

Московский городской педагогический университет

Платформы Data Engineering '25

Лабораторная работа 2.1. Построение аналитических витрин  
и внедрение продвинутых dbt-концепций

Выполнил: Ежергин С.С.  
учебная группа - 251м

Проверил: доцент департамента информатики,  
управления и технологий  
Босенко Т.М.

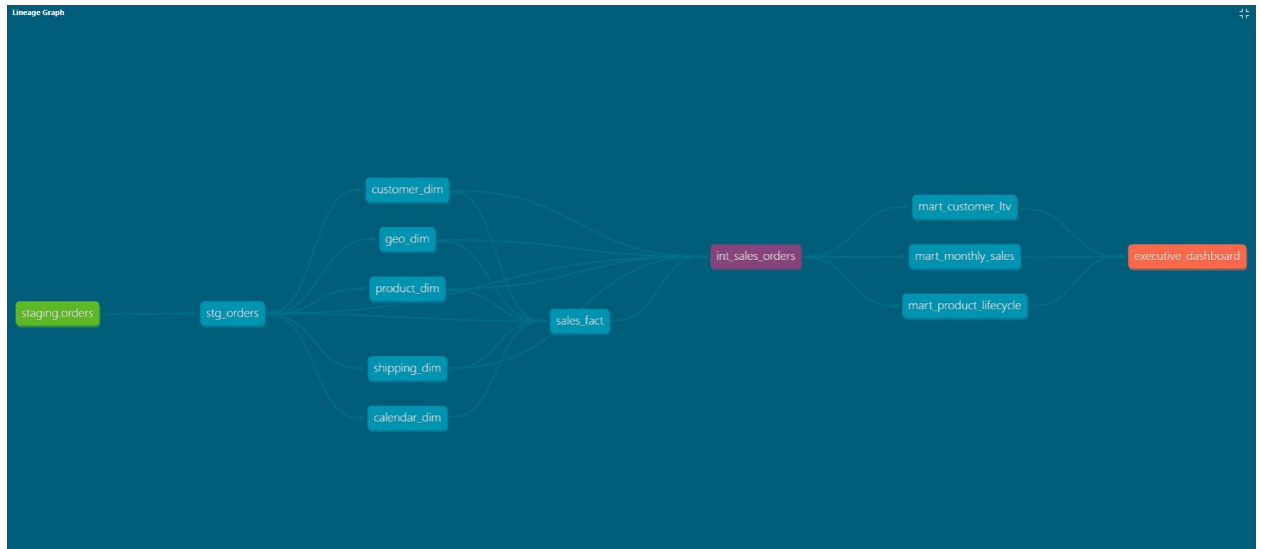
Москва - 2025

## Архитектура проекта

Проект реализует многоуровневую архитектуру DWH с использованием dbt:

- Staging-слой - очистка и подготовка исходных данных
- Intermediate-слой - денормализованные модели с инкапсулированной бизнес-логикой
- Marts-слой - специализированные аналитические витрины для бизнес-пользователей
- Snapshots - отслеживание исторических изменений данных

## Скриншот графа зависимостей



## Промежуточная модель (int\_sales\_orders)

- Создана в папке: models/intermediate/
- Материализована как: view
- В схеме: dw\_intermediate
- Содержит: денормализованные данные (объединение фактов + измерений)

## Витрины используют промежуточную модель

- mart\_monthly\_sales → {{ ref('int\_sales\_orders') }}
- mart\_customer\_ltv → {{ ref('int\_sales\_orders') }}
- mart\_product\_lifecycle → {{ ref('int\_sales\_orders') }}

## Ключевые фрагменты кода

### 1. Промежуточная модель int\_sales\_orders.sql

```
{{ config(materialized='view') }}

SELECT

    -- Ключи
    f.order_id,

    -- Измерения из customer_dim
    c.customer_id,
    c.customer_name,
    c.segment as customer_segment,

    -- Измерения из product_dim
    p.product_id,
    p.product_name,
    p.category,
    p.subcategory,

    -- Измерения из geo_dim
    g.city,
    g.state,
    g.region,

    -- Измерения из shipping_dim
    s.ship_mode,

    -- Даты
    o.order_date,
    o.ship_date,

    -- Метрики из sales_fact
    f.sales,
    f.profit,
    f.quantity,
    f.discount
```

```

FROM {{ ref('sales_fact') }} AS f
LEFT JOIN {{ ref('stg_orders') }} AS o ON f.order_id = o.order_id
LEFT JOIN {{ ref('customer_dim') }} AS c ON f.customer_id_key =
c.customer_id_key
LEFT JOIN {{ ref('product_dim') }} AS p ON f.product_id_key =
p.product_id_key
LEFT JOIN {{ ref('shipping_dim') }} AS s ON f.shipping_id = s.shipping_id
LEFT JOIN {{ ref('geo_dim') }} AS g ON f.geo_id = g.geo_id

