

OBSERVABILITY.

ТРИМАЄМО ІТ СИСТЕМИ ПІД КОНТРОЛЕМ

by Artem Mylytsia

Lecturer, Chernihiv Polytechnic National University / Software Engineer, Grammarly

Зв'язок:

linkedin.com/in/artem-mylytsia-2b292b5b/

cs.stu.cn.ua/mylyczya-artem-yurijovych/

Професійний шлях



2012 —
Ступінь магістра зі спеціальності "Системне програмування" в НУ "Чернігівська Політехніка"



2021 - 2025
Ledger - від DevOps до Observability Team Lead

2025 - теперішній час
Grammarly - Software Engineer



Фокус задач:

Проектування, розробка, впровадження, оптимізація і підтримка:

- програмного забезпечення і інфраструктури
- систем спостереження

1

2

3

4

5

2010 - 2021

PortaOne - від Junior Tech Engineer до Head of Support department



2024 - теперішній час

НУ "Чернігівська Політехніка" - викладач



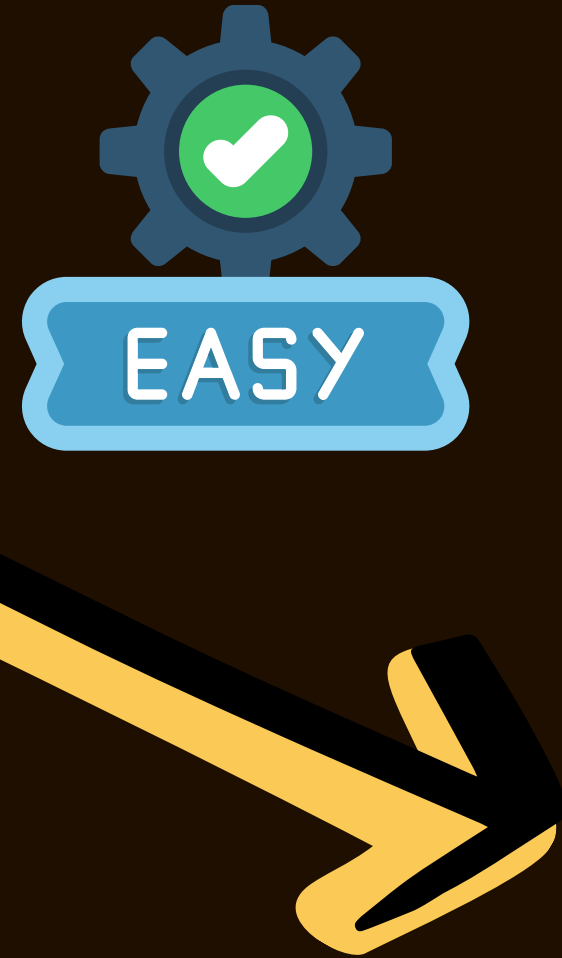


aws



Використання

- NASA
- Національний університет “Чернігівська політехніка”
- ? Будь-які ? тех бізнеси



```
pythonProject1 - main.py - Administrator

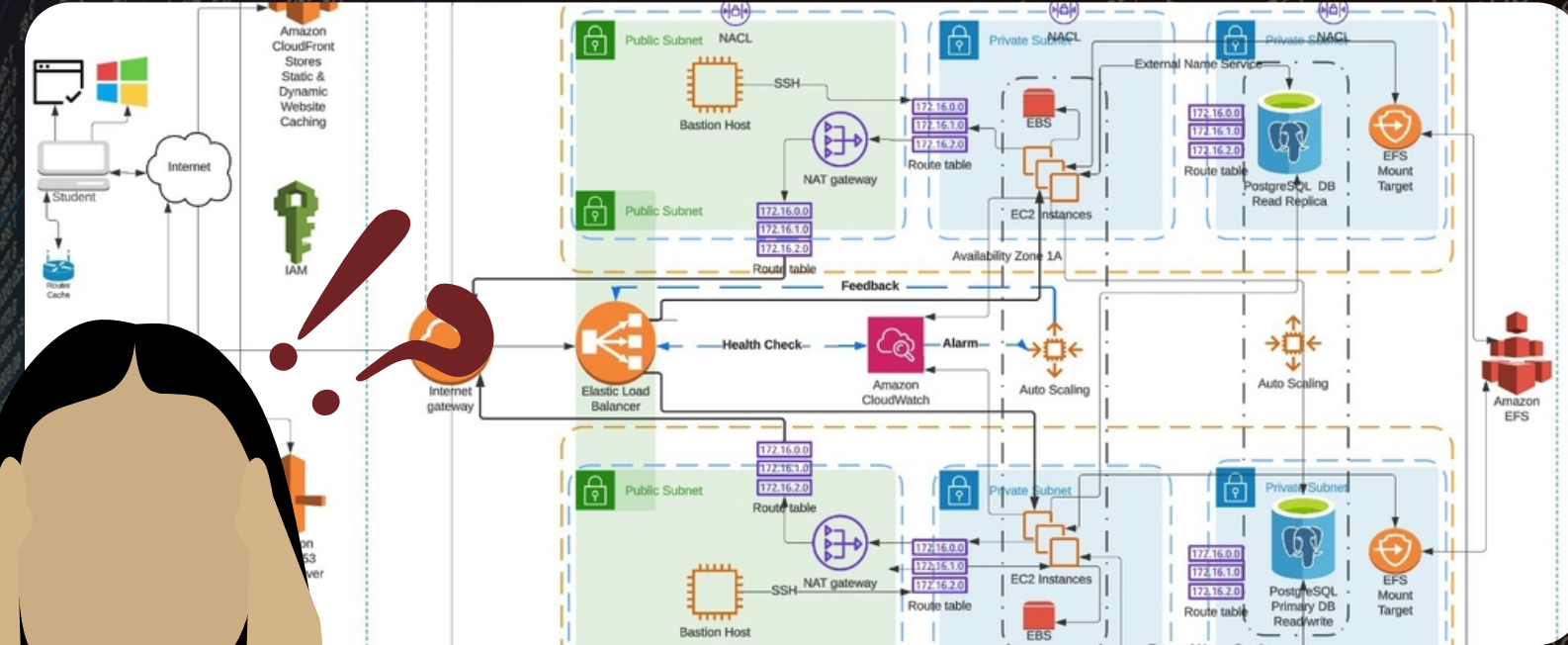
pythonProject1
  main.py

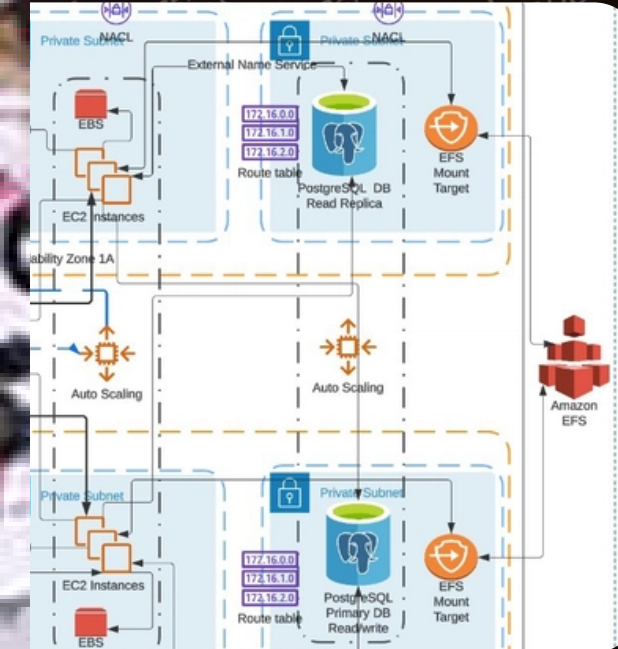
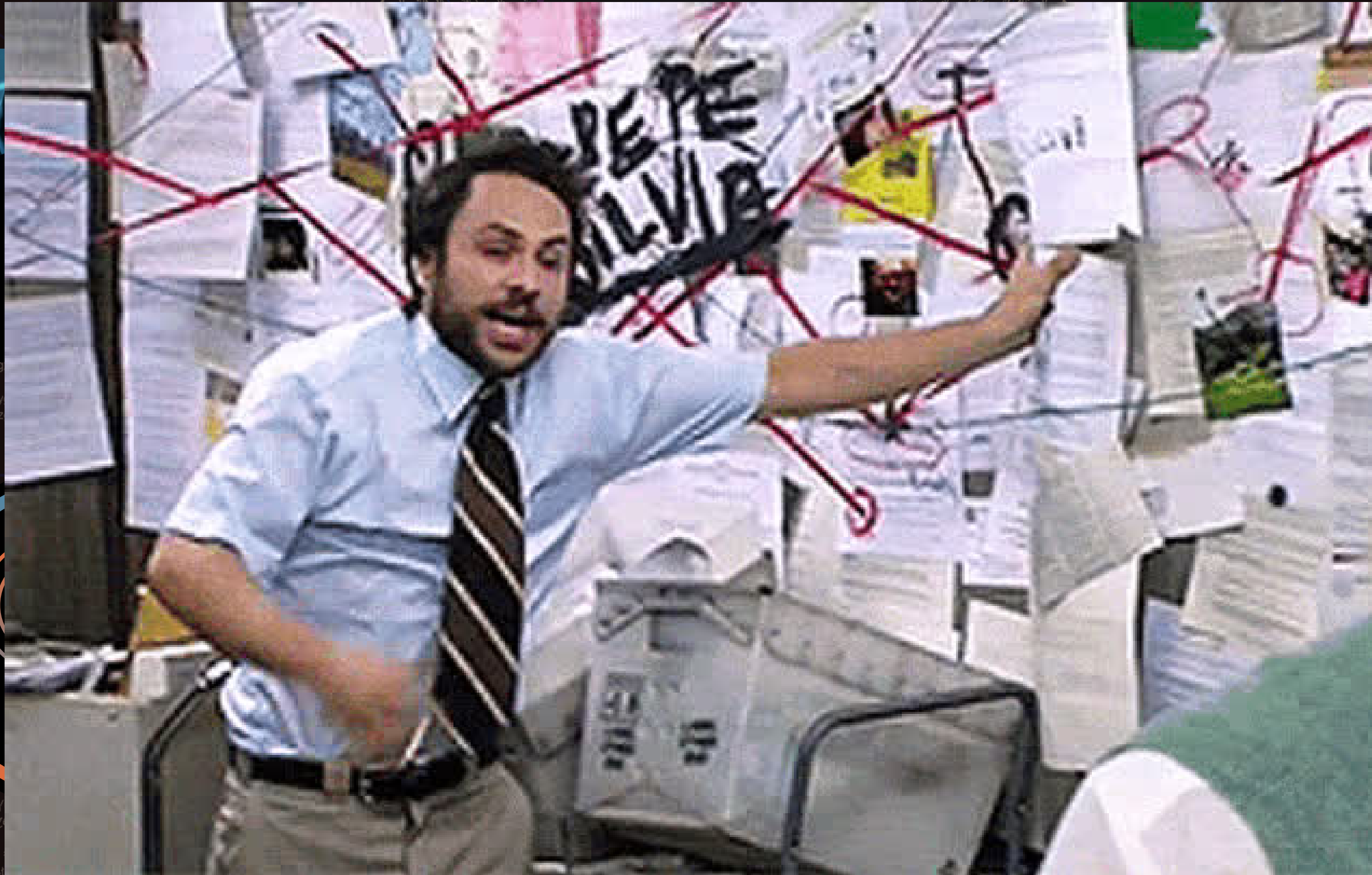
Project
  pythonProject1 C:\Users\storm\Documents\2020-21\CS395\pythonProject1
    venv library root
    main.py
  External Libraries
  Scratches and Consoles

Run: main
C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\Scripts\python.exe C:\Users\storm\Documents\2020-21\CS395\pythonProject1/main.py
2021-03-11 01:27:28.867716: W tensorflow/stream_executor/platform/default/dso_loader.cc:60] Could not load dynamic library 'cudart64_110.dll'; dlopen: cudart64_110.dll not found
2021-03-11 01:27:28.867890: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
Traceback (most recent call last):
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1/main.py", line 1, in <module>
    import tensorflow as tf
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\__init__.py", line 41, in <module>
