

Oracle Service dashboards and alerts in Grafana using Oracle Enterprise Manager plugin

Author: Nourhene Jallali

Supervisors: Miroslav Potocky
Andrzej Nowicki



Table of Contents

01

Project Overview

02

Oracle Service Metrics

03

Oracle Service Dashboards

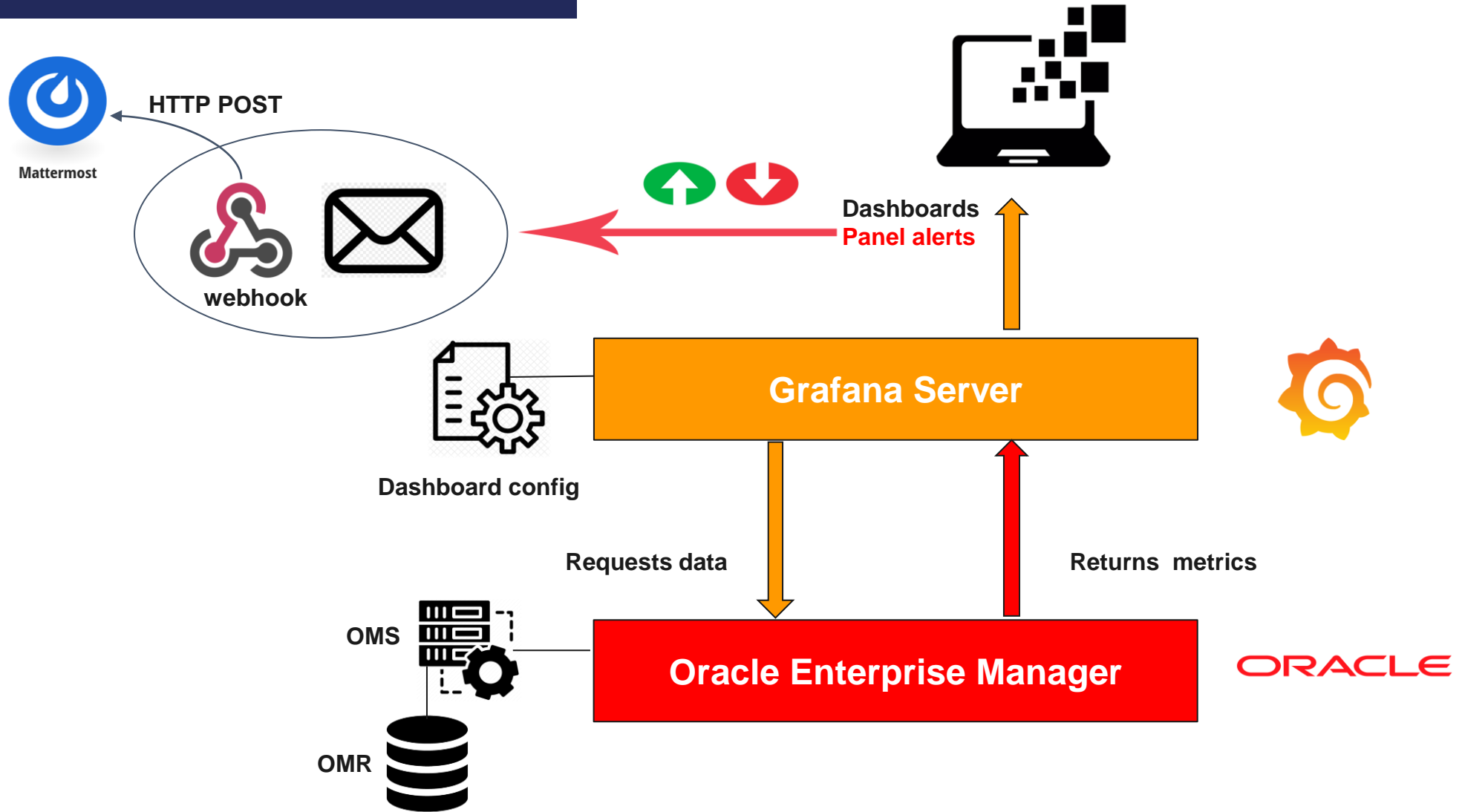
04

Grafana Alerts

05

Future Work

Project Overview



Oracle Service Metrics

For more details on these metrics, you can read more here:

