_by_prefix \(source\)](#)
 - [get_query_results_as_dict \(source\)](#)
 - [get_single_value \(source\)](#)
- [SQL generators](#)
 - [date_spine \(source\)](#)
 - [deduplicate \(source\)](#)
 - [haversine_distance \(source\)](#)
 - [group_by \(source\)](#)
 - [star \(source\)](#)
 - [union_relations \(source\)](#)
 - [generate_series \(source\)](#)
 - [generate_surrogate_key \(source\)](#)
 - [safe_add \(source\)](#)
 - [safe_divide \(source\)](#)
 - [safe_subtract \(source\)](#)
 - [pivot \(source\)](#)
 - [unpivot \(source\)](#)
 - [width_bucket \(source\)](#)
- [Web macros](#)

dbt alternatives?

dbt

Software Development practices for Analytics:

- Source control
- Tests
- Documentation
- Development environment