    from tensorflow.python.tools import module_util as _module_util
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\__init__.py", line 48, in <module>
    from tensorflow.python import keras
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\keras\__init__.py", line 27, in <module>
    from tensorflow.python.keras import models
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\keras\models.py", line 26, in <module>
    from tensorflow.python.keras.engine import functional
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\keras\engine\functional.py", line 38, in <module>
    from tensorflow.python.keras.engine import training as training_lib
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\keras\engine\training.py", line 52, in <module>
    from tensorflow.python.keras.engine import data_adapter
  File "C:\Users\storm\Documents\2020-21\CS395\pythonProject1\venv\lib\site-packages\tensorflow\python\keras\engine\data_adapter.py", line 61, in <module>
    from scipy import sparse as scipy_sparse # pylint: disable=g-import-not-at-top
  File "C:\Users\storm\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.7_qbz5n2kfra8p0\localCache\local-packages\Python37\site-packages\scipy\__init__.py", line 130, in <module>
    from . import _distributor_init
  File "C:\Users\storm\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.7_qbz5n2kfra8p0\localCache\local-packages\Python37\site-packages\scipy\_distributor_init.py", line 61, in
    WinDLL(os.path.abspath(filename))
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.7_3.7.2544.0_x64__qbz5n2kfra8p0\lib\ctypes\__init__.py", line 364, in __init__
    self._handle = _dlopen(self._name, mode)
OSError: [WinError 126] The specified module could not be found

Process finished with exit code 1

Packages installed successfully: Installed packages: 'scipy' (3 minutes ago)
```





