



Montreal

# Monitoring v2.0 avec Prometheus et Grafana



# Agenda



## Première partie

### Introduction à Prometheus et Grafana

- Présentation (15 min)
  - Parcours et historique
  - Survol de l'univers monitoring
  - Prometheus et Grafana
- Atelier (25 min)
- Pause (15 min)

## Deuxième partie

### Réalisation de Niveaux de Service

- Présentation (10 min)
  - Principes et objectif
  - Exemple
- Atelier (~)

# Référence



- Code pour la séance: <https://github.com/HugoLafleur/prom-toolkit>
- Documentation
  - [Prometheus \[Stockage | Configuration | Exporters | node\\_exporter\]](#)
  - [Grafana \[Provisioning\]](#)
- Bonnes lectures
  - [Heroic @ Spotify](#)
  - [Robust Perception Blog](#)
  - [Google SRE Book \[Monitoring | Service Level Objectives\]](#)
  - [CNCF Landscape \[Monitoring\]](#)



\$ /usr/bin/whoami



# Hugo Lafleur

Sr. Manager, Infrastructure | Octave Group



## Dossier

Name: Hugo Lafleur

Role: Making things work,  
making things happen

Likes: Tech, mountains, games

Dislikes: Le Métro

Quirk: Star Wars nerd

Weak against: Good beer



WE **ACTIVATE** THE SPACE THAT EXISTS BETWEEN  
BRANDS AND PEOPLE

Companies like these trust us to enhance their customer experiences.



We've created better ways for brands to engage audiences through content, audio visual systems, digital platforms, and more. How can we help yours?



Octave Music Group, Inc. is transforming the way businesses use music, content and technology to engage consumers in shared spaces. Our offerings are comprised of TouchTunes, the largest in-venue interactive music platform, and PlayNetwork, the leading global provider of music and branded media for customer experience. As of 2017, we reach more than 100 million consumers every day in 127+ countries around the world through 450+ brands, 185,000 locations and 2,500 local partners.

# TouchTunes

(Circa 1996-ish)



# PLAYNETWORK®

(Circa 1996)



