Visualising Product Metrics

Purpose

Product metrics provide insights into API utilisation by storing all details related to the API. It is important to have a dashboard to support product metrics. Hive table is used for visualising the data. With these insights, we can determine if a feature (new API) is creating value, analyse usage trends, and understand response patterns. Product metrics help us make informed decisions about future investments in the feature. As Splunk is being decommissioned, this solution can serve the purpose of querying logs. Additionally, we can proactively monitor the stored data for any unexpected behaviour and take necessary actions.

If you're interested in understanding the system architecture, how API logs are published, consumed, and persisted in Google Cloud Storage, refer to the following documents.

- **ADT** - API Logs for Product Metrics - ADT
- **GCS Sink Service** - Kafka Connect GCS Sink Service

Onboarding The Api Audit Logging Service

In order to have product metrics of your service, the application should be integrated with the audit-logging-service. Detailed process for onboarding the audit-logging-service is documented in below link

- **Integrating with the audit logging service** - Audit Logging Service Integration Guide

Data Discovery For Visualisation

API log records are stored in a GCS bucket, and a Hive table is created on top of this data. To visualise the data, we use Data Discovery. The following Active Directory (AD) groups are essential for gaining access to the API logs:

(Check if you have access to Data Discovery before raising the requests below.)

- **Test / Sandbox Table: us_dv_audit_log_dev.api_logs** (SELECT * FROM us_dv_audit_log_dev.api_logs WHERE service_name = 'NRT' LIMIT 10;)
- **Production Table: us_dv_audit_log_prod.api_logs** (SELECT * FROM us_dv_audit_log_prod.api_logs WHERE service_name = 'NRT' LIMIT 10;)
  Some more sample queries provided below with screenshots