https://docs.oracle.com/cd/E11857_01/em.111/e12135/views.htm#BACCF CAB

Oracle Metrics

User metrics

Name of the Metric	Calculation	Which impact measures	User community	Collection Frequency
Latency	Latency as calculated by AWR views in DB	User perceived service performance	DB instance	Every 30 Minutes
Throughput	Throughput as calculated by AWR views in DB	User perceived service performance	DB instance	Every 30 Minutes

Management metrics

Name of the Metric	Calculation	Which impact measures	User community	Collection Frequency
Number of instances	Count of all instances	Service growth	Deployment type (RAC, Single, Cloud)	Every 24 hours
Number of schemas	Count of all schemas	Service growth	ALL	Every 24 hours
OCPU count	Sum of physical CPUs / 2 (for intel and AMD)	Licensing requirement	ALL	Every 24 hours
Feature pack usage	List of feature packs per database	Licensing requirement	Feature pack	Every 24 hours
Database status	The availability of the database	Service health	ALL	Every 5 Minutes

The first screenshot shows a query for 'TOTAL_DETECTED_USAGES':

```
SELECT NAME, DATABASE_NAME, TARGET_TYPE, CURRENTLY_USED,
SUM(DETECTED_USAGES) AS TOTAL_DETECTED_USAGES
FROM sysman.MGMT$DB_FEATUREUSAGE
GROUP BY DATABASE_NAME, CURRENTLY_USED, TARGET_TYPE, NAME;
```

The second screenshot shows a query for 'number_schemas':