```

## 2. Индивидуальная mart-модель mart\_product\_lifecycle.sql

```

{{ config(materialized='table') }}

SELECT
    product_id,
    product_name,
    category,
    subcategory,
    MIN(order_date) as first_sale_date,
    MAX(order_date) as last_sale_date,
    (MAX(order_date) - MIN(order_date)) as days_between_sales,
    COUNT(DISTINCT order_id) as total_orders,
    SUM(quantity) as total_quantity_sold,
    SUM(sales) as total_sales
FROM {{ ref('int_sales_orders') }}
GROUP BY product_id, product_name, category, subcategory
ORDER BY days_between_sales DESC

```

## 3. Кастомный тест test\_is\_positive.sql

```

{% test is_positive(model, column_name) %}

SELECT *

```

```
FROM {{ model }}  
WHERE {{ column_name }} < 0  
{% endtest %}
```

#### 4. Применение кастомного теста в schema.yml

```
# models/marts/schema.yml  
- name: mart_product_lifecycle  
  description: "Анализ жизненного цикла продуктов"  
  columns:  
    - name: total_quantity_sold  
      tests:  
        - not_null  
        - is_positive  
    - name: total_sales  
      tests:  
        - not_null  
        - is_positive  
    - name: days_between_sales  
      tests:  
        - not_null  
        - is_positive
```

#### 5. Снимок данных snapshot\_product\_dim.sql

```
{% snapshot snapshot_product_dim %}  
{{  
  config(  
    target_schema='dw_snapshots',  
    strategy='check',  
    unique_key='product_id_key',  
    check_cols=['category', 'subcategory'],
```

```
)  
}}  
SELECT product_id_key, product_id, category, subcategory  
FROM {{ ref('product_dim') }}  
{% endsnapshot %}
```

#### 6. models\marts\exposures.yml:

```
version: 2  
  
exposures:  
  - name: executive_dashboard  
    type: dashboard  
    maturity: high  
    owner:  
      name: "Sales Department"  
      email: "sales@superstore.com"  
    depends_on:  
      - ref('mart_monthly_sales')  
      - ref('mart_customer_ltv')  
      - ref('mart_product_lifecycle')
```

## Результаты

Скриншот выполнения dbt run

```

(dbt-env) C:\pde_magistr\superstore_dwh>dbt run --select int_sales_orders
15:15:15 Running with dbt=1.10.15
15:15:15 Registered adapter: postgres=1.9.1
15:15:16 Unable to do partial parsing because a project config has changed
15:15:17 [WARNING][MissingArgumentsPropertyInGenericTestDeprecation]: Deprecated
functionality
Found top-level arguments to test `relationships`. Arguments to generic tests
should be nested under the `arguments` property.
15:15:17 Found 10 models, 22 data tests, 1 source, 448 macros
15:15:17
15:15:17 Concurrency: 1 threads (target='dev')
15:15:17
15:15:18 1 of 1 START sql view model dw_intermediate.int_sales_orders ..... [RUN]
15:15:18 1 of 1 OK created sql view model dw_intermediate.int_sales_orders ..... [CREATE VIEW in 0.16s]
15:15:18
15:15:18 Finished running 1 view model in 0 hours 0 minutes and 1.00 seconds (1.00s).
15:15:18
15:15:18 Completed successfully
15:15:18
15:15:18 Done. PASS=1 WARN=0 ERROR=0 SKIP=0 NO-OP=0 TOTAL=1
15:15:18 [WARNING][DeprecationsSummary]: Deprecated functionality
Summary of encountered deprecations:
- MissingArgumentsPropertyInGenericTestDeprecation: 3 occurrences
To see all deprecation instances instead of just the first occurrence of each,
run command again with the `--show-all-deprecations` flag. You may also need to
run with `--no-partial-parse` as some deprecations are only encountered during
parsing.

(dbt-env) C:\pde_magistr\superstore_dwh>dbt run --select mart_monthly_sales mart_customer_ltv mart_product_lifecycle
15:17:00 Running with dbt=1.10.15
15:17:00 Registered adapter: postgres=1.9.1
15:17:01 Found 13 models, 22 data tests, 1 source, 448 macros
15:17:01
15:17:01 Concurrency: 1 threads (target='dev')
15:17:01
15:17:02 1 of 3 START sql table model dw_test.mart_customer_ltv ..... [RUN]
15:17:02 1 of 3 OK created sql table model dw_test.mart_customer_ltv ..... [SELECT 793 in 0.28s]
15:17:02 2 of 3 START sql table model dw_test.mart_monthly_sales ..... [RUN]
15:17:02 2 of 3 OK created sql table model dw_test.mart_monthly_sales ..... [SELECT 426 in 0.17s]
15:17:02 3 of 3 START sql table model dw_test.mart_product_lifecycle ..... [RUN]
15:17:03 3 of 3 OK created sql table model dw_test.mart_product_lifecycle ..... [SELECT 1894 in 0.27s]
15:17:03
15:17:03 Finished running 3 table models in 0 hours 0 minutes and 1.43 seconds (1.43s).
15:17:03
15:17:03 Completed successfully
15:17:03
15:17:03 Done. PASS=3 WARN=0 ERROR=0 SKIP=0 NO-OP=0 TOTAL=3