# Parcours

1



# Évolution





# Exploration

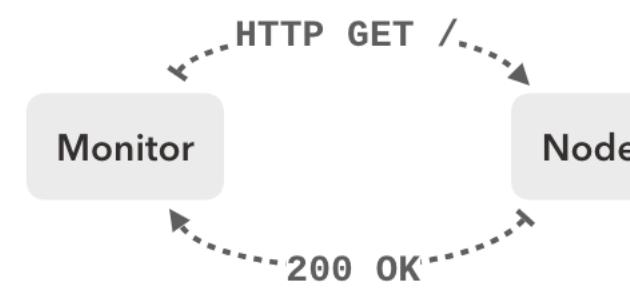
Écosystème et concepts

# Quelques contributeurs...

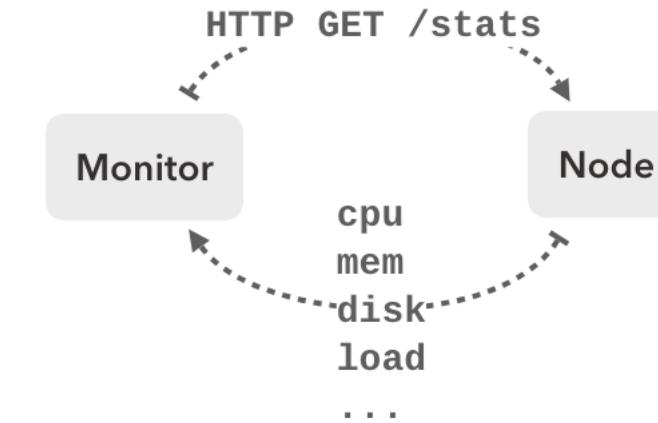


# Approche

## Sondes



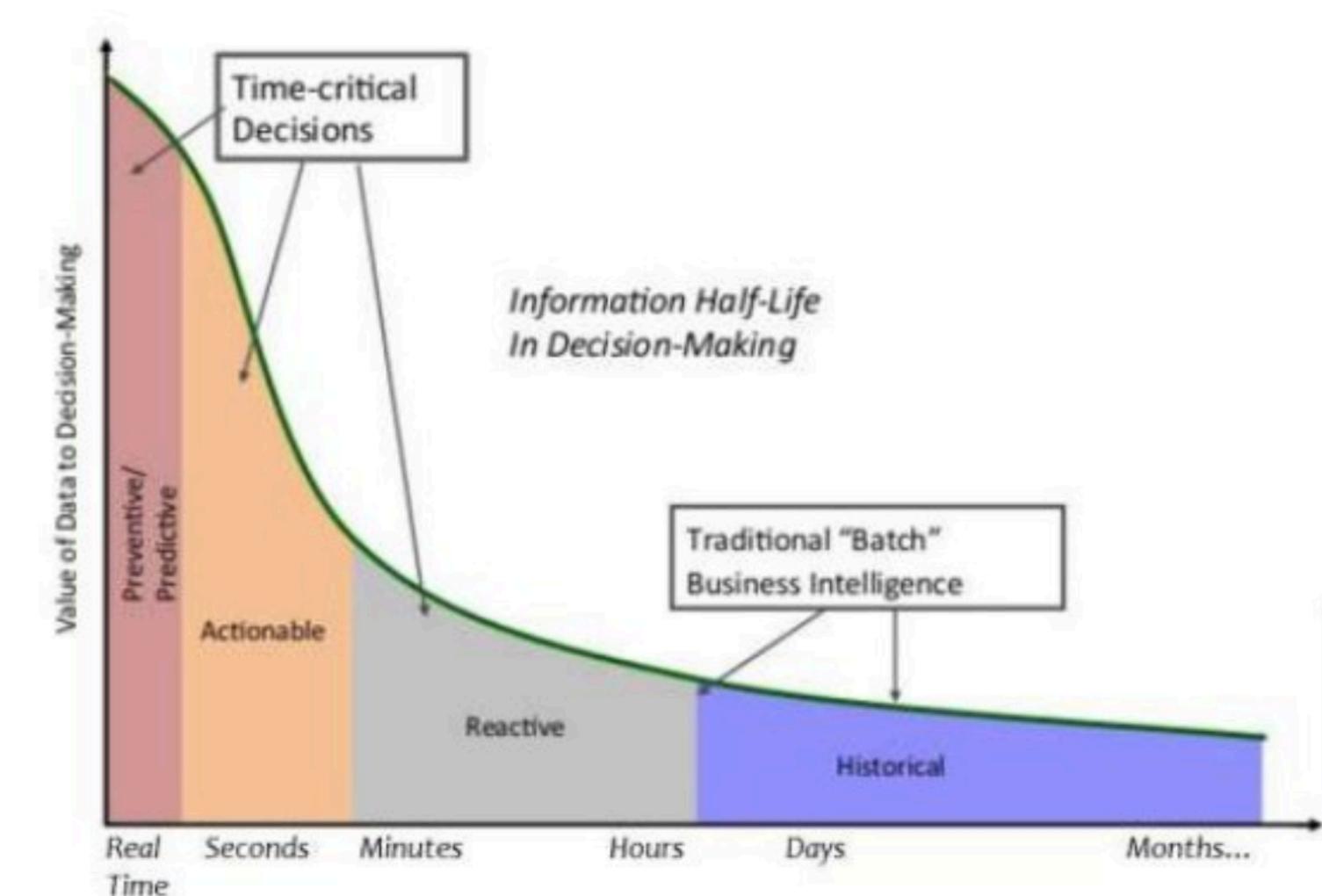
## Séries chronologiques



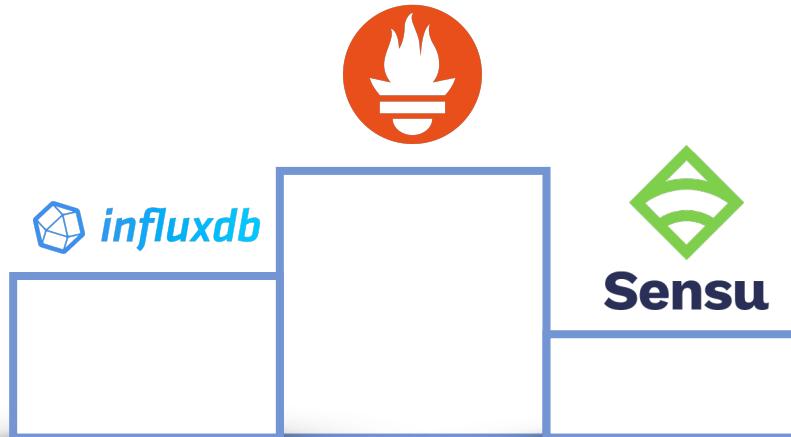


# Réception

Court ou long terme?