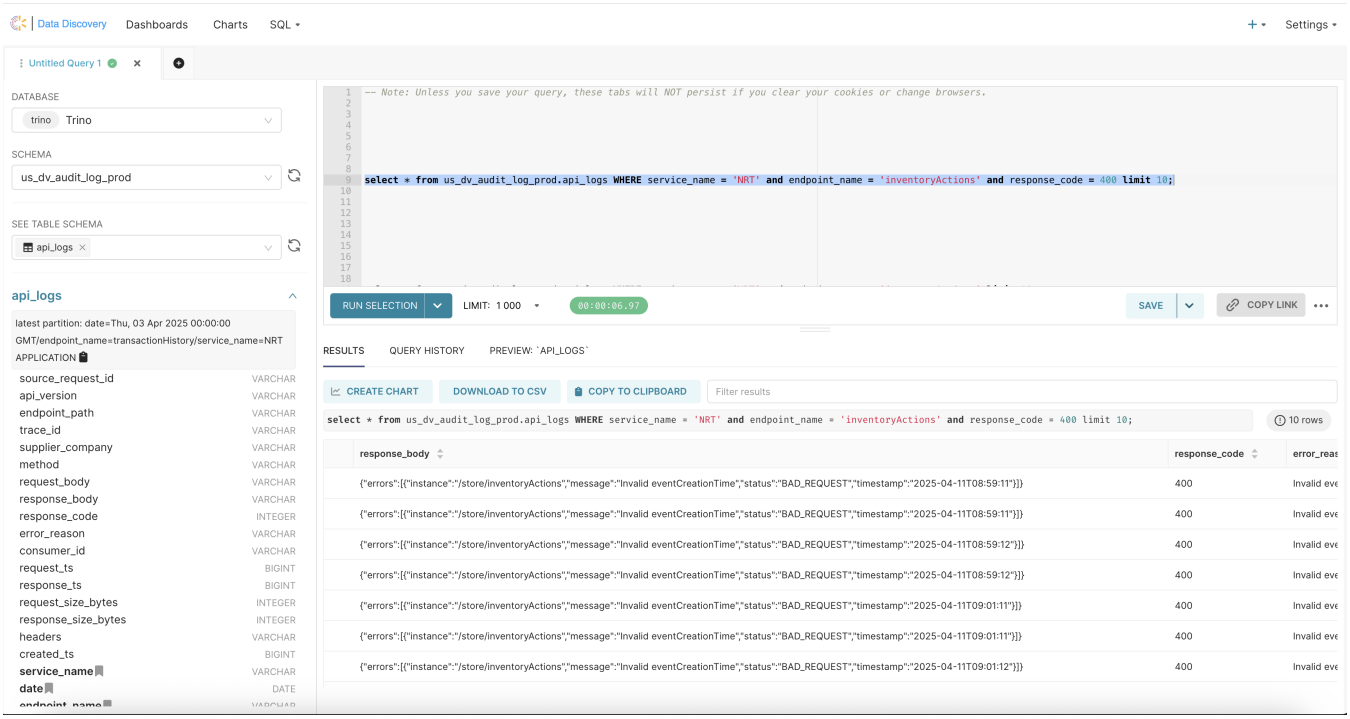
Required AD Groups and Access Details

AD Group Name	Access Description	Request Access via	Notes / Links
GCP-DATA-DISCOVERY-PROD-ROLE-LOGIN	Access to Data Discovery	Service Now	**Link** - Data Discovery
GCP-DL-DV-LUMINATE-DEV-READ	Access to test data in Data Discovery	Service Now	**Schema** - us_dv_audit_log_dev **Table** - api_logs
GCP-DL-DV-LUMINATE-PROD-READ	Access to production data in Data Discovery	Service Now	**Schema** - us_dv_audit_log_prod **Table** - api_logs

Reference screenshots

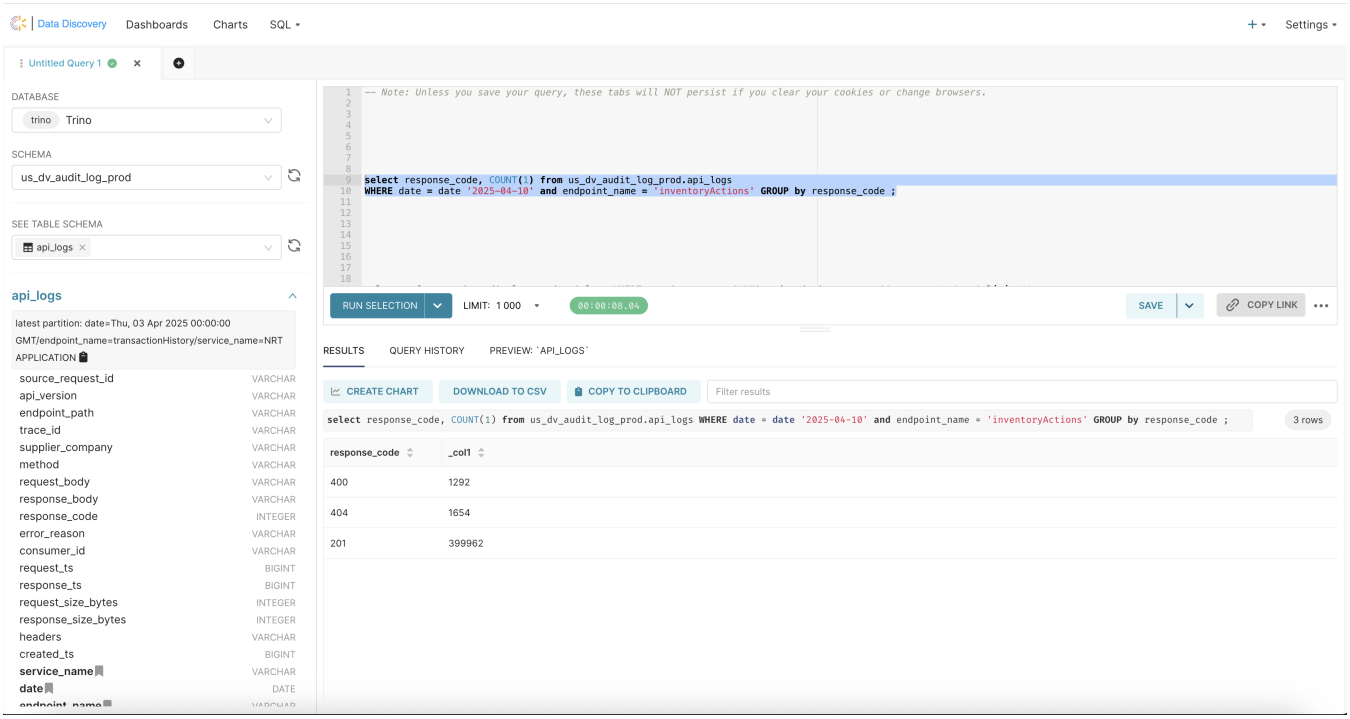
Query by Service Name, Endpoint Name, and Response Code:

```
 SELECT * FROM us_dv_audit_log_prod.api_logs WHERE service_name = 'NRT' AND endpoint_name = 'inventoryActions'
AND response_code = 400 LIMIT 10;
```