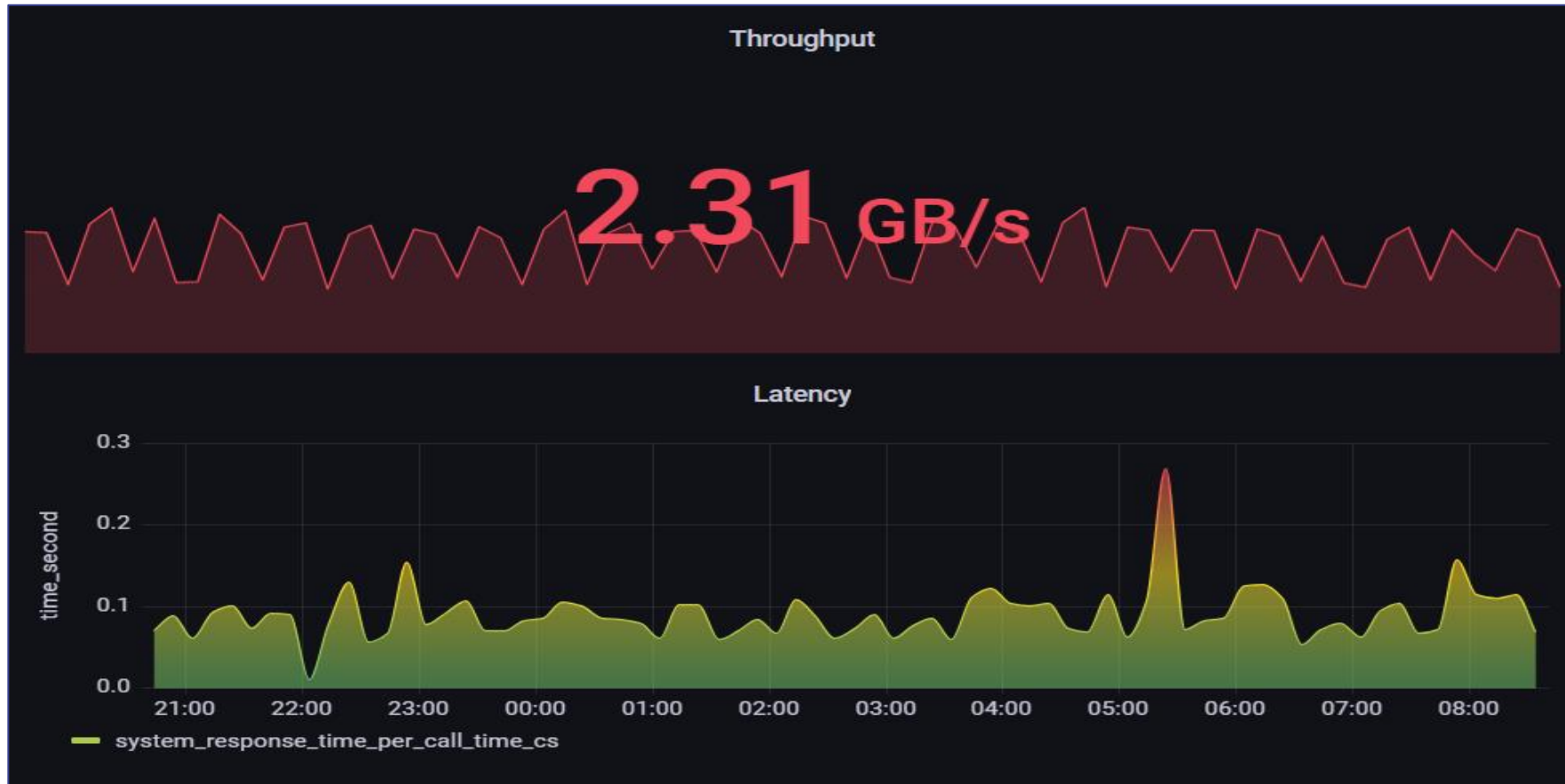
```
select substr(target_name,0,instr(target_name,',',1)-1) as target_name ,
count(distinct USERNAME) as number_schemas
from MGMT$DB_USERS
group by substr(target_name,0,instr(target_name,',',1)-1)
order by 1;
```

The third screenshot shows a query for 'Number_of_Oracle_CPU':

```
SELECT SUM(CPU_COUNT) / 2 AS Number_of_Oracle_CPU
FROM sysman.MGMT$OS_HW_SUMMARY;
```

