

Upskill with StackOps Series

Log Analysis (ESQL) & Dashboarding in StackOps

Gary Wong, Lead Product Manager, StackOps
Boyang Tang, Software Engineer (II), StackOps

StackOps Upskill Series

Webinar (Online)

**Accelerating Deployment
Discovery with SOA**

● Jan 2026

Workshop (In-person)

**Visualisation &
Observability in StackOps**

● Feb 2026

Workshop (In-person)

**Overview & Ingestion in
StackOps**

Webinar (Online)

**Log Analysis (ESQL) &
Dashboarding**

● Mar 2026

Workshop (In-person)

**Applying Machine
Learning in StackOps**

**Register for our March
workshop**

Webinar (Online)

AI Assistants

● Apr 2026

Workshop (In-person)

**App Monitoring in
StackOps**

Webinar (Online)

**Detection & Recovery
through AI/ML**

● May 2026

Workshop (In-person)

**Visualisation &
Observability in
StackOps**

Upskill Webinars (Online)

This new series will cover short technical sessions focused on specific platform features

Technical Workshops (In-person)

Continuing with our classroom-based sessions with hands-on labs. From foundational ingestion to advanced machine learning, these instructor-led sessions equip you with the skills to transform raw data into actionable insights.

Learning Events Page

Refer to StackOps Learning Events page for latest updates on upcoming and past events:

<https://go.gov.sg/stackops-learning-events>



Q & A

Please add your questions for the Q &A session at the end!

<https://pigeonhole.at/STACKOPS>



Agenda

- Dashboarding in Elastic
- Why ES|QL?
- Live ES|QL exploration
- Q&A

What is Dashboarding?

A dashboard is:

- Collection of visualizations
- Powered by queries
- Interactive via filters & time range

StackOps Innersource Dashboards

 Innersource /  SGTS / StackOps / Visualize Library

apex_tenants_dashboard

central_siem_alert_dashboard

central_siem_rule_dashboard

central_threat_intel_dashboard

endpoint_metrics_correlation_dashboard

kubernetes_dashboard

project_alert_dashboard

project_endpoint_login_dashboard

project_siem_alert_dashboard

project_siem_exception_analysis_dashboard

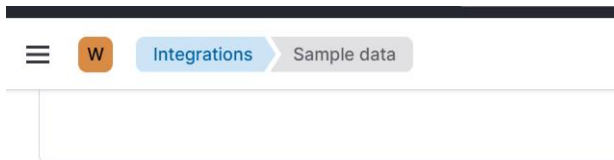
project_slo_dashboard

project_synthetics_dashboard

<https://sgts.gitlab-dedicated.com/innersource/sgts/stackops/visualize-library>

Sample data

Integrations -> Sample
data -> eCommerce orders



✓ Other sample data sets

INSTALLED

Sample eCommerce Data

This dashboard contains sample data for you to play with. You can view it, search it, and interact with the visualizations. For more information about features, check our docs.

Sum of revenue

\$80,027.35

Median spending

\$67

Category revenue over time



Conversion rate over time



Performance over time



Top items sold

2.2

Items sold per day

150.29

Item	Revenue	Profit	Margin	Units	Price
Item 1	\$10,000	\$2,000	20%	100	\$100
Item 2	\$8,000	\$1,600	20%	80	\$100
Item 3	\$6,000	\$1,200	20%	60	\$100
Item 4	\$4,000	\$800	20%	40	\$100
Item 5	\$2,000	\$400	20%	20	\$100

Sample eCommerce orders

Sample data, visualizations, and dashboards for tracking eCommerce orders.

Remove

View data
▼

Why ES|QL?

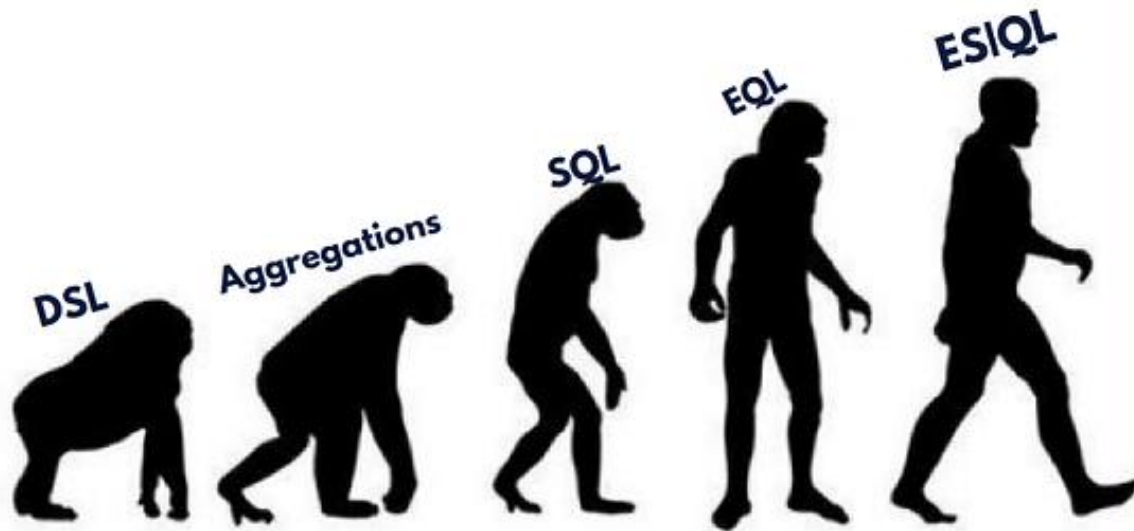
Traditional approaches in Elastic:

- Query DSL -> Powerful but complex
- KQL -> Great for filtering
- SQL -> Familiar but limited for pipeline

ES|QL introduces:

- Pipeline based querying
- Inline transformations
- Built-in aggregations
- Analytics-ready outputs
- Easy to read
- Continuous performance improvement (2.4x faster for aggregation since 8.15)

The Evolution of Querying in Elasticsearch



KQL Sample

≡

+

Q

customer_first_name.keyword: "Abd" OR taxful_total_price > 200

DSL Sample (Daily revenue for the last 30 days)

```

1 // Query DSL
2 POST kibana_sample_data_ecommerce/_search
3 {
4   "size": 0,
5   "query": {
6     "bool": {
7       "filter": [
8         {
9           "range": {
10            "order_date": {
11              "gte": "now-30d/d",
12              "lte": "now/d"
13            }
14          }
15        ]
16      }
17    },
18  },
19  "aggs": {
20    "daily_revenue": {
21      "date_histogram": {
22        "field": "order_date",
23        "fixed_interval": "1d"
24      },
25      "aggs": {
26        "revenue": {
27          "sum": {
28            "field": "taxful_total_price"
29          }
30        }
31      }
32    }
33  }
34 }

```

```

// ES|QL
POST _query
{
  "query": ""
  FROM kibana_sample_data_ecommerce
  | WHERE order_date >= NOW() - 30days
  | STATS revenue = SUM(taxful_total_price) BY order_date = DATE_TRUNC(1 day, order_date)
  | SORT order_date DESC
  ""
}

```

ES|QL Mental Model

```
1  FROM index
2  | WHERE condition
3  | STATS aggregation BY field
4  | SORT field DESC
5  | LIMIT 10
```

Lets explore ES|QL

Documentations:

[ES|QL reference](#) | [Elasticsearch Reference](#)

[ES|QL syntax reference](#) | [Elasticsearch Reference](#)

Reading raw data

```
FROM kibana_sample_data_ecommerce  
| LIMIT 10
```

Read 10 row of data from the index

(ES|QL default limit is 1000)

```
1 FROM kibana_sample_data_ecommerce  
2 | LIMIT 10
```

Select specific fields

```
FROM kibana_sample_data_ecommerce  
| KEEP order_date, customer_full_name, total_quantity,  
taxful_total_price, category  
| LIMIT 10
```

- Only how the 5 specified fields
- Return 10 rows

```
1 FROM kibana_sample_data_ecommerce  
2 | KEEP order_date, customer_full_name, total_quantity, taxful_total_price, category  
3 | LIMIT 10
```

Filter Data

```
FROM kibana_sample_data_ecommerce
| KEEP order_date, customer_full_name, total_quantity,
taxful_total_price, category
| WHERE taxful_total_price > 200
| SORT taxful_total_price DESC
| LIMIT 10
```

- Only keep 5 specified columns
- Filter by total price > 200
- Sort total price by desc order
- Return 10 rows

```
1 FROM kibana_sample_data_ecommerce
2 | KEEP order_date, customer_full_name, total_quantity, taxful_total_price, category
3 | WHERE taxful_total_price > 200
4 | SORT taxful_total_price DESC
5 | LIMIT 10
```

Aggregations

```
FROM kibana_sample_data_ecommerce  
| STATS total_sales = SUM(taxful_total_price) BY  
category  
| SORT total_sales DESC
```

- Calculate total sales by adding total price
- Group the total sales by category
- Sort the total sales in desc order

TLDR: Total sales aggregated by category

Common use case:

- Error count grouped by IPs

```
1 FROM kibana_sample_data_ecommerce  
2 | STATS total_sales = SUM(taxful_total_price) BY category  
3 | SORT total_sales DESC
```

Runtime calculation

```
FROM kibana_sample_data_ecommerce  
| EVAL avg_price_per_quantity = taxful_total_price / total_quantity  
| KEEP avg_price_per_quantity, taxful_total_price, total_quantity  
| LIMIT 10
```

```
1 FROM kibana_sample_data_ecommerce  
2 | EVAL avg_price_per_quantity = taxful_total_price / total_quantity  
3 | KEEP avg_price_per_quantity, taxful_total_price, total_quantity  
4 | LIMIT 10  
5
```