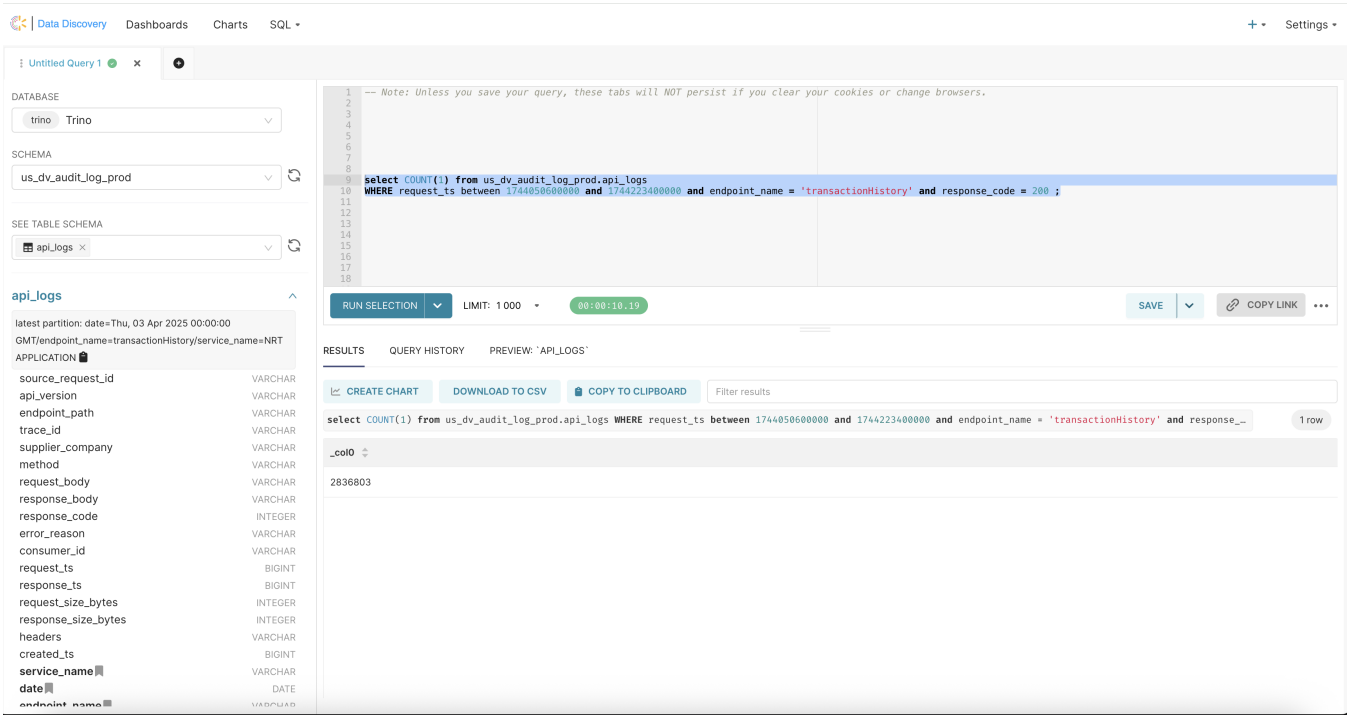
Group by Response Code

```
SELECT response_code, COUNT(1) FROM us_dv_audit_log_prod.api_logs WHERE date = DATE '2025-04-10' AND
endpoint_name = 'inventoryActions' GROUP BY response_code;
```



Count Requests in a Time Range

```
SELECT COUNT(1) FROM us_dv_audit_log_prod.api_logs WHERE request_ts BETWEEN 1744050600000 AND 1744223400000 AND
endpoint_name = 'transactionHistory' AND response_code = 200;
```