# Sélection



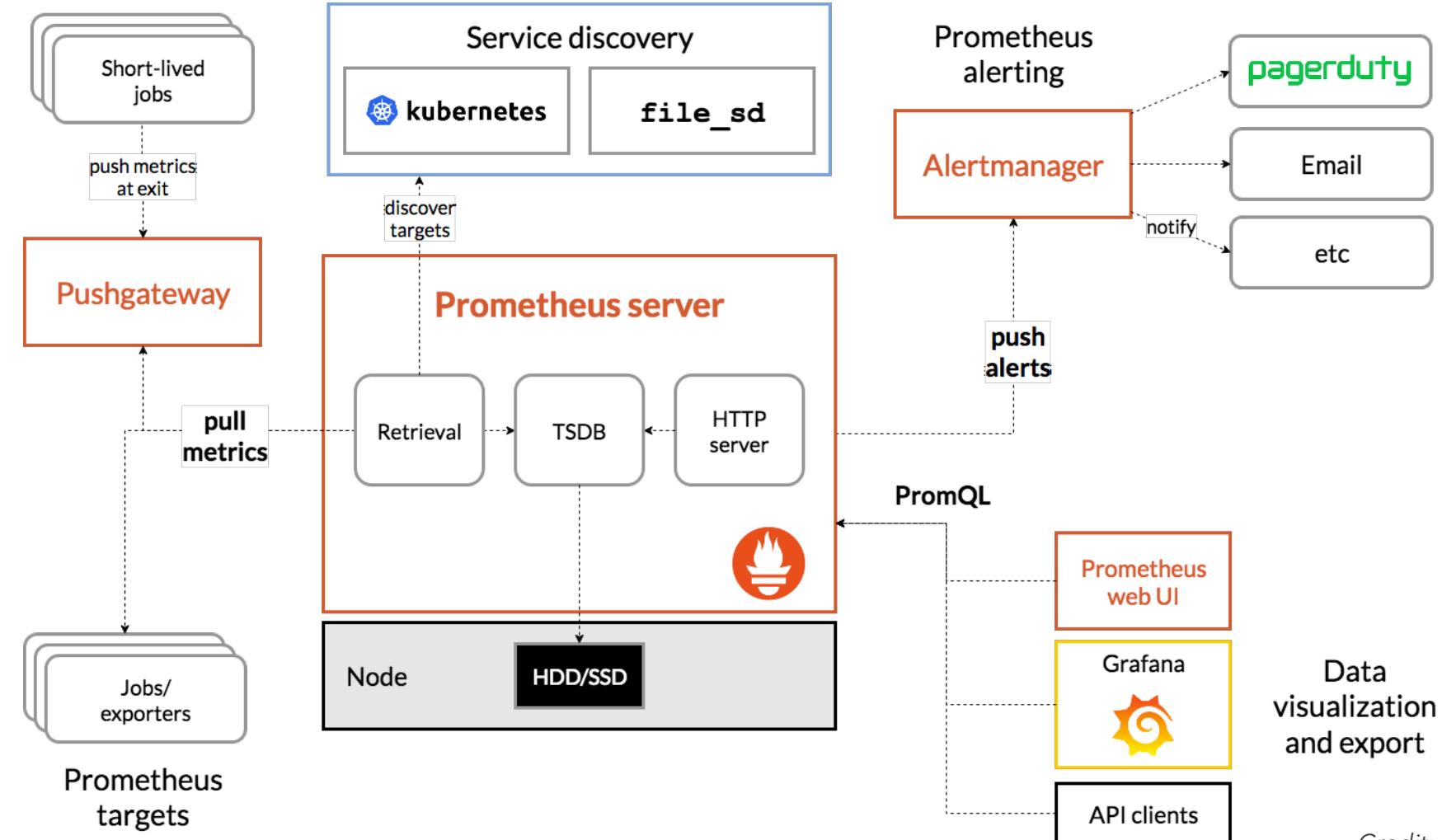
- Solution monitoring tout-en-un
- Déploiements simples et complexes
- TSDB performante
- Workflow intuitif
- Découverte de services robuste
- Intégrations avec stockage long-terme
- Expression de niveaux de service



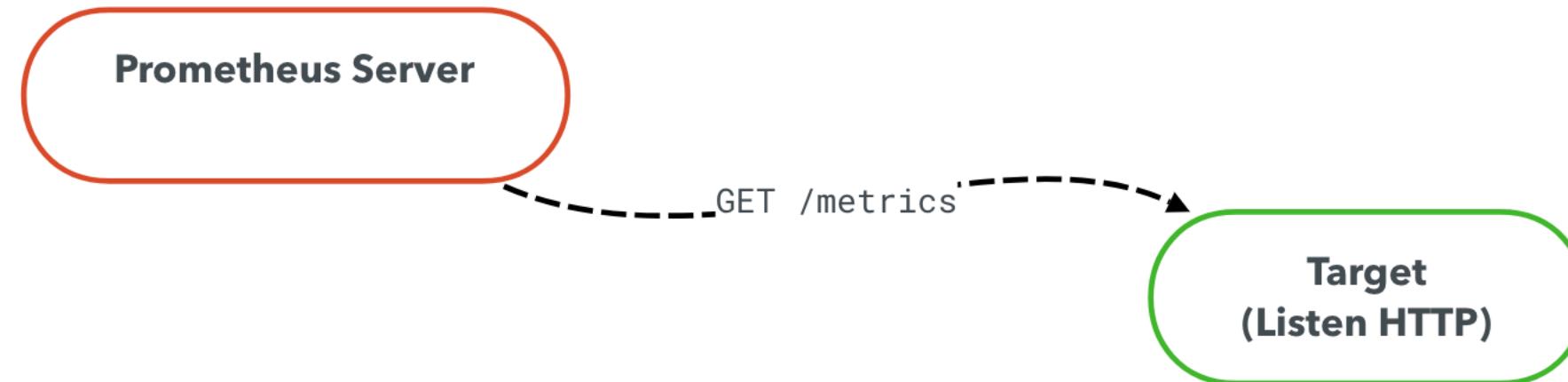
# Prometheus

Qu'est-ce que ça mange en hiver ?