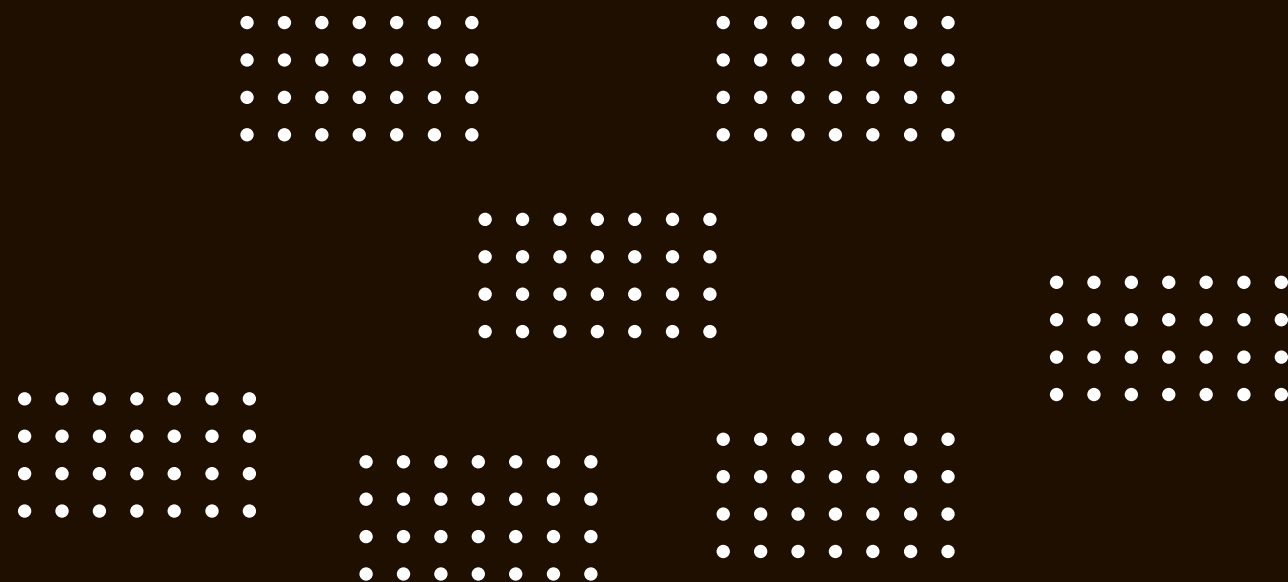
Monitoring

VS

Observability

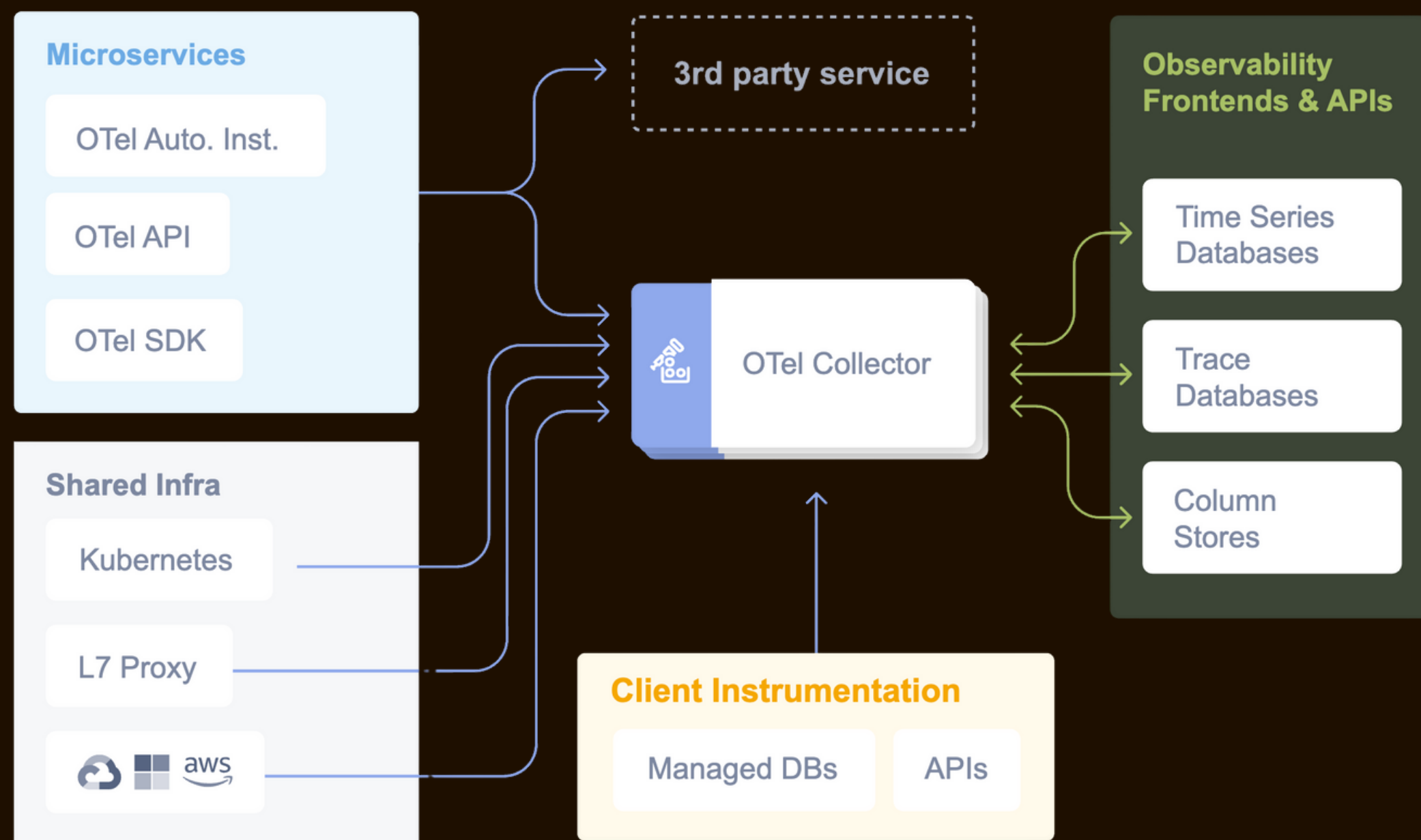
Observability

- Це не інструмент, а різниця між хаосом і контролем
- Це інженерна дисципліна має відповісти на 3 питання:
 - **Що** зараз коїться із системою?
 - **Чому** це коїться?
 - **Як** мені це виправити?



Observability підхід

- “Інструментувати” програми і інфраструктуру
 - через ENV змінні або SDK
