

Product queries

Results with small data samples

1,000 initial products + 1,000 inserted

Scenario	Benchmark T1 (s)	Product Indices T1 (s)	Product + Order Proposal Indices T1 (s)
Create new products	2.029	2.017	2.090
Fetch all products (100 runs)	0.045	0.046	0.051
Fetch products by category (100 runs)	0.025	0.025	0.026
Fetch products by category IF they have order proposals (100 runs)	0.006	0.006	0.006
Fetch products by category IF they DO NOT have order proposals (100 runs)	0.026	0.026	0.027
Fetch products by SCM (100 runs)	0.012	0.011	0.012
Fetch products by SCM IF they have order proposals (100 runs)	0.006	0.005	0.006
Fetch products by SCM IF they DO NOT have order proposals (100 runs)	0.013	0.012	0.013
Fetch products by subcategory (100 runs)	0.005	0.005	0.005
Fetch products by subcategory IF they have order proposals (100 runs)	0.006	0.005	0.006
Fetch products by subcategory IF they DO NOT have order proposals (100 runs)	0.006	0.006	0.006

Results with average data samples

100,000 initial products + 50,000 inserted

Scenario	Benchmark T2 (s)	Product Indices T2 (s)	Product + Order Proposal Indices T2 (s)
Create new products	113.063	105.490	108.663
Fetch all products (100 runs)	3.595	3.387	3.512
Fetch products by category (100 runs)	2.127	2.231	2.326
Fetch products by category IF they have order proposals (100 runs)	0.065	0.048	0.051
Fetch products by category IF they DO NOT have order proposals (100 runs)	2.140	2.168	2.300
Fetch products by SCM (100 runs)	0.589	0.547	0.582
Fetch products by SCM IF they have order proposals (100 runs)	0.055	0.023	0.025
Fetch products by SCM IF they DO NOT have order proposals (100 runs)	0.581	0.546	0.574
Fetch products by subcategory (100 runs)	0.056	0.005	0.039
Fetch products by subcategory IF they have order proposals (100 runs)	0.056	0.005	0.040
Fetch products by subcategory IF they DO NOT have order proposals (100 runs)	0.056	0.005	0.040

Results with larger data samples

700,000 initial products + 50,000 inserted

Scenario	Benchmark T3 (s)	Product Indices T3 (s)	Product + Order Proposal Indices T3 (s)
----------	---------------------	------------------------------	---

Create new products	112.316	108.488	108.420
Fetch all products (100 runs)	3.392	3.334	3.357
Fetch products by category (100 runs)	0.050	0.005	0.005
Fetch products by category IF they have order proposals (100 runs)	0.050	0.005	0.005
Fetch products by category IF they DO NOT have order proposals (100 runs)	0.050	0.005	0.005
Fetch products by SCM (100 runs)	0.562	0.528	0.556
Fetch products by SCM IF they have order proposals (100 runs)	0.054	0.022	0.024
Fetch products by SCM IF they DO NOT have order proposals (100 runs)	0.562	0.550	0.553
Fetch products by subcategory (100 runs)	0.050	0.007	0.005
Fetch products by subcategory IF they have order proposals (100 runs)	0.050	0.006	0.005
Fetch products by subcategory IF they DO NOT have order proposals (100 runs)	0.050	0.007	0.005

Raw Test Data

Benchmark Run 1

No code changes, 1000 existing products + 1000 inserted

[illegible]

Benchmark Run 2

No code changes, 100000 existing products + 50000 inserted

```
tests/test_query_speed.py::ProductQueryPerformanceTest::test_daily_import_
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
```

Benchmark Run 3

No code changes, 700000 existing products + 50000 inserted

```
tests/test_query_speed.py::ProductQueryPerformanceTest::test_daily_import_
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
```

Product Indices Test Run 1

Indices added for Product on supply_chain_manager, category, subcategory; 1000 existing products + 1000 inserted

```
tests/test_query_speed.py::ProductQueryPerformanceTest::test_daily_import_
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
```

Product Indices Test Run 2

Indices added for Product on supply_chain_manager, category, subcategory; 100000 existing products + 50000 inserted

tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product

Product + OrderProposal Indices Test Run 3

Indices added for Product on supply_chain_manager, category, subcategory AND for OrderProposal on calculated_for; 700000 existing products + 50000 inserted

tests/test_query_speed.py::ProductQueryPerformanceTest::test_daily_import_
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product
tests/test_query_speed.py::ProductQueryPerformanceTest::test_fetch_product