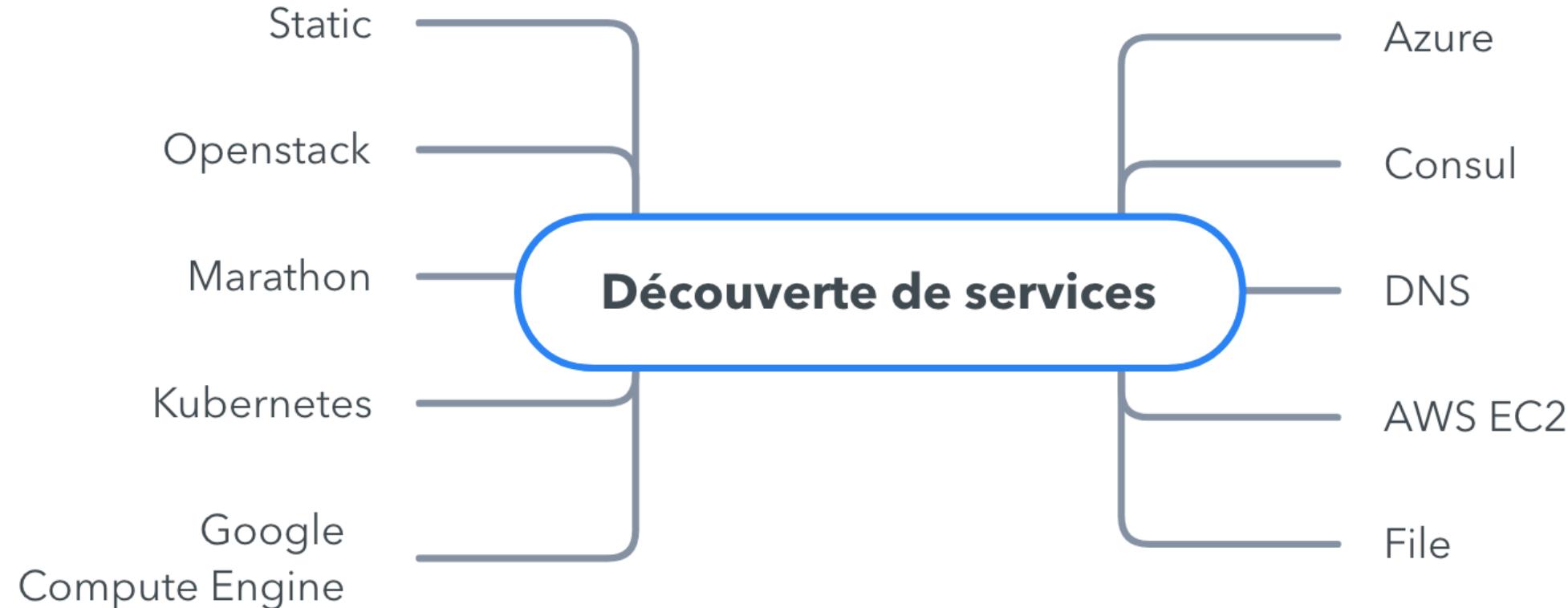
# Architecture et déploiement



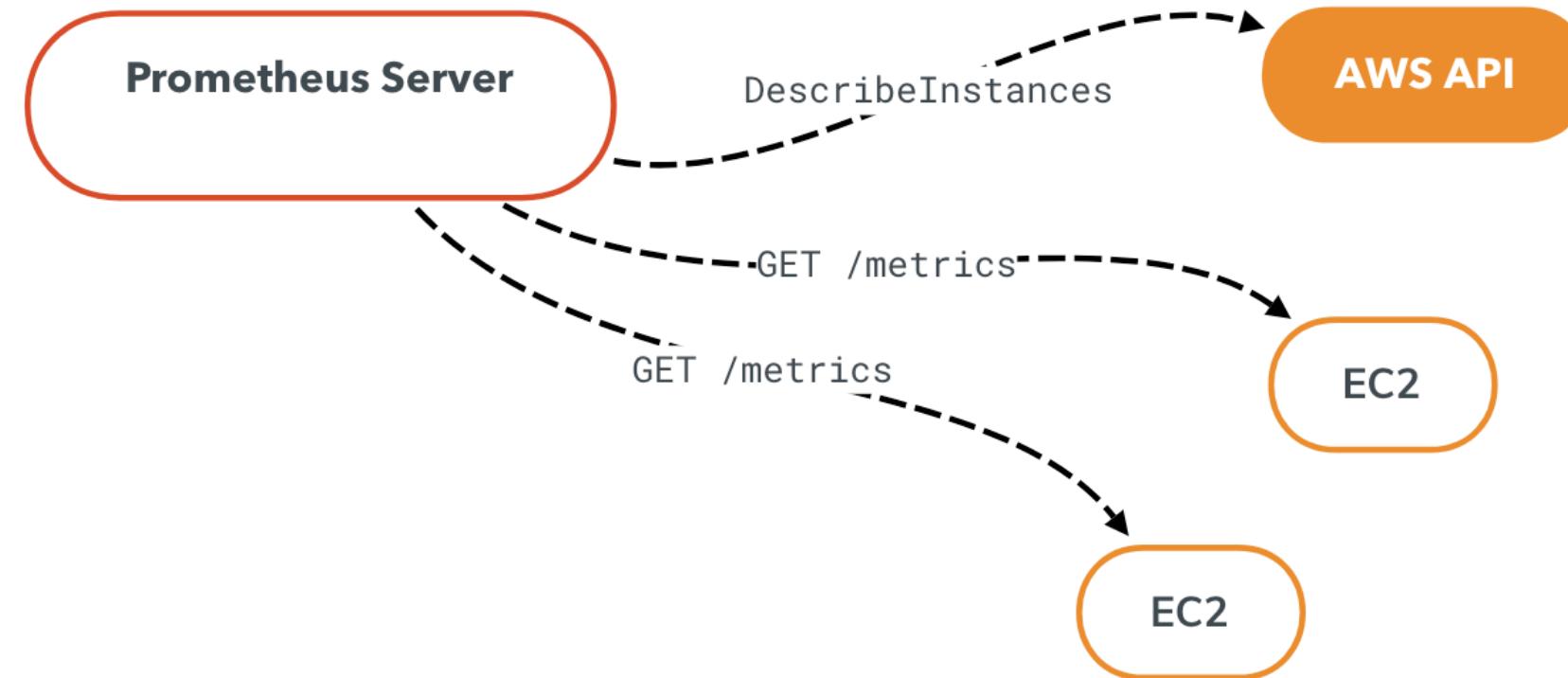
# Collecte



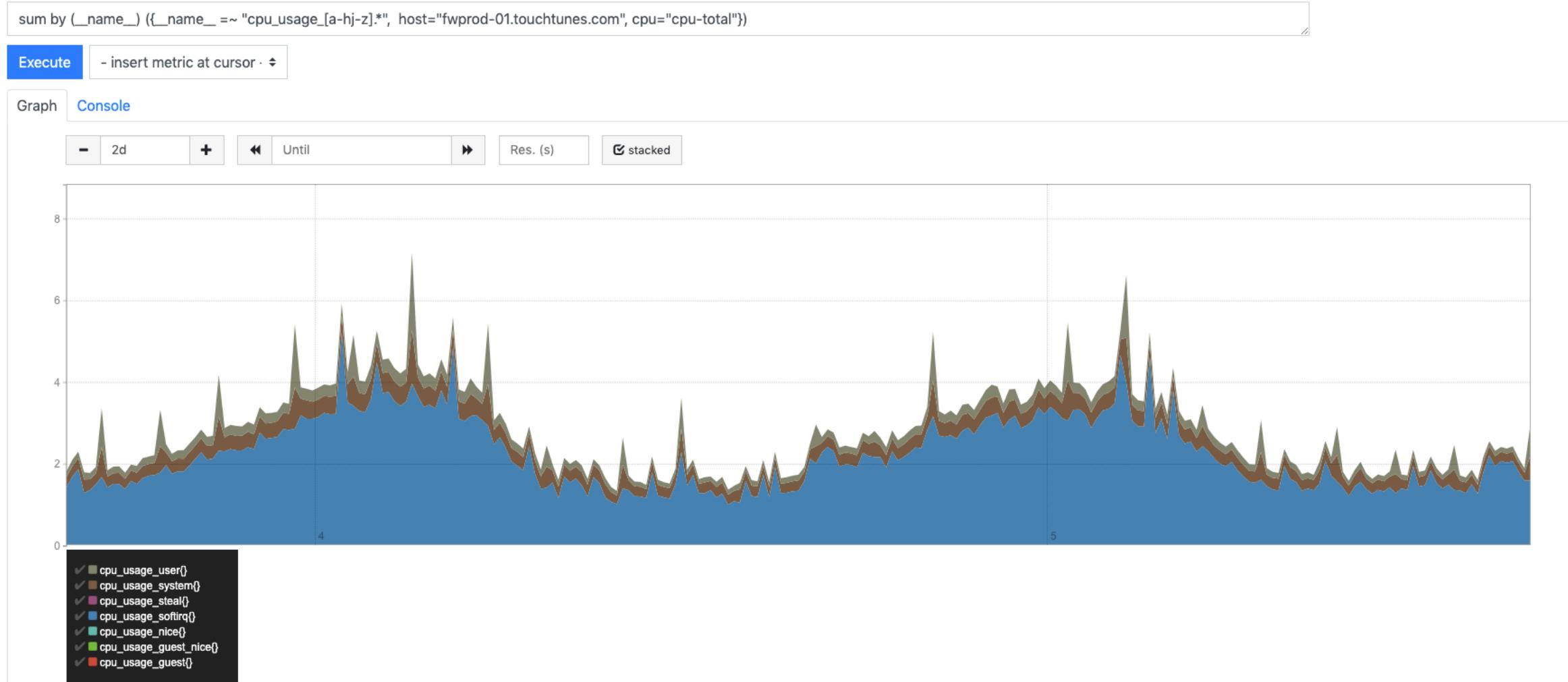
# Découverte de services



# Découverte de services (AWS)



# PromQL – Prometheus Web UI

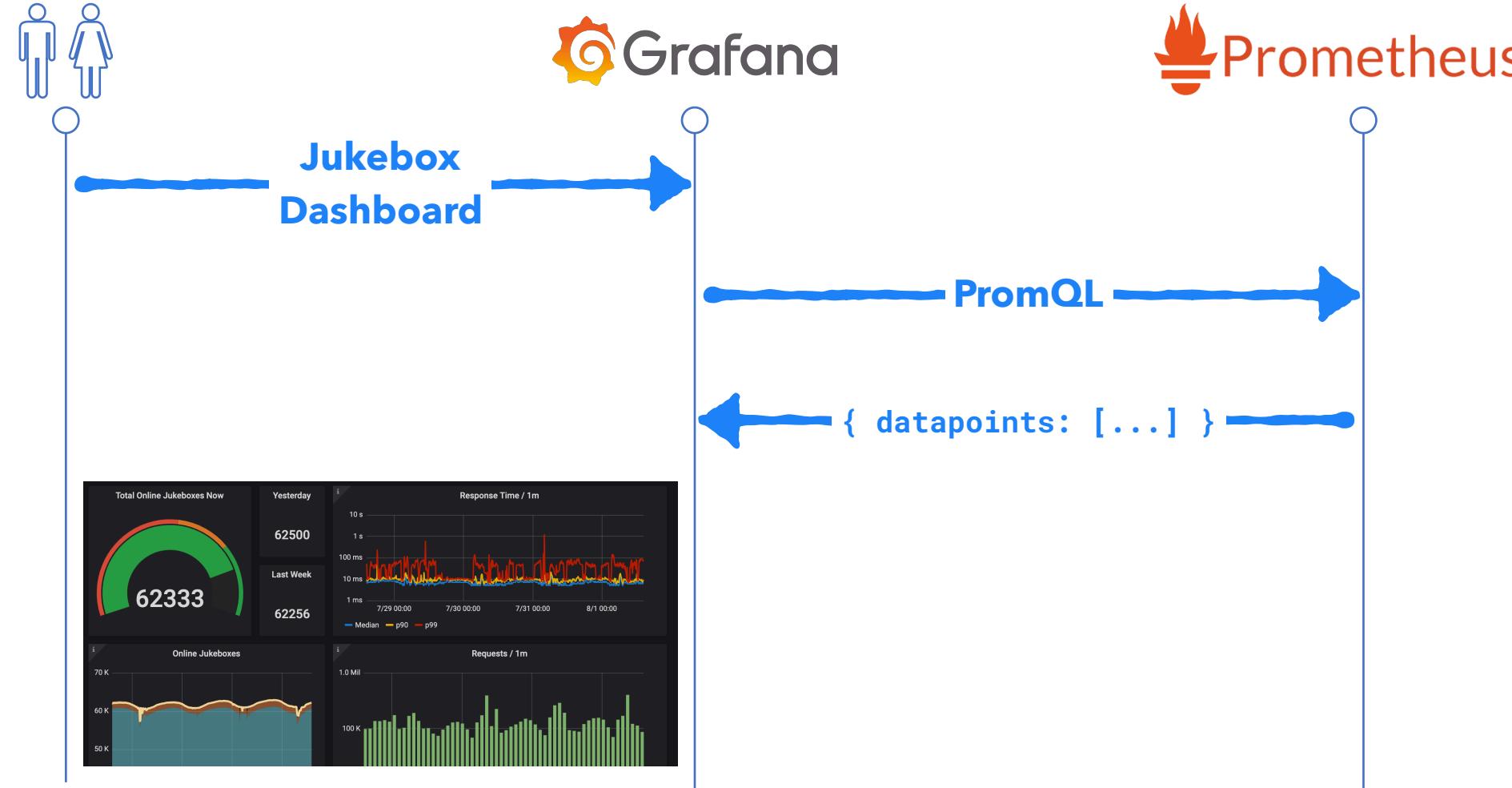




# PromQL – Grafana



# Visualisation





# Alertes et séries dérivées

*Flexibilité*