```

Скриншот выполнения dbt test



```

(dbt-env) C:\pde_magistr\superstore_dwh>dbt test
15:23:04 Running with dbt=1.10.15
15:23:04 Registered adapter: postgres=1.9.1
15:23:05 Found 11 models, 31 data tests, 1 snapshot, 1 source, 1 exposure, 450 macros
15:23:05
15:23:05 Concurrency: 1 threads (target='dev')
15:23:05
15:23:06 1 of 31 START test is_non_negative_interval_mart_product_lifecycle_days_between_sales [RUN]
15:23:06 1 of 31 PASS is_non_negative_interval_mart_product_lifecycle_days_between_sales [PASS in 0.22s]
15:23:06 2 of 31 START test is_positive_mart_customer_ltv_number_of_orders ..... [RUN]
15:23:06 2 of 31 PASS is_positive_mart_customer_ltv_number_of_orders ..... [PASS in 0.18s]
15:23:06 3 of 31 START test is_positive_mart_customer_ltv_total_sales_lifetime ..... [RUN]
15:23:06 3 of 31 PASS is_positive_mart_customer_ltv_total_sales_lifetime ..... [PASS in 0.18s]
15:23:06 4 of 31 START test is_positive_mart_monthly_sales_number_of_orders ..... [RUN]
15:23:06 4 of 31 PASS is_positive_mart_monthly_sales_number_of_orders ..... [PASS in 0.06s]
15:23:06 5 of 31 START test is_positive_mart_monthly_sales_total_sales ..... [RUN]
15:23:06 5 of 31 PASS is_positive_mart_monthly_sales_total_sales ..... [PASS in 0.17s]
15:23:06 6 of 31 START test is_positive_mart_product_lifecycle_total_orders ..... [RUN]
15:23:06 6 of 31 PASS is_positive_mart_product_lifecycle_total_orders ..... [PASS in 0.18s]
15:23:07 7 of 31 START test is_positive_mart_product_lifecycle_total_quantity_sold ..... [RUN]
15:23:07 7 of 31 PASS is_positive_mart_product_lifecycle_total_quantity_sold ..... [PASS in 0.17s]
15:23:07 8 of 31 START test is_positive_mart_product_lifecycle_total_sales ..... [RUN]
15:23:07 8 of 31 PASS is_positive_mart_product_lifecycle_total_sales ..... [PASS in 0.18s]
15:23:07 9 of 31 START test not_null_calendar_dim_date_key ..... [RUN]
15:23:07 9 of 31 PASS not_null_calendar_dim_date_key ..... [PASS in 0.07s]
15:23:07 10 of 31 START test not_null_customer_dim_customer_id ..... [RUN]
15:23:07 10 of 31 PASS not_null_customer_dim_customer_id ..... [PASS in 0.19s]
15:23:07 11 of 31 START test not_null_customer_dim_customer_id_key ..... [RUN]
15:23:07 11 of 31 PASS not_null_customer_dim_customer_id_key ..... [PASS in 0.18s]
15:23:07 12 of 31 START test not_null_geo_dim_geo_id ..... [RUN]
15:23:07 12 of 31 PASS not_null_geo_dim_geo_id ..... [PASS in 0.18s]
15:23:08 13 of 31 START test not_null_mart_customer_ltv_total_sales_lifetime ..... [RUN]
15:23:08 13 of 31 PASS not_null_mart_customer_ltv_total_sales_lifetime ..... [PASS in 0.17s]
15:23:08 14 of 31 START test not_null_mart_monthly_sales_total_sales ..... [RUN]
15:23:08 14 of 31 PASS not_null_mart_monthly_sales_total_sales ..... [PASS in 0.18s]
15:23:08 15 of 31 START test not_null_mart_product_lifecycle_days_between_sales ..... [RUN]
15:23:08 15 of 31 PASS not_null_mart_product_lifecycle_days_between_sales ..... [PASS in 0.06s]