- Зібрати, підкорегувати, зберегти дані
- Візуалізувати



Встановлення і запуск: Single node

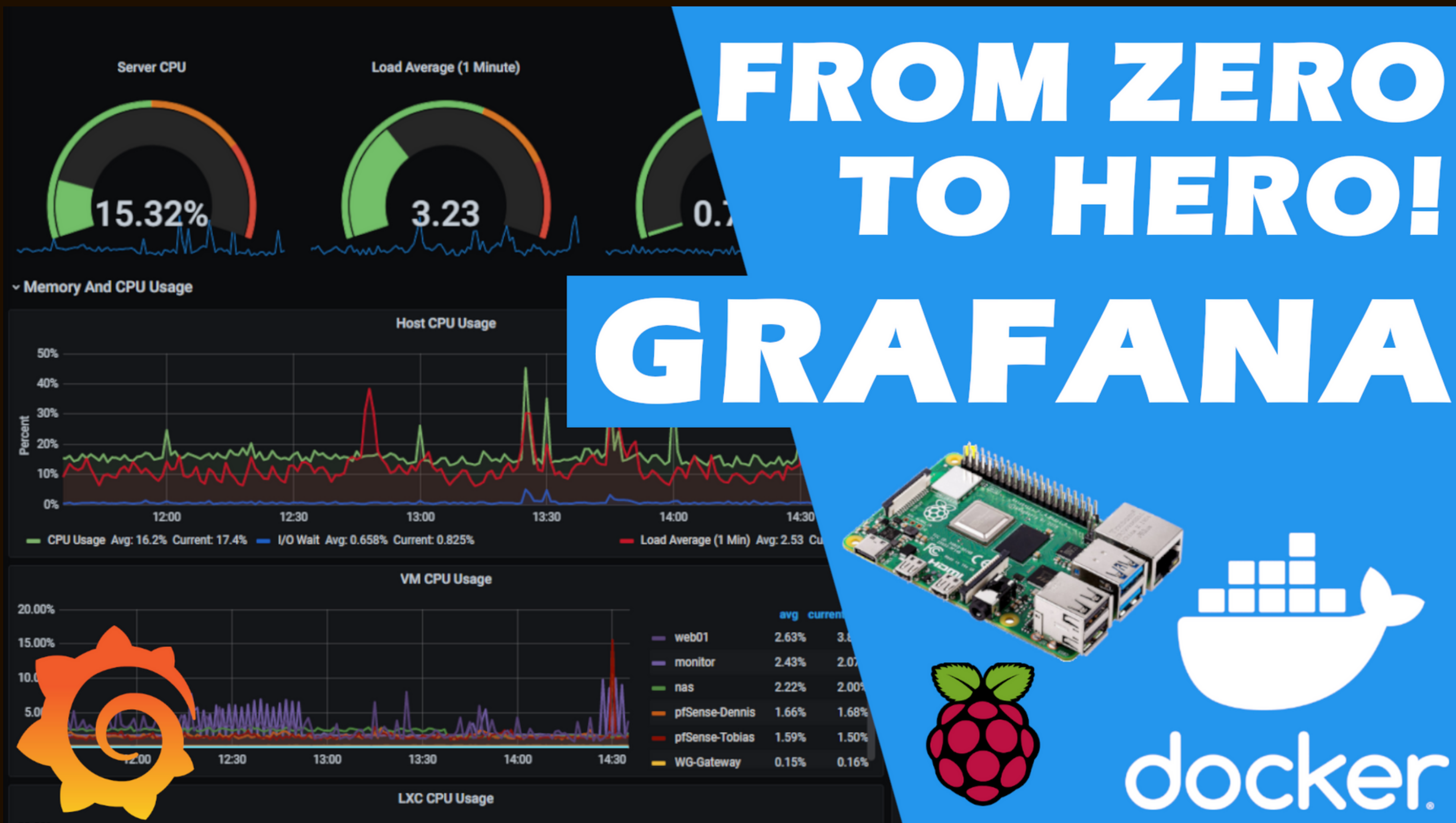
- На будь-якому ноуті чи Raspberry Pi
 - <https://docs.victoriametrics.com/victoriametrics/quick-start/#starting-vm-single-via-docker>

Starting VictoriaMetrics Single Node via Docker

Download the newest available [VictoriaMetrics release](#) from [DockerHub](#) or [Quay](#) :