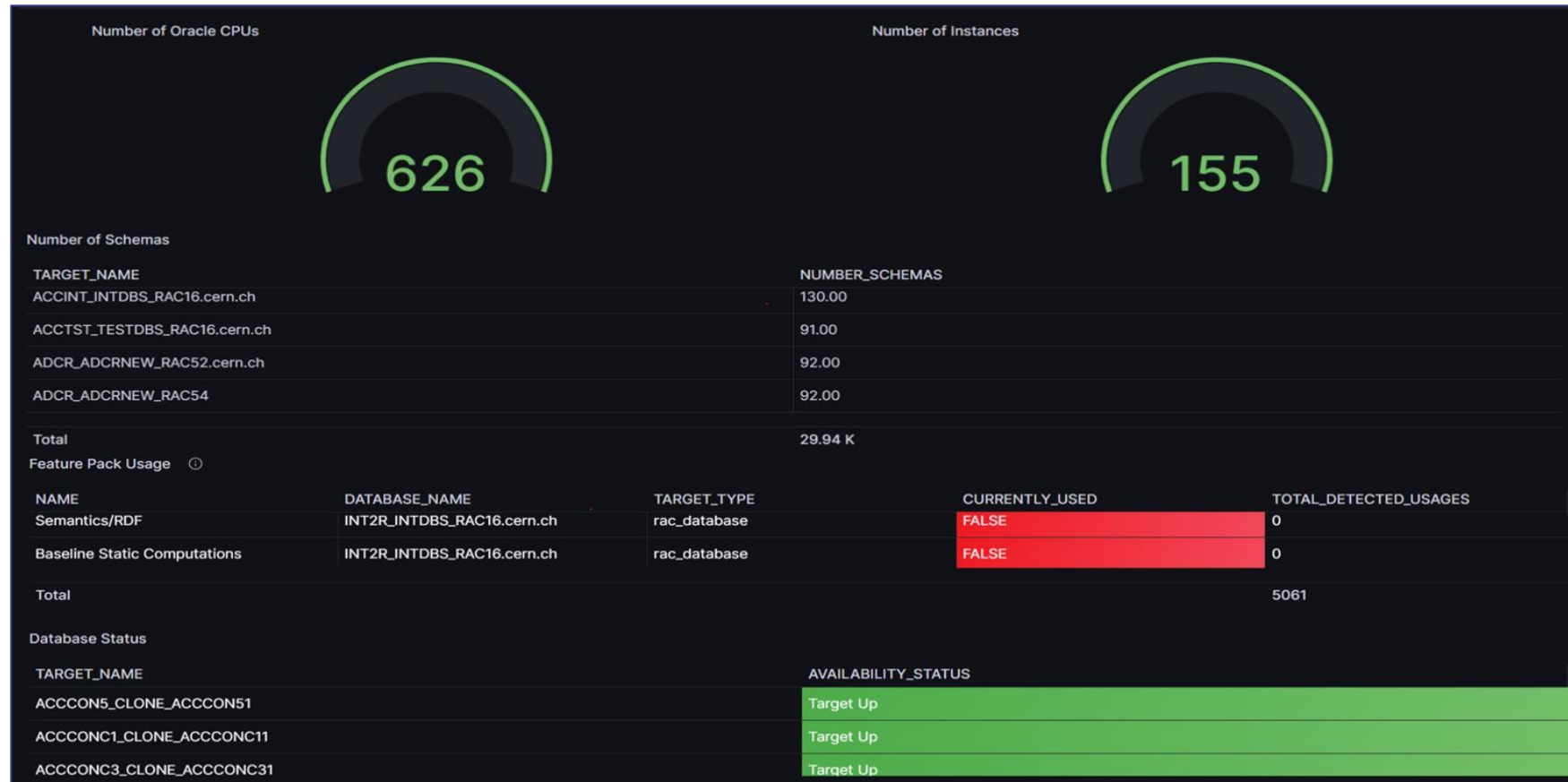
Oracle Service Dashboards

User Metrics

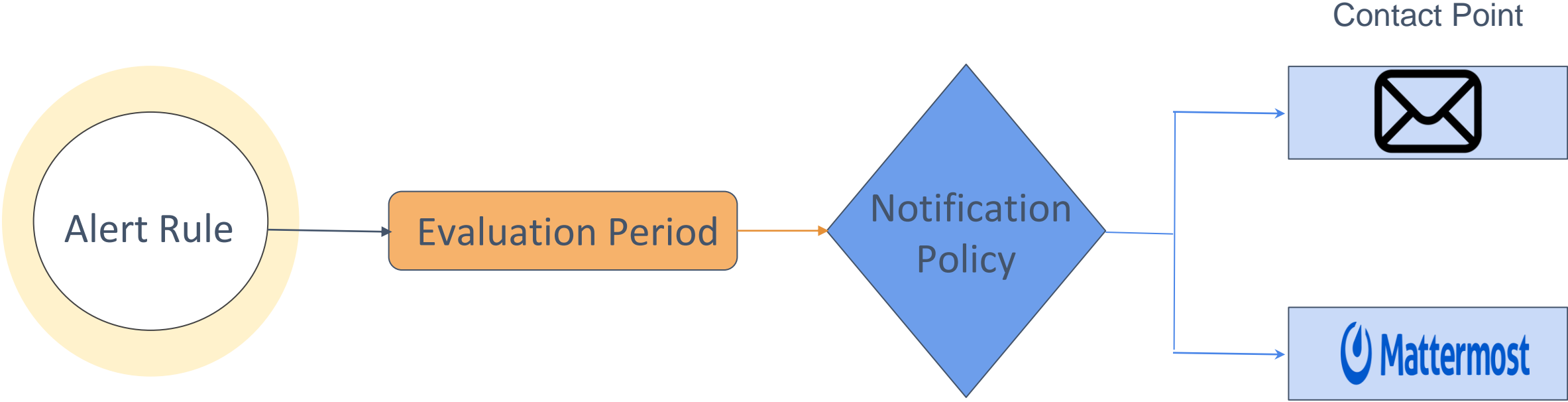


Oracle Service Dashboards

Management Metrics



Grafana Alerts



Grafana Alerts

Alert State: Firing

Email

The screenshot shows an Outlook email interface. The main content is an email from Grafana with the subject "[FIRING:1] High Service Latency alert (2)". The email body features the Grafana logo and the following text:

Firing: 1 alert for alertname=High Service Latency grafana_folder=alert

Firing **High Service Latency**

Value: [var=A0 metric=system_response_time_per_call_time_cs labels={} value=0.12708973060551873]

Summary: The latency exceeds 2 seconds, which can severely impact user experience and system performance.

Labels: alertname: High Service Latency
grafana_folder: alert
latency: 2

Buttons: Silence, Go to Dashboard, Go to Panel, Source

Go to alerts page

Sent by Grafana v9.2.3
© 2022 Grafana Labs

Grafana Alerts

Webhook Configuration

Incoming Webhooks > Edit

Title grafana-alerts

Specify a title, of up to 64 characters, for the webhook settings page.

Description This webhook allows Grafana to send au

Describe your incoming webhook.

Channel webhook integration test

This is the default public or private channel that receives the webhook payloads. When setting up the webhook, you must belong to the private channel.

Lock to this channel

grafana-alerts [Edit](#) - [Delete](#)

This webhook allows Grafana to send automated notifications directly to this Mattermost channel.

URL: <https://mattermost.web.cern.ch>

Created by njallali on Friday, July 26, 2024

Contact points

Alertmanager Grafana

Choose how to notify your contact points when an alert instance fires

Update contact point

Name *
Nour Jallali

Integration
Slack

Recipient
Specify channel, private group, or IM channel (can be an encoded ID or a name) - required unless you provide a webhook

Token
Provide a Slack API token (starts with "xoxb") - required unless you provide a webhook

Webhook URL
Optionally provide a Slack incoming webhook URL for sending messages, in this case the token isn't necessary