*Performance*

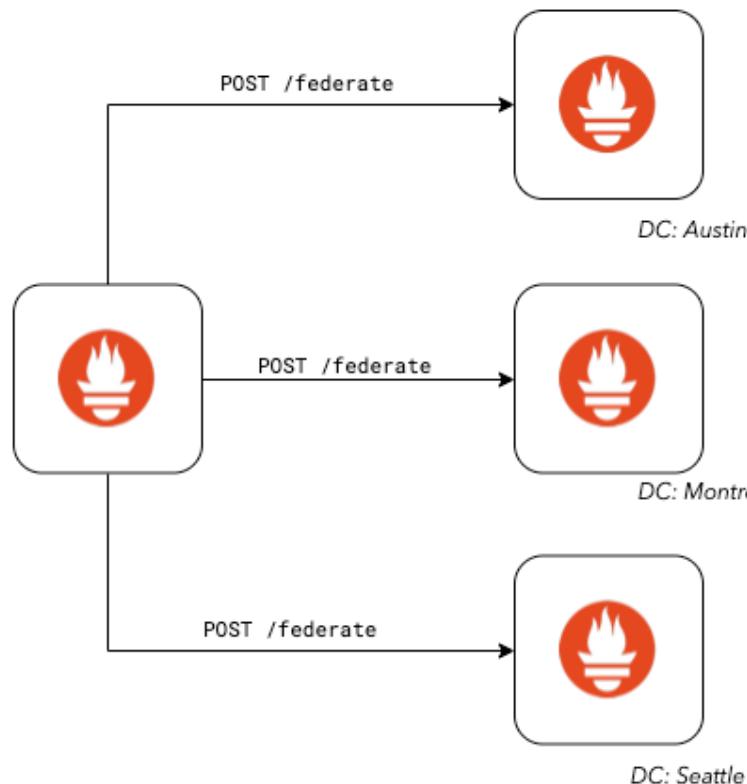
*Enrichissement*

*Notifications*

```
record: cpu_usage
expr: |
  max without(cpu, instance) (
    label_replace(
      {__name__=~"cpu.*",cpu="cpu-total"}, "mode", "$1", "__name__", "cpu_usage_(.*)"
    )
  )
```

```
alert: HOST_Unreachable_Or_Down
expr: up == 0
for: 30s
```