15:23:08 16 of 31 START test not_null_mart_product_lifecycle_total_quantity_sold ..... [RUN]
15:23:08 16 of 31 PASS not_null_mart_product_lifecycle_total_quantity_sold ..... [PASS in 0.18s]
15:23:08 17 of 31 START test not_null_mart_product_lifecycle_total_sales ..... [RUN]
15:23:08 17 of 31 PASS not_null_mart_product_lifecycle_total_sales ..... [PASS in 0.19s]
15:23:08 18 of 31 START test not_null_product_dim_product_id_key ..... [RUN]
15:23:08 18 of 31 PASS not_null_product_dim_product_id_key ..... [PASS in 0.07s]
15:23:09 19 of 31 START test not_null_sales_fact_customer_id_key ..... [RUN]
15:23:09 19 of 31 PASS not_null_sales_fact_customer_id_key ..... [PASS in 0.07s]
15:23:09 20 of 31 START test not_null_sales_fact_geo_id ..... [RUN]
15:23:09 20 of 31 PASS not_null_sales_fact_geo_id ..... [PASS in 0.06s]
15:23:09 21 of 31 START test not_null_sales_fact_order_id ..... [RUN]
15:23:09 21 of 31 PASS not_null_sales_fact_order_id ..... [PASS in 0.06s]
15:23:09 22 of 31 START test not_null_sales_fact_product_id_key ..... [RUN]
15:23:09 22 of 31 PASS not_null_sales_fact_product_id_key ..... [PASS in 0.06s]
15:23:09 23 of 31 START test not_null_shipping_dim_shipping_id ..... [RUN]
15:23:09 23 of 31 PASS not_null_shipping_dim_shipping_id ..... [PASS in 0.05s]
15:23:09 24 of 31 START test relationships_sales_fact_customer_id_key_customer_id_key_ref_customer_dim_ [RUN]
15:23:09 24 of 31 PASS relationships_sales_fact_customer_id_key_customer_id_key_ref_customer_dim_ [PASS in 0.06s]
15:23:09 25 of 31 START test relationships_sales_fact_geo_id_geo_id_ref_geo_dim_ ..... [RUN]
15:23:09 25 of 31 PASS relationships_sales_fact_geo_id_geo_id_ref_geo_dim_ ..... [PASS in 0.19s]
15:23:09 26 of 31 START test relationships_sales_fact_product_id_key_product_id_key_ref_product_dim_ [RUN]
15:23:09 26 of 31 PASS relationships_sales_fact_product_id_key_product_id_key_ref_product_dim_ [PASS in 0.08s]
15:23:09 27 of 31 START test unique_calendar_dim_date_key ..... [RUN]
15:23:09 27 of 31 PASS unique_calendar_dim_date_key ..... [PASS in 0.06s]
15:23:09 28 of 31 START test unique_customer_dim_customer_id_key ..... [RUN]
15:23:09 28 of 31 PASS unique_customer_dim_customer_id_key ..... [PASS in 0.18s]
15:23:10 29 of 31 START test unique_geo_dim_geo_id ..... [RUN]
15:23:10 29 of 31 PASS unique_geo_dim_geo_id ..... [PASS in 0.18s]
15:23:10 30 of 31 START test unique_product_dim_product_id_key ..... [RUN]
15:23:10 30 of 31 PASS unique_product_dim_product_id_key ..... [PASS in 0.06s]
15:23:10 31 of 31 START test unique_shipping_dim_shipping_id ..... [RUN]
15:23:10 31 of 31 PASS unique_shipping_dim_shipping_id ..... [PASS in 0.06s]
15:23:10
15:23:10 Finished running 31 data tests in 0 hours 0 minutes and 4.95 seconds (4.95s).
15:23:10
15:23:10 Completed successfully
15:23:10
15:23:10 Done. PASS=31 WARN=0 ERROR=0 SKIP=0 NO-OP=0 TOTAL=31