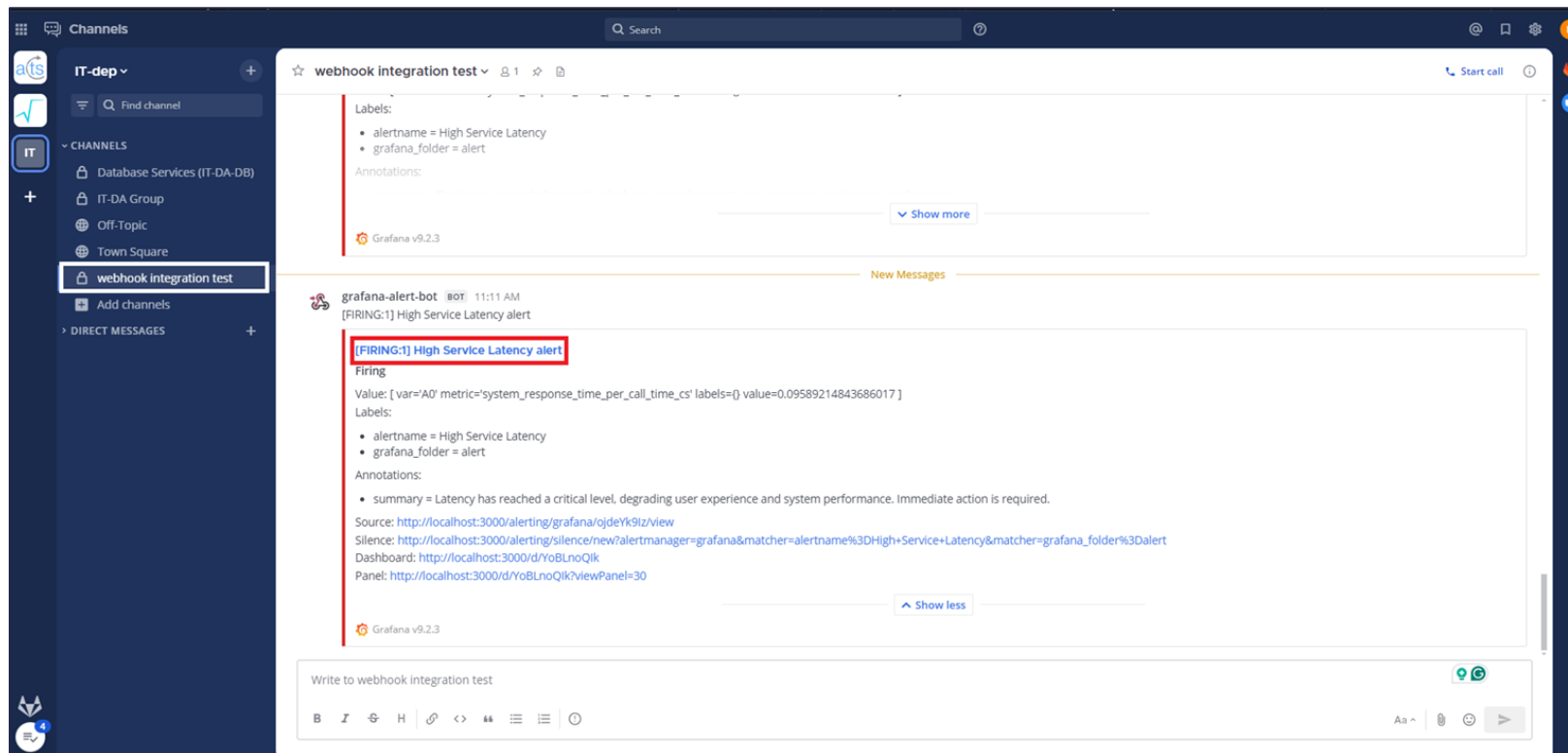
Configured Clear

Buttons: Test Duplicate Delete

Grafana Alerts

Alert State: Firing

Webhook



The screenshot shows a Slack interface with a channel named "webhook integration test". The channel contains a message from a bot named "grafana-alert-bot" at 11:11 AM. The message content is as follows:

[FIRING:1] High Service Latency alert

Firing

Value: [var='A0' metric='system_response_time_per_call_time_cs' labels={} value=0.09589214843686017]

Labels:

- alertname = High Service Latency
- grafana_folder = alert

Annotations:

- summary = Latency has reached a critical level, degrading user experience and system performance. Immediate action is required.

Source: <http://localhost:3000/alerting/grafana/ojdeYk9Iz/view>

Silence: http://localhost:3000/alerting/silence/new?alertmanager=grafana&matcher=alertname%3DHigh+Service+Latency&matcher=grafana_folder%3Dalert

Dashboard: <http://localhost:3000/d/YoBLnoQik>

Panel: <http://localhost:3000/d/YoBLnoQik?viewPanel=30>

The message is displayed in a Slack channel with a dark blue sidebar on the left containing channel navigation options like "Database Services (IT-DA-DB)", "IT-DA Group", "Off-Topic", "Town Square", and "webhook integration test". The main content area shows the alert details with expandable sections for labels and annotations.

Future work

- Creating a Grafana dashboard for the Oracle Cloud Infrastructure
- See HOT/COLD tablespace files within the database

Thank you!

“ Monitor smarter, act faster.”