# Fédération



- Unification des données agrégées
- Distribution de la plateforme



- Alertes
- Copie intégrale des données



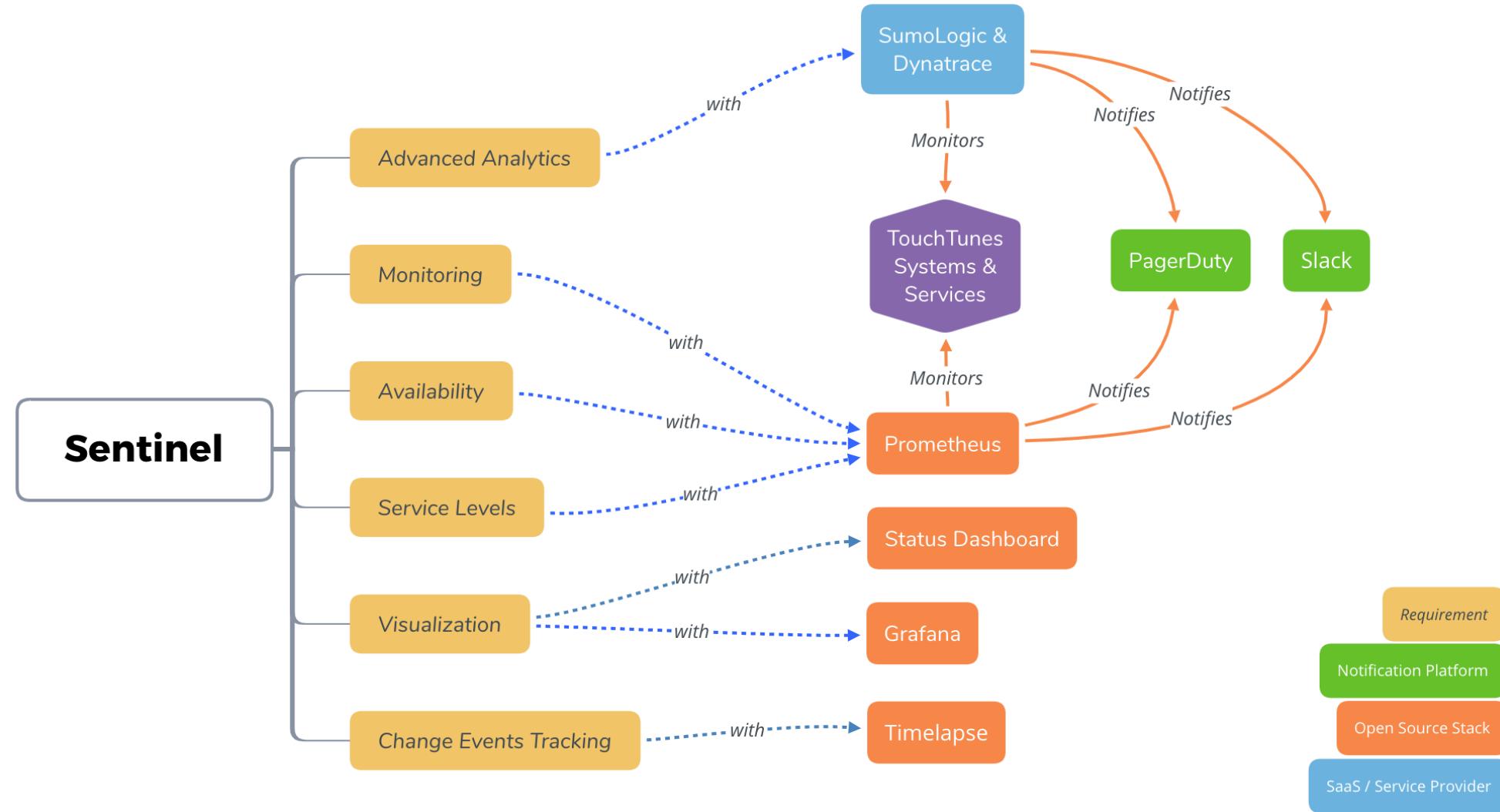
# Topo

Un aperçu de notre topologie

# Musique du monde

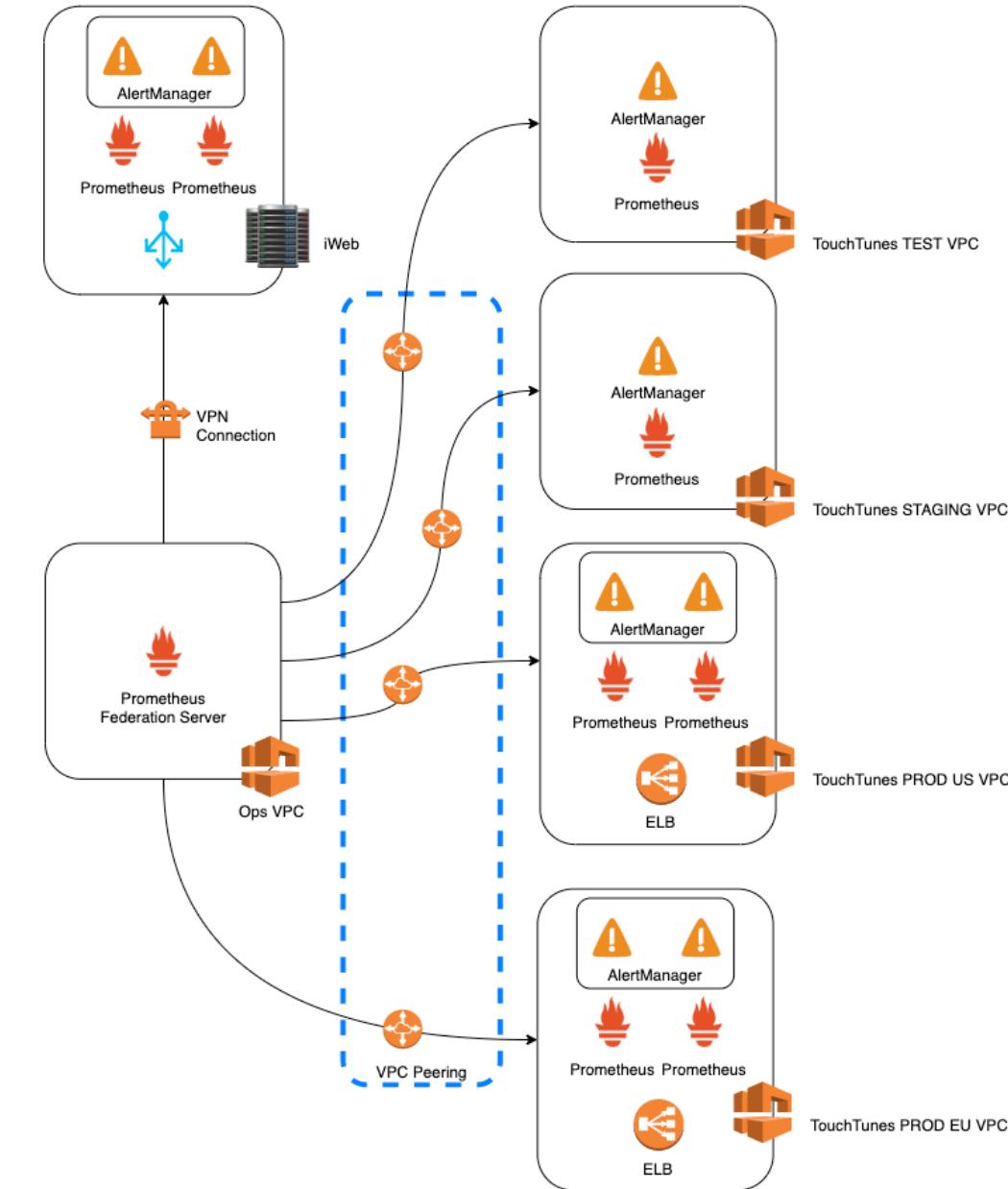


# Vision



# Prometheus

AWS  
*VPC Peering*  
*Instances locales*  
*Fédération des SLO*  
*Alertmanager en grappes*





# Atelier

# Référence



Code: <https://github.com/HugoLafleur/prom-toolkit>

# Découverte de services: Kubernetes

```
- job_name: kubernetes-nodes
  bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
  metrics_path: /metrics
  scrape_interval: 15s
  scrape_timeout: 10s
  scheme: https
  tls_config:
    ca_file: /var/run/secrets/kubernetes.io/serviceaccount/ca.crt
    insecure_skip_verify: false

  kubernetes_sd_configs:
    - role: node

  relabel_configs:
    - separator: ;
      regex: __meta_kubernetes_node_label_(.+)
      replacement: $1
      action: labelmap
    - separator: ;
      regex: (.*)
      target_label: __address__
      replacement: kubernetes.default.svc:443
      action: replace
    - source_labels: [__meta_kubernetes_node_name]
      separator: ;
      regex: (.)
      target_label: __metrics_path__
      replacement: /api/v1/nodes/${1}/proxy/metrics_path
      action: replacement

# Nom de la scrape_config (label "job")
# Bearer token pour contacter les API de K8s
# Path de metrics (défaut)
# Période de collecte (défaut)
# Attente max pour la collecte
# http/https
# Certificat pour contact les API de K8s
# ...
# ...

# Role des composantes à découvrir
#
# Manipulation de collecte / labels
# Sélection des labels qui commencent par
# "__meta_kubernetes_node_label_" et
# création/valuation de labels "(.)"
#
# Manipulation du endpoint pour la collecte
# Remplacement manuel du label __address__
# par kubernetes.default.svc:443
#
# Manipulation du endpoint pour la collecte
# Replacement par regex de la route pour
# récupérer les métriques correspondant aux
# noeuds
```

# PromQL



<b>metric name</b>	<b>label "host"</b>	<b>label "mode"</b>	1468433300	1468433360	1468433420	1468433480	1468433540	1468433600	1468433660	1468433720
cpu	server-01.example.com	user								
cpu	server-01.example.com	system								
cpu	server-02.example.com	user								
cpu	server-02.example.com	system								



# Pause

```
$ sleep 900
```



# Niveaux de service

# Niveaux de service

Concepts et définitions

## SLA: Service Level Agreement

*Un contrat, existant entre un fournisseur de service et un client, régissant les attentes opérationnelles et les conséquences en cas d'écart.*

## SLI: Service Level Indicator

*Une des dimensions utilisées pour qualifier un niveau de service.*

## SLO: Service Level Objective

*L'objectif de niveau de service cherche à établir une attente de fiabilité et de qualité.*

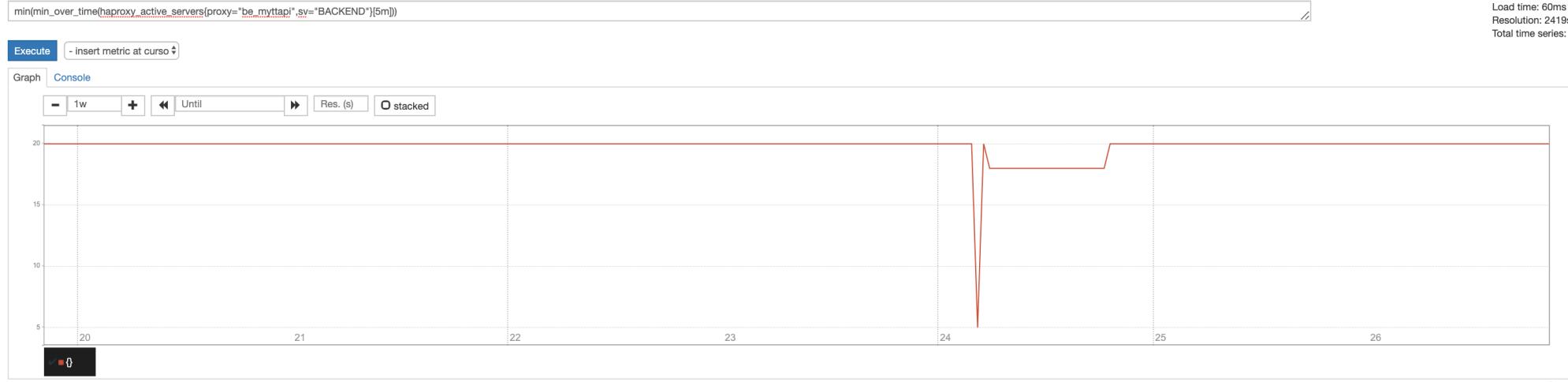
# Objectif de niveau de service (Exemple)