Few more useful queries

Group by endpoint_name, its count for a service on a particular date

```
select endpoint_name, COUNT(1) from us_dv_audit_log_prod.api_logs
WHERE date = date '2025-05-01' and service_name = 'NRT' GROUP BY endpoint_name ;
```

Group by response_code, its count for a service, endpoint_name on a particular date

```
SELECT response_code, COUNT(1) from us_dv_audit_log_prod.api_logs
WHERE date = date '2025-05-01' and service_name = 'NRT'
and endpoint_name = 'nrti_directshipment' GROUP BY response_code ;
```

Group by consumer_id, its count for a service, endpioint_name on a particular date

```
SELECT consumer_id, COUNT(1) from us_dv_audit_log_prod.api_logs
WHERE date = date '2025-05-01' and service_name = 'NRT'
and endpoint_name = 'nrti_directshipment' GROUP BY consumer_id ;
```

Group by response_code, endpoint_name, its count for a service, consumer_id on a particular date

```
SELECT response_code,endpoint_name, COUNT(1) from us_dv_audit_log_prod.api_logs
WHERE date = date '2025-05-01' and service_name = 'NRT'
and consumer_id = 'de2fa644-2c7e-11ee-a2a7-ddf5fc7d0ef2' GROUP BY response_code, endpoint_name ;
```

Traversing response_body / request_body to get root field

```
SELECT
  *
FROM
  us_dv_audit_log_prod.api_logs
WHERE
  service_name = 'NRT'
  AND endpoint_name = 'nrti_directshipment'
  AND date = DATE '2025-05-15'
  AND CAST(json_extract_scalar(request_body, '$.vendorId') AS INTEGER) = 600162
LIMIT 10;


--
SELECT
  *
FROM
  us_dv_audit_log_prod.api_logs
WHERE
  service_name = 'NRT'
  AND endpoint_name = 'nrti_dc'
  AND date = DATE '2025-05-15'
  AND CAST(json_extract_scalar(response_body, '$.dcNbr') AS INTEGER) = 6003
LIMIT 10;


--
SELECT
  CAST(json_extract_scalar(response_body, '$.dcNbr') AS INTEGER) AS dc_nbr
FROM
  us_dv_audit_log_prod.api_logs
WHERE
  service_name = 'NRT'
  AND endpoint_name = 'nrti_dc'
  AND date = DATE '2025-05-15'
LIMIT 10;
```

Group by extracting vendorId from request_body and count

```
SELECT
  CAST(json_extract_scalar(request_body, '$.vendorId') AS INTEGER) AS dc_nbr,
  COUNT(*) AS record_count
FROM
  us_dv_audit_log_prod.api_logs
WHERE
  service_name = 'NRT'
  AND endpoint_name = 'nrti_directshipment'
  AND date = DATE '2025-05-15'
GROUP BY
  CAST(json_extract_scalar(request_body, '$.vendorId') AS INTEGER)
ORDER BY
  record_count DESC
LIMIT 10;


-- Another way same result
WITH parsed_logs AS (
  SELECT
    CAST(json_extract_scalar(request_body, '$.vendorId') AS INTEGER) AS dc_nbr
  FROM
    us_dv_audit_log_prod.api_logs
  WHERE
    service_name = 'NRT'
    AND endpoint_name = 'nrti_directshipment'
    AND date = DATE '2025-05-15'
)
SELECT
  dc_nbr,
  COUNT(*) AS record_count
FROM
  parsed_logs
GROUP BY
  dc_nbr
ORDER BY
  record_count DESC
LIMIT 10;
```

Traversing request_body to find by store_nbr

```
WITH parsed_logs AS (
  SELECT
    *,
    CAST(json_extract(request_body, '$.destinations') AS array(json)) AS destinations
  FROM
    us_dv_audit_log_prod.api_logs
  WHERE
    service_name = 'NRT'
    AND endpoint_name = 'nrti_directshipment'
    AND date = DATE '2025-05-15'
),
exploded AS (
  SELECT
    *,
    json_extract_scalar(dest, '$.storeNbr') AS store_nbr
  FROM
    parsed_logs,
    UNNEST(destinations) AS t(dest)
)
SELECT *
FROM exploded
WHERE store_nbr = '4503'
LIMIT 10;
```

Link to get milliseconds (UTC timezone used in the fields) - https://currentmillis.com/