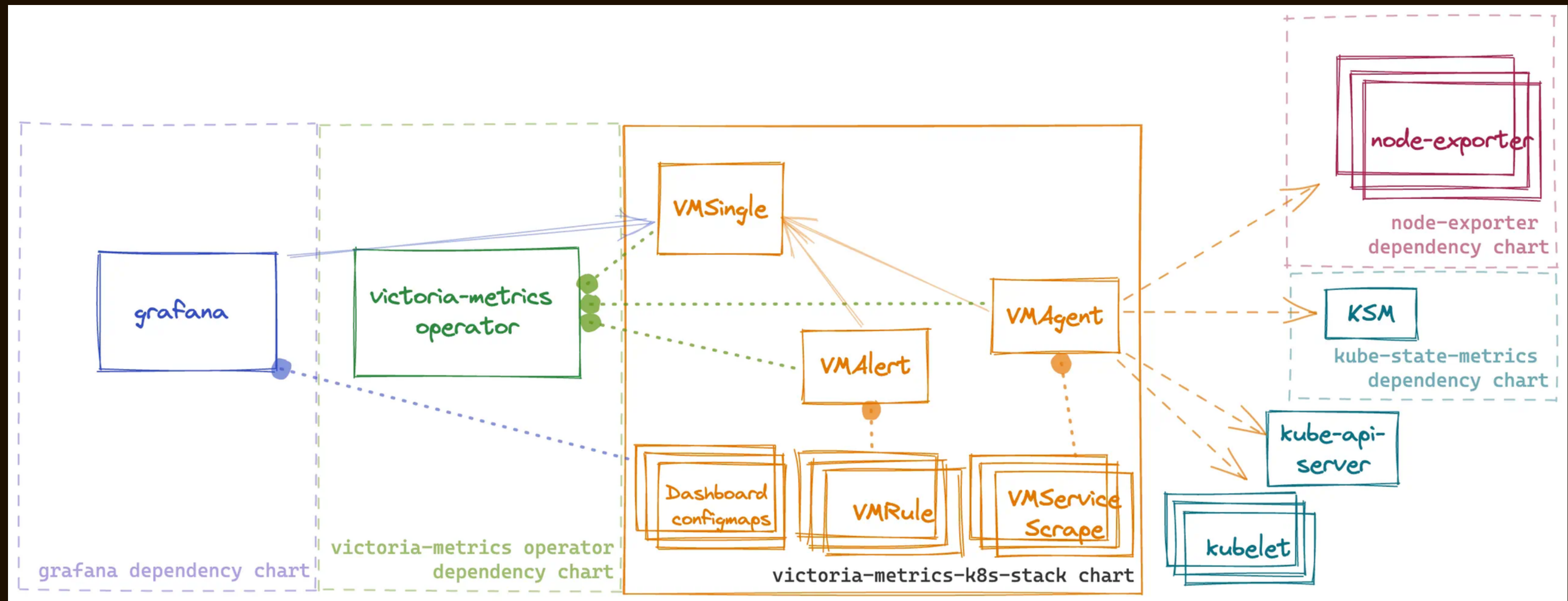
```
docker pull victoriametrics/victoria-metrics:v1.126.0
docker run -it --rm -v `pwd`/victoria-metrics-data:/victoria-metrics-data -p 8428:8428 \
victoriametrics/victoria-metrics:v1.126.0 --selfScrapeInterval=5s -storageDataPath=victoria-metrics-data
```

Встановлення і запуск: Single node

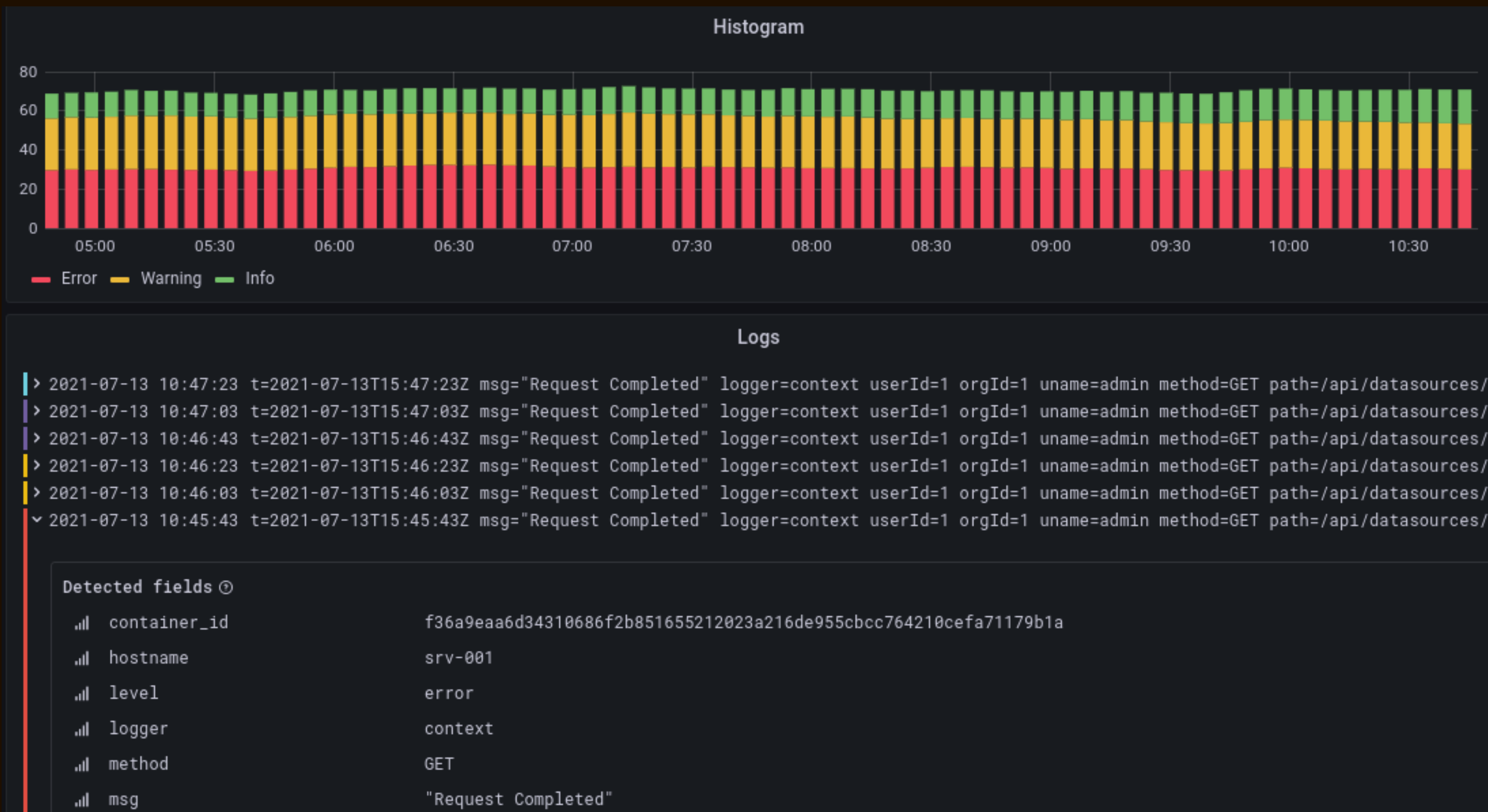


Встановлення і запуск: HA Cluster

- В хмарному відмовостійкому кластерному середовищі
 - <https://docs.victoriametrics.com/victoriametrics/cluster-victoriametrics>