```

Скриншот выполнения dbt snapshot



```

(dbt-env) C:\pde_magistr\superstore_dwh>dbt snapshot
15:24:01 Running with dbt=1.10.15
15:24:01 Registered adapter: postgres=1.9.1
15:24:02 Found 11 models, 31 data tests, 1 snapshot, 1 source, 1 exposure, 450 macros
15:24:02
15:24:02 Concurrency: 1 threads (target='dev')
15:24:02
15:24:02 1 of 1 START snapshot dw_snapshots.snapshot_product_dim ..... [RUN]
15:24:03 1 of 1 OK snapshot dw_snapshots.snapshot_product_dim ..... [SELECT 1894 in 0.17s]
15:24:03
15:24:03 Finished running 1 snapshot in 0 hours 0 minutes and 1.00 seconds (1.00s).
15:24:03
15:24:03 Completed successfully
15:24:03
15:24:03 Done. PASS=1 WARN=0 ERROR=0 SKIP=0 NO-OP=0 TOTAL=1
(dbt-env) C:\pde_magistr\superstore_dwh>

```

## Скриншот данных из mart\_product\_lifecycle

Данные из mart\_product\_lifecycle:

ers	total_quantity_sold	total_sales	product_name	category	subcategory	first_sale_date	last_sale_date	days_between_sales	total_ord
14	119.0	17057.341	Ibico Hi-Tech Manual Binding System Office Supplies	Binders		2014-01-06	2017-12-30	1454 days	
14	119.0	17057.341	GBC Binding covers Office Supplies	Binders		2014-01-06	2017-12-30	1454 days	
6	76.0	350.150	Acco Four Pocket Poly Ring Binder with Label Holder, Smoke, 1" Office Supplies	Binders		2014-01-07	2017-12-24	1447 days	
9	105.0	620.784	Xerox 225 Office Supplies	Paper		2014-01-06	2017-12-10	1434 days	
4	69.0	152.490	Newell 327 Office Supplies	Art		2014-01-20	2017-12-21	1431 days	
8	66.0	743.600	Memorex Micro Travel Drive 8 GB	Technology	Accessories	2014-01-09	2017-12-07	1428 days	
6	70.0	317.442	Avery Metallic Poly Binders Office Supplies	Binders		2014-01-13	2017-12-10	1427 days	
9	133.0	447.294	Acco Pressboard Covers with Storage Hooks, 9 1/2" x 11", Executive Red Office Supplies	Binders		2014-01-20	2017-12-17	1427 days	
7	84.0	215.364	Rogers Handheld Barrel Pencil Sharpener Office Supplies	Art		2014-01-06	2017-12-02	1426 days	
9	119.0	19481.406	GE 30524EE4	Technology	Phones	2014-01-06	2017-12-02	1426 days	

Всего записей в таблице: 1894

## Выводы

Преимущества использования промежуточных моделей и витрин:

- 1) Инкапсуляция бизнес-логики - сложные JOIN и преобразования вынесены в промежуточный слой, что предотвращает дублирование кода
- 2) Упрощение витрин - аналитические модели становятся простыми агрегациями над готовыми денормализованными данными
- 3) Повторное использование - одна промежуточная модель может использоваться в нескольких витринах
- 4) Облегчение тестирования - бизнес-логика тестируется в одном месте
- 5) Улучшение производительности - оптимизированные промежуточные модели ускоряют построение витрин

- 6) Специализация витрин - каждая витрина решает конкретную бизнес-задачу и содержит только необходимые данные
- 7) Упрощение сопровождения - изменения в бизнес-логике вносятся в одном месте (intermediate-модели)
- 8) Лучшая документация - четкое разделение ответственности между слоями улучшает понимание архитектуры

По сравнению с работой напрямую с единой таблицей фактов, предложенная архитектура обеспечивает лучшую масштабируемость, поддерживаемость и соответствие бизнес-потребностям.