Time-Based Analysis

```
FROM kibana_sample_data_ecommerce
| WHERE order_date > NOW() - 90d
| EVAL week = DATE_TRUNC(1 week, order_date)
| STATS revenue = SUM(taxful_total_price) BY week
| SORT week DESC
```

- Filter to include orders from last 90 days
- Create a weekly bucket using order date
- Group weekly revenue
- Sort week in desc order to show latest week on top

TLDR: Calculate weekly revenue

```
1 FROM kibana_sample_data_ecommerce
2 | WHERE order_date > NOW() - 90d
3 | EVAL week = DATE_TRUNC(1 week, order_date)
4 | STATS revenue = SUM(taxful_total_price) BY week
5 | SORT week DESC
```

Customer Behaviour Analysis

```
FROM kibana_sample_data_ecommerce
```

```
| STATS
```

```
total_spent = SUM(taxful_total_price),  
orders = COUNT(*)
```

```
BY customer_id, customer_full_name
```

```
| WHERE orders >= 3
```

```
| EVAL tier =
```

```
CASE(
```

```
total_spent > 500, "VIP",  
total_spent > 250, "Regular",  
"Occasional"
```

```
)
```

```
| STATS
```

```
customers = COUNT(*),  
avg_spent = AVG(total_spent),  
total_revenue = SUM(total_spent)
```

```
BY tier
```

```
| SORT total_revenue DESC
```

```
// Group number of orders & total spent by customer id & full name  
// Filter customers with 3 or more orders
```

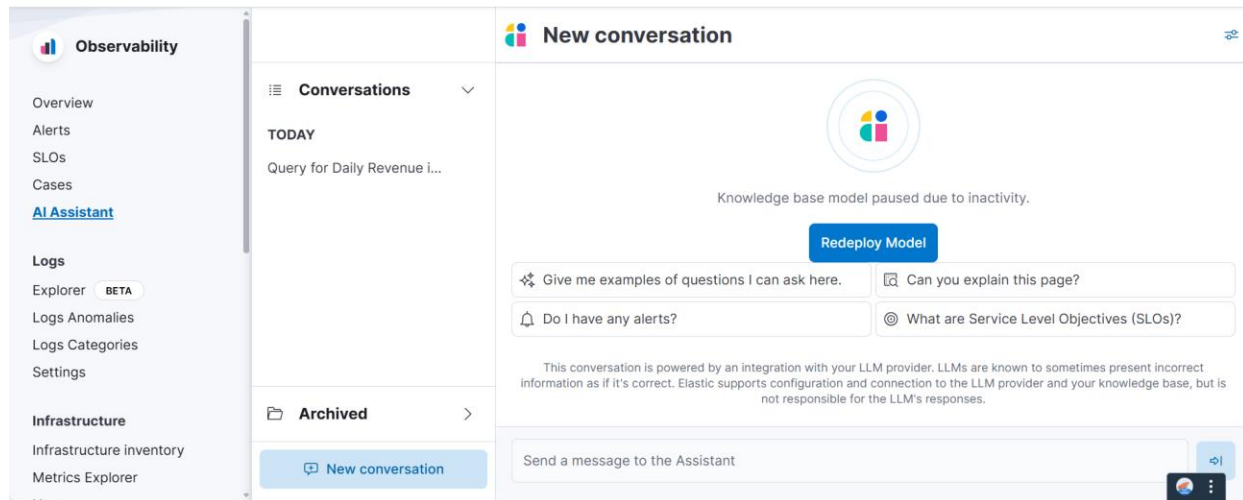
```
// Categorise customers based on total spending
```

```
// Get total revenue, customers and avg spent grouped in tier
```

AI Can Help!

Elastic AI Assistant

- Contextual insights
- Chat interface



Sample Dashboard with ES|QL



https://sgts.gitlab-dedicated.com/innersource/sgts/stackops/visualize-library/-/tree/main/dashboards/project_slo_dashboard?ref_type=heads

Let's hear from you!

Applying Machine Learning in StackOps

Workshop Details



27 March



9:30 AM to 1:00 PM



GovTech PDD

* Key Takeaways

- ✓ Introduction to Elastic Machine Learning in StackOps
- ✓ Benefits and Pre-requisites for ML
- ✓ Kick Start: Using Pre-Configured Anomaly Detection Jobs (Hands-on)
- ✓ Configuring Custom ML Jobs for Specific Use Cases (Hands-on)
- ✓ Analysing Results from ML Jobs

Last day to register



<https://go.gov.sg/stackops-workshop-signup>

Q & A

<https://pigeonhole.at/STACKOPS>



Share your feedback on the session

Please share your feedback on the session to help us improve in future!

<https://go.gov.sg/stackops-webinar-feedback>



<https://go.gov.sg/stackops-webinar-feedback>

Thank you.