Order Proposal queries

Results with small data samples

1,000 initial order proposals (and products) + 1,000 inserted

Scenario	Benchmark T1 (s)	Product + Order Proposal Indices T1 (s)
Create new order proposals	0.267	0.255
Fetch all order proposals (100 runs)	0.072	0.076
Fetch order proposals by category (100 runs)	0.037	0.038
Fetch order proposals by category, latest only (100 runs)	0.047	0.045

Fetch order proposals with strictly positive quantities (100 runs)	0.040	0.038
Fetch order proposals with strictly positive quantities, latest only (100 runs)	0.053	0.047
Fetch order proposals by SCM (100 runs)	0.010	0.009
Fetch order proposals by SCM, latest only (100 runs)	0.017	0.014
Fetch order proposals by subcategory (100 runs)	0.004	0.004
Fetch order proposals by subcategory, latest only (100 runs)	0.009	0.008
Fetch order proposals by supplier_id (100 runs)	0.075	0.074
Fetch order proposals by supplier_id, latest only (100 runs)	0.084	0.084
Fetch all order proposals, latest only (100 runs)	0.082	0.083

Results with average data samples

100,000 initial order proposals (and products) + 50,000 inserted

Scenario	Benchmark T2 (s)	
Create new order proposals	12.453	12.964
Fetch all order proposals (100 runs)	5.895	5.738
Fetch order proposals by category (100 runs)	3.957	3.867
Fetch order proposals by category, latest only (100 runs)	4.199	4.209

Fetch order proposals with strictly positive quantities (100 runs)	2.802	2.865
Fetch order proposals with strictly positive quantities, latest only (100 runs)	3.054	3.118
Fetch order proposals by SCM (100 runs)	0.712	0.721
Fetch order proposals by SCM, latest only (100 runs)	0.766	0.765
Fetch order proposals by subcategory (100 runs)	0.160	0.161
Fetch order proposals by subcategory, latest only (100 runs)	0.164	0.165
Fetch order proposals by supplier_id (100 runs)	0.095	0.072
Fetch order proposals by supplier_id, latest only (100 runs)	0.105	0.082
Fetch all order proposals, latest only (100 runs)	6.080	6.057

Results with larger data samples

700,000 initial order proposals (and products) + 50,000 inserted

Scenario	Benchmark T3 (s)	
Create new order proposals	12.642	13.457
Fetch all order proposals (100 runs)	5.781	5.800
Fetch order proposals by category (100 runs)	3.868	3.935
Fetch order proposals by category, latest only (100 runs)	4.314	4.237
Fetch order proposals with strictly positive quantities (100 runs)	2.810	2.864

Fetch order proposals with strictly positive quantities, latest only (100 runs)	3.045	2.969
Fetch order proposals by SCM (100 runs)	0.727	0.711
Fetch order proposals by SCM, latest only (100 runs)	0.798	0.746
Fetch order proposals by subcategory (100 runs)	0.197	0.161
Fetch order proposals by subcategory, latest only (100 runs)	0.201	0.167
Fetch order proposals by supplier_id (100 runs)	0.122	0.096
Fetch order proposals by supplier_id, latest only (100 runs)	0.132	0.107
Fetch all order proposals, latest only (100 runs)	6.045	6.109

Raw Test Data

Benchmark Run 1

No code changes, 1,000 existing OrderProposals (and Products) + 1,000 OPs inserted

```
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_create_
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
```

Benchmark Run 2

No code changes, 100,000 existing OrderProposals (and Products) + 50,000 OPs inserted

```
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_create_
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_o
```


Product + Order Proposal Indices Test Run 2

Indices added for Product on supply_chain_manager, category, subcategory AND for OrderProposal on product_id, supplier_id, calculated_for, quantity and created_at; 100,000 existing OrderProposals (and Products) + 50,000 OPs inserted

```
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_create_order_proposal PASSED [ 0%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 1%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 2%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 3%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 4%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 5%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 6%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 7%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 8%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 9%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [10%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [11%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [12%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [13%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [14%]
```

Product + OrderProposal Indices Test Run 3

Indices added for Product on supply_chain_manager, category, subcategory AND for OrderProposal on product_id, supplier_id, calculated_for, quantity and created_at; 300,000 existing OrderProposals (and 100,000 Products) + 50,000 OPs inserted

```
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_create_order_proposal PASSED [ 0%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 1%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 2%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 3%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 4%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 5%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 6%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 7%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 8%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [ 9%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [10%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [11%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [12%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [13%]
tests/test_query_speed.py::OrderProposalQueryPerformanceTest::test_fetch_order_proposal PASSED [14%]
```