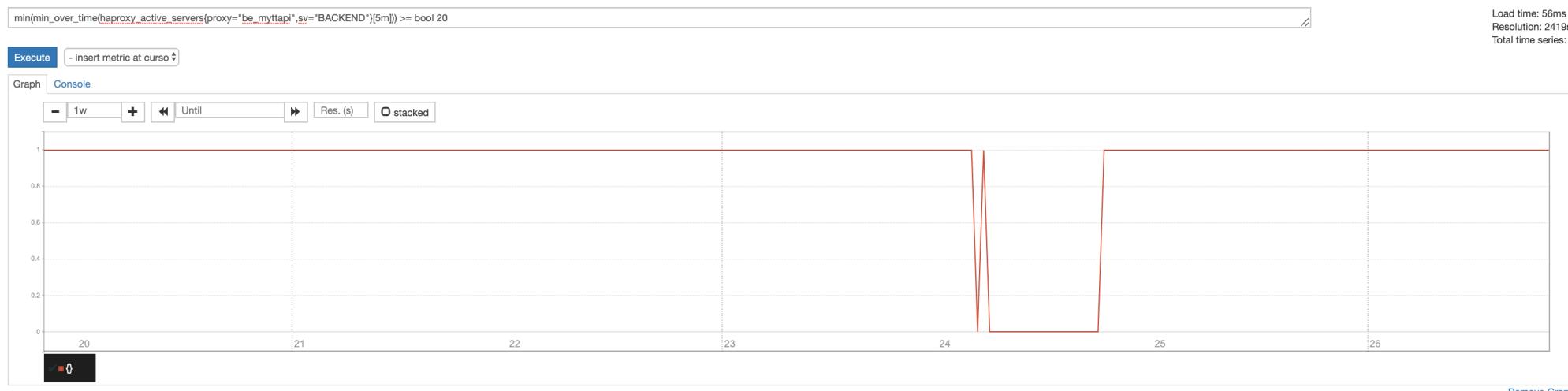
**Service: Meetup API; Objectif de niveau de service: 99%**

Métrique	Objectif	Raison
Nombre de serveurs disponibles	Minimum $\geq 20$ sur 5 minutes	Un volume minimum de serveurs est nécessaire pour soutenir la charge; une fréquence élevée de défaillances serveurs peut indiquer un problème
Volume d'erreurs (HTTP 500)	Maximum $\leq 0.1\%$ sur 5 minutes	Les erreurs HTTP 500 représentent une exception non gérée par le code du serveur, elles ont un effet direct sur l'expérience utilisateur
Latence (Temps de réponse de l'API)	Médiane $\leq 250ms$ sur 5 minutes	L'API devrait répondre rapidement; autrement, l'expérience utilisateur est affectée
Latence (Temps de réponse de l'API)	90 <sup>e</sup> percentile $\leq 500ms$ sur 5 minutes	Des temps de réponse élevés peuvent se produire occasionnellement, mais on veut s'assurer d'une qualité d'expérience minimale.