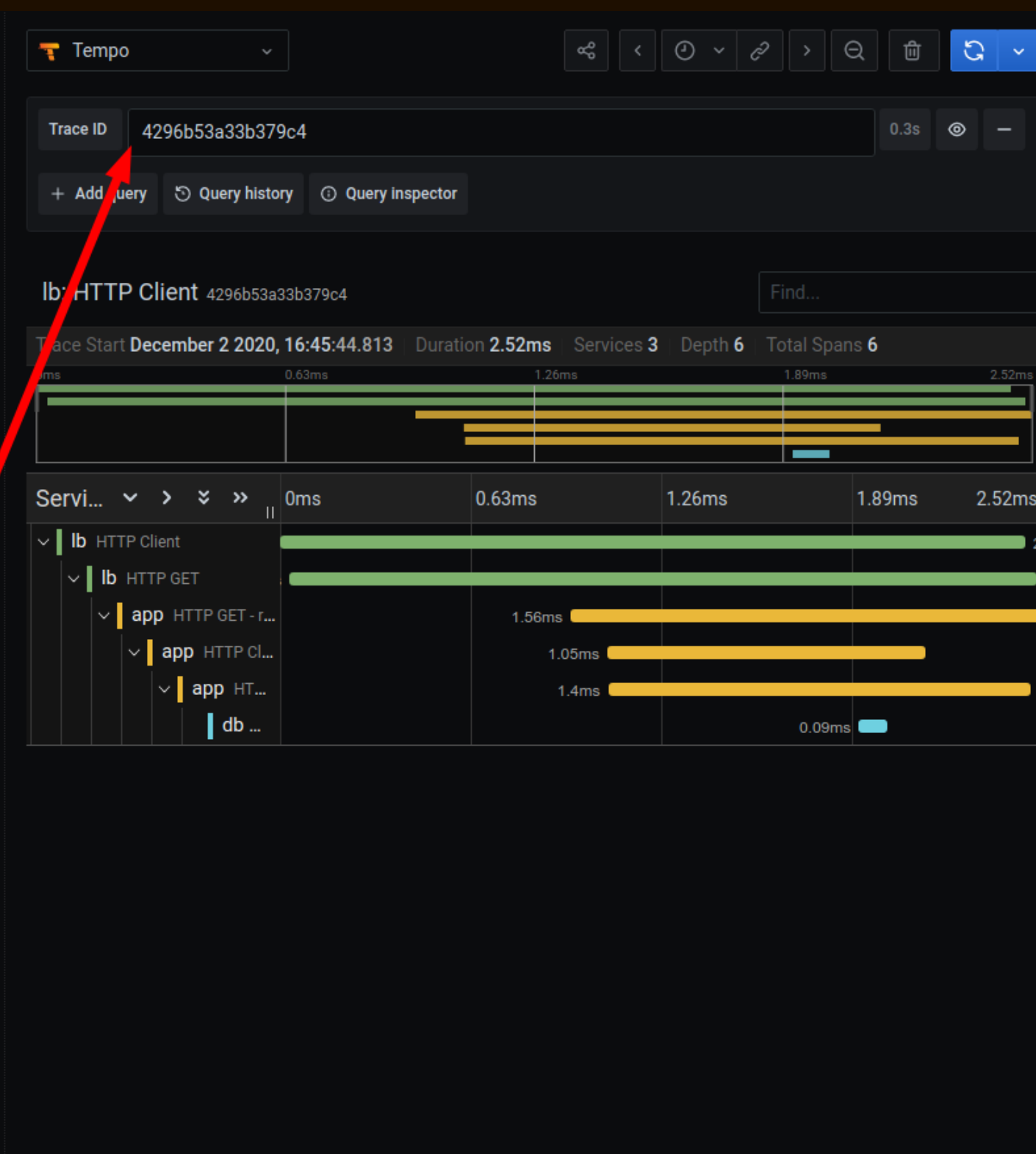
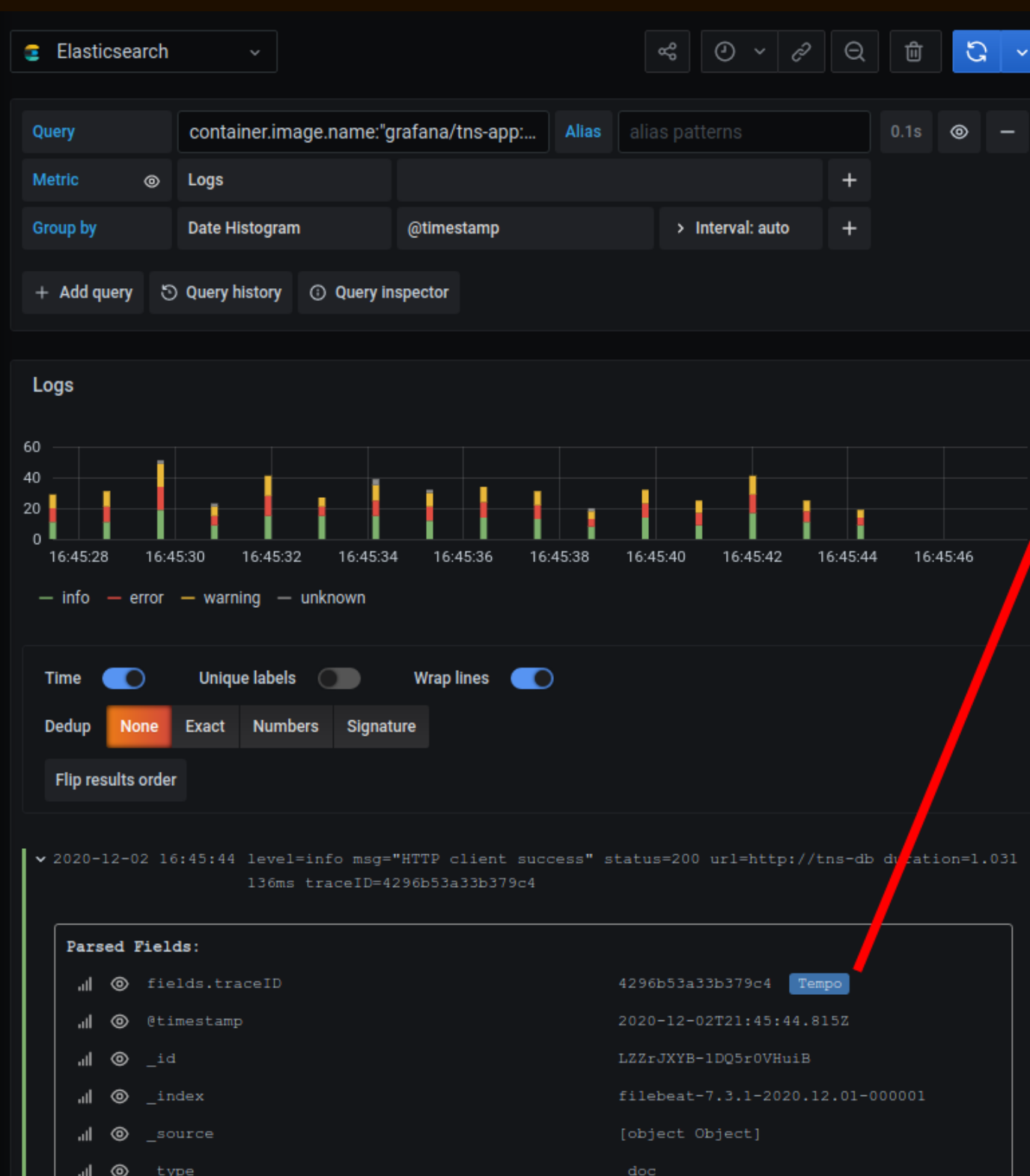
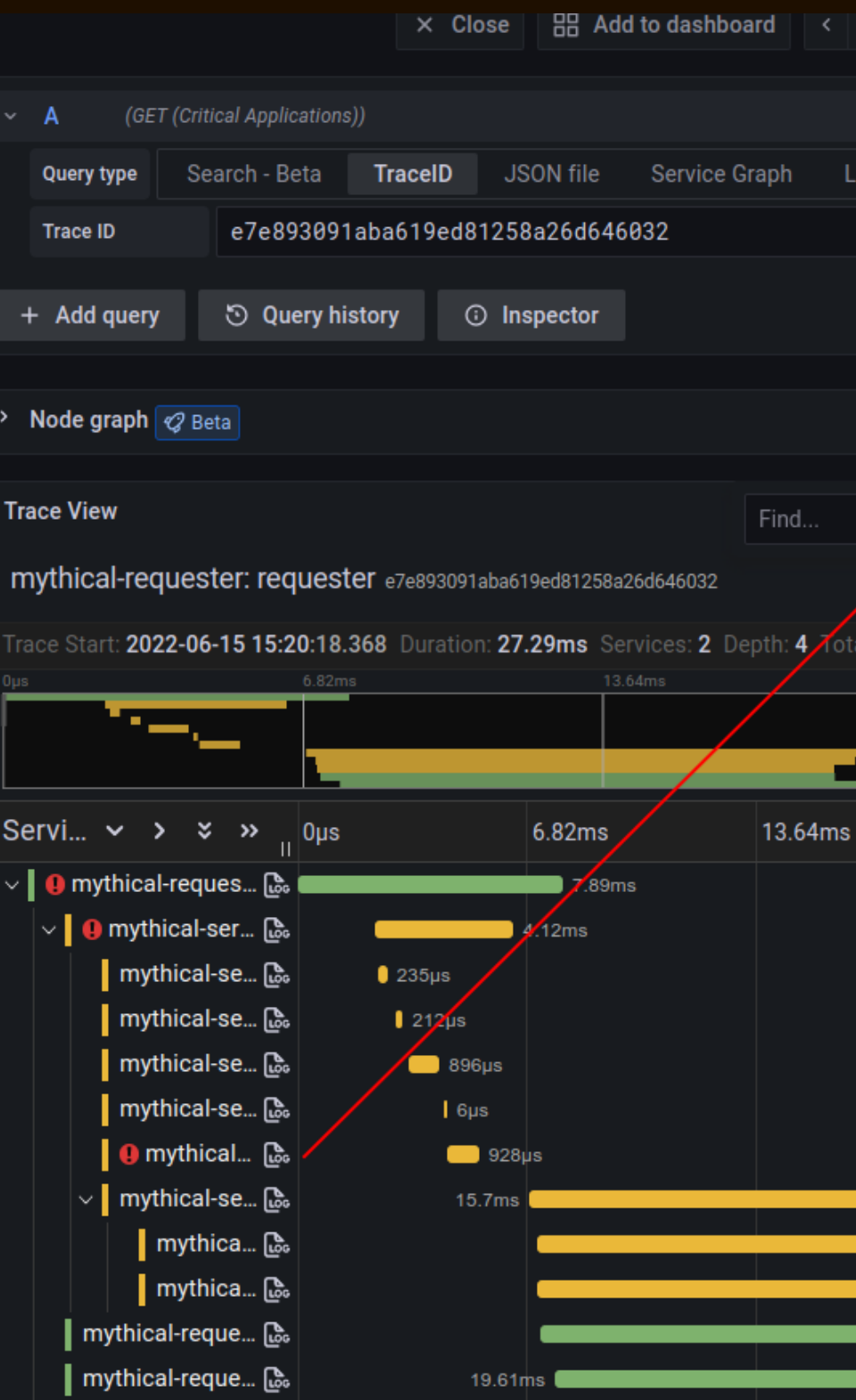
Стовпи Observability: Logs