# Nombre de serveurs

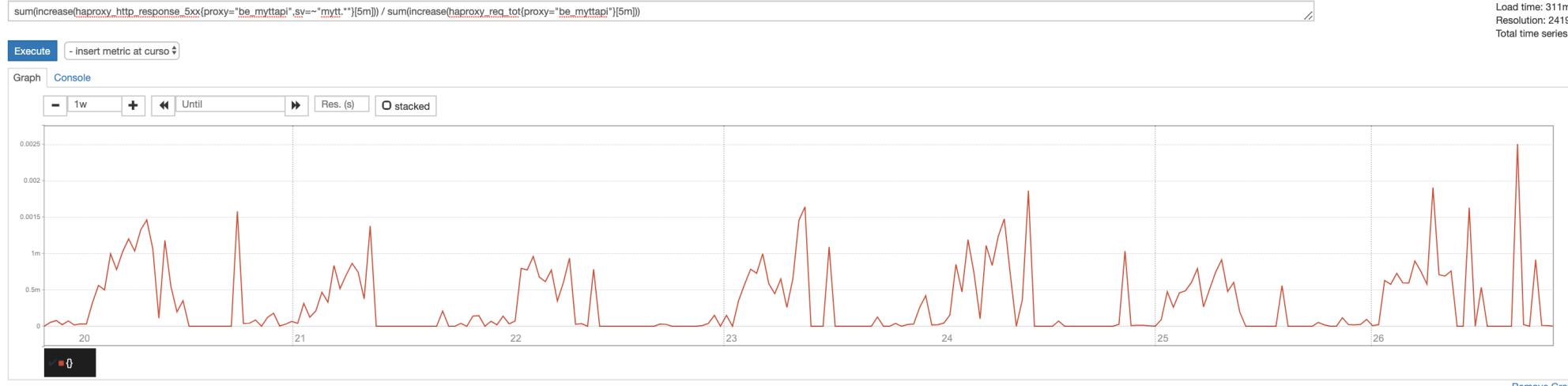


Métrique

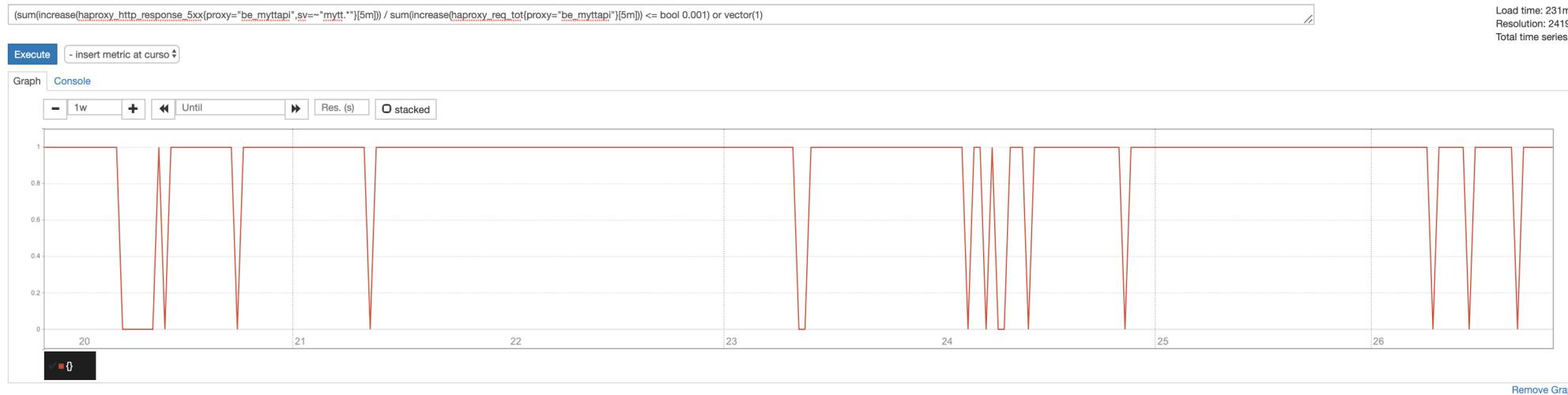


SLI

# Volume d'erreurs

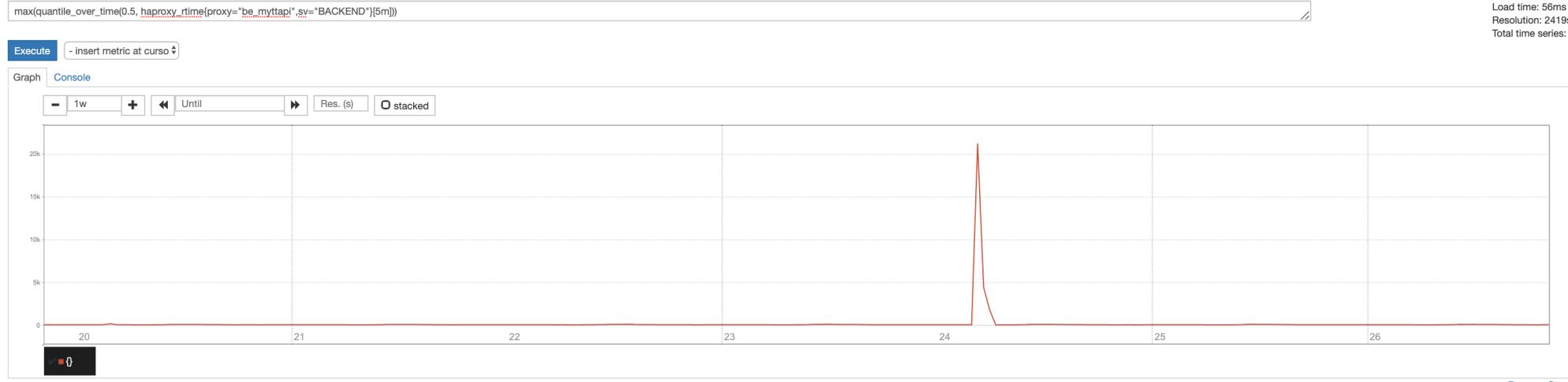


Métrique

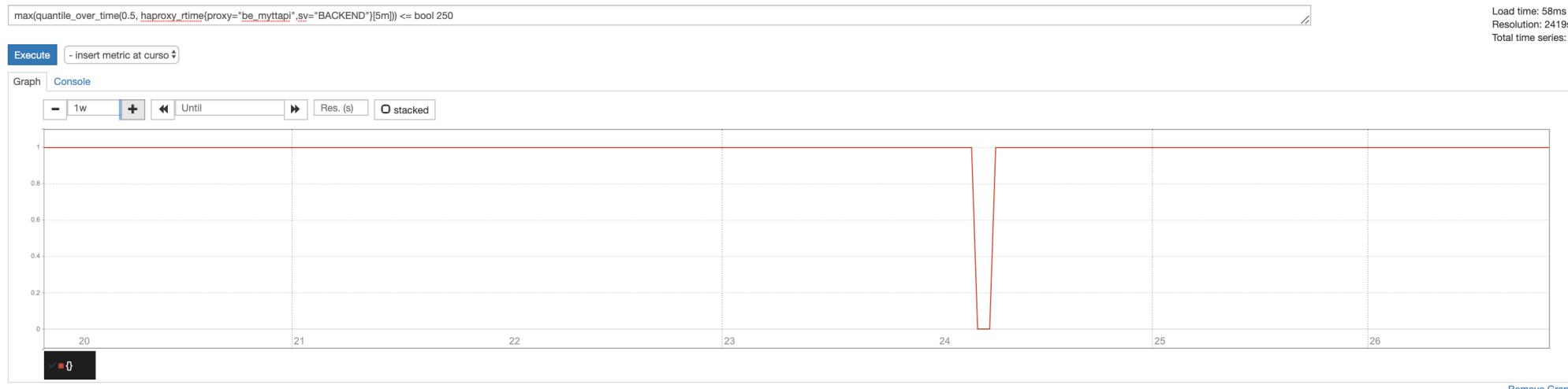


SLI

# Latence (médiane)

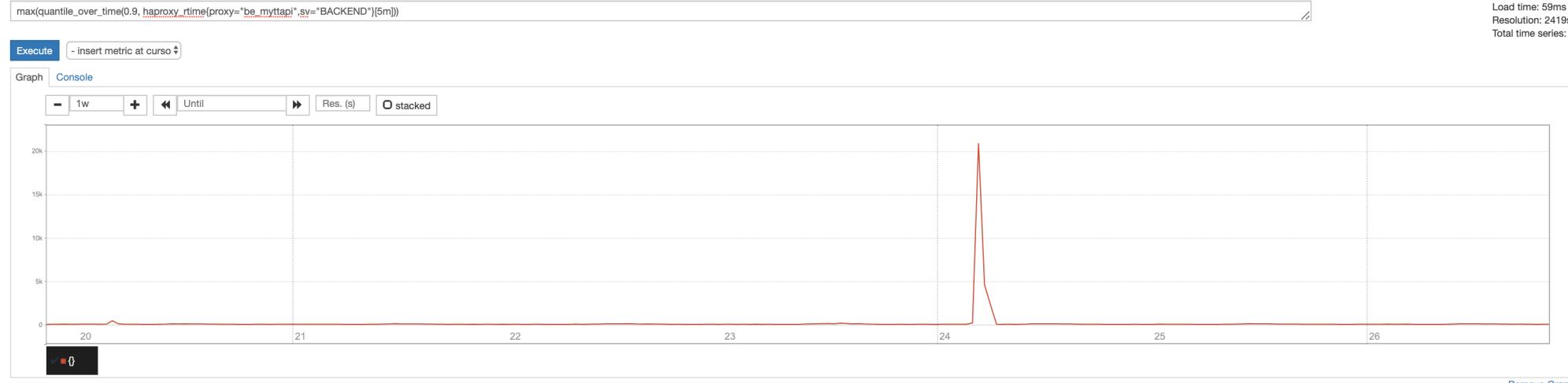


Métrique

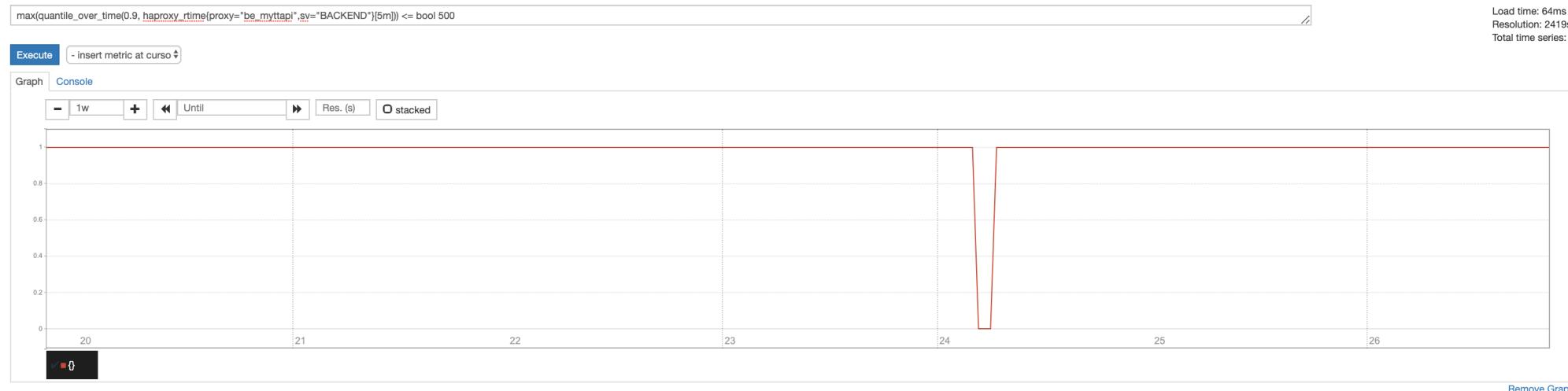


SLI

# Latence (90<sup>e</sup> percentile)