Стовпи Observability: Metrics



СтоВПи Observability: Traces



Стовпи Observability: підсумок

- **Logs**
 - структурований json

- **Metrics**
 - ресурсні
 - програмні

- **Traces**

- **Events**
 - системні
 - програмні

- **Профілювання**

Мають бути тісно пов'язані між собою тегами контексту

Інструменти Observability

- збір і збагачення даних → Open telemetry
- візуалізація → Grafana
- сховище логів → Loki / Opensearch / VictoriaLogs
- трейси → Tempo / Jaeger / Zipkin

Observability практики



- sampling (вибірка)
 - прибрати частину успішних логів і трейсів
- downsample (агрегація) і архівування / ротація
 - старі метрики, логи, події
- трейс → знаходимо помилку → знаходимо метрики і логи
- маскувати сенситивні дані в логах



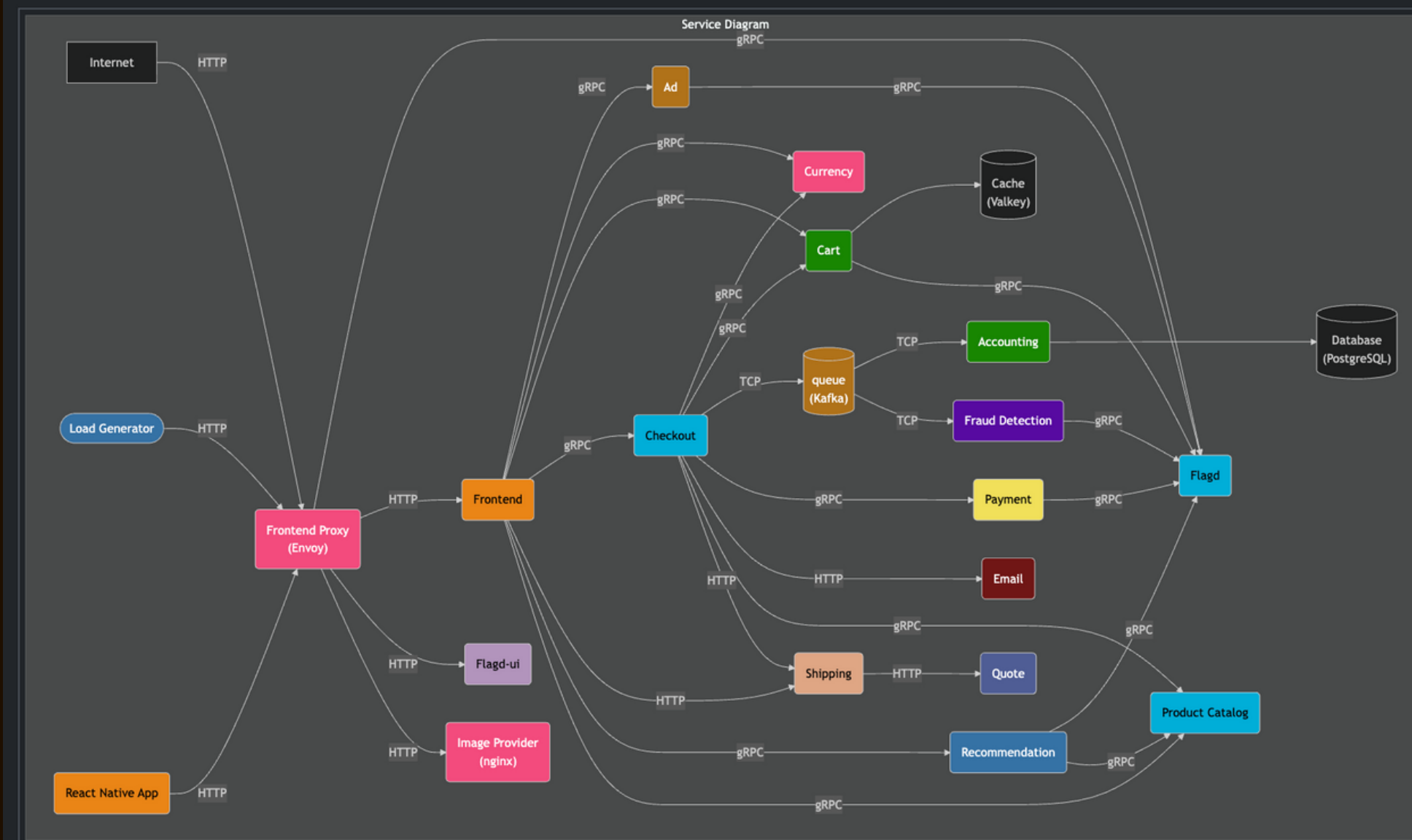
- Дешборди з купою графіків які ніколи не подивляться
- TBs неструктурованих логів (по loglevel type, service, request ids, infra і т.п.)
- Трейсинг кожного запиту з низькою кореляцією до помилок
- Отримання великої кількості безглузвих або хибно-позитивних сповіщень
 - #alert-servicename
 - #alert-servicename-critical
 - #alerts-servicename-critical-p0
 - #alerts-escalation-management

Що спробувати наступним

- <https://opentelemetry.io/docs/demo/architecture>

Demo Architecture

OpenTelemetry Demo is composed of microservices written in different programming languages that talk to each other over gRPC and HTTP; and a load generator which uses [Locust](#) to fake user traffic.

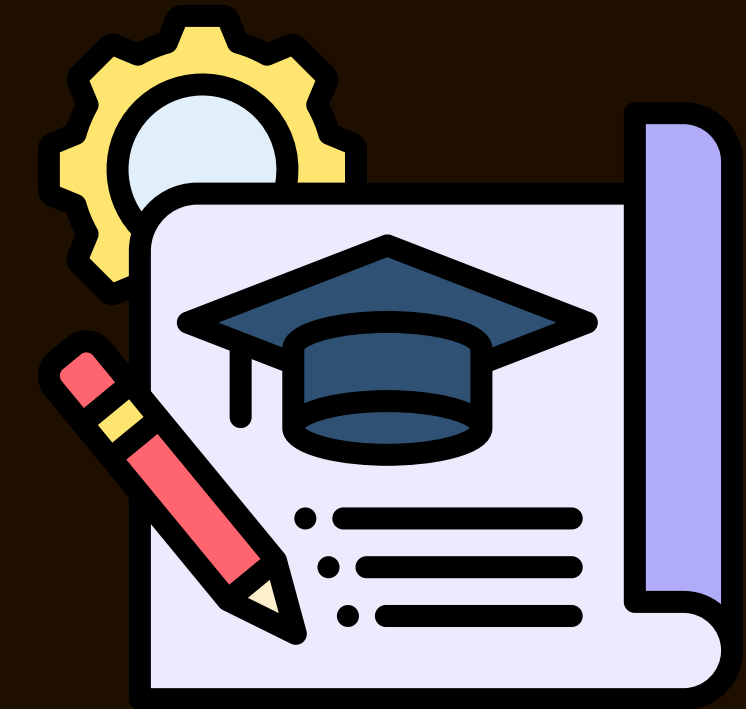


Service Legend



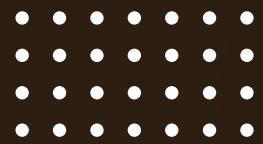
Що спробувати наступним

- Звернутись до мене за участю в університетських проєктах =)



Цікаве почитати

- <https://opentelemetry.io/docs>
- <https://opentelemetry.io/docs/demo/architecture>
- <https://grafana.com>
- <https://victoriametrics.com/case-studies/grammarly>
- <https://www.grammarly.com/blog/engineering>



Observability isn't just tools, it is designing systems to be debuggable.



THANK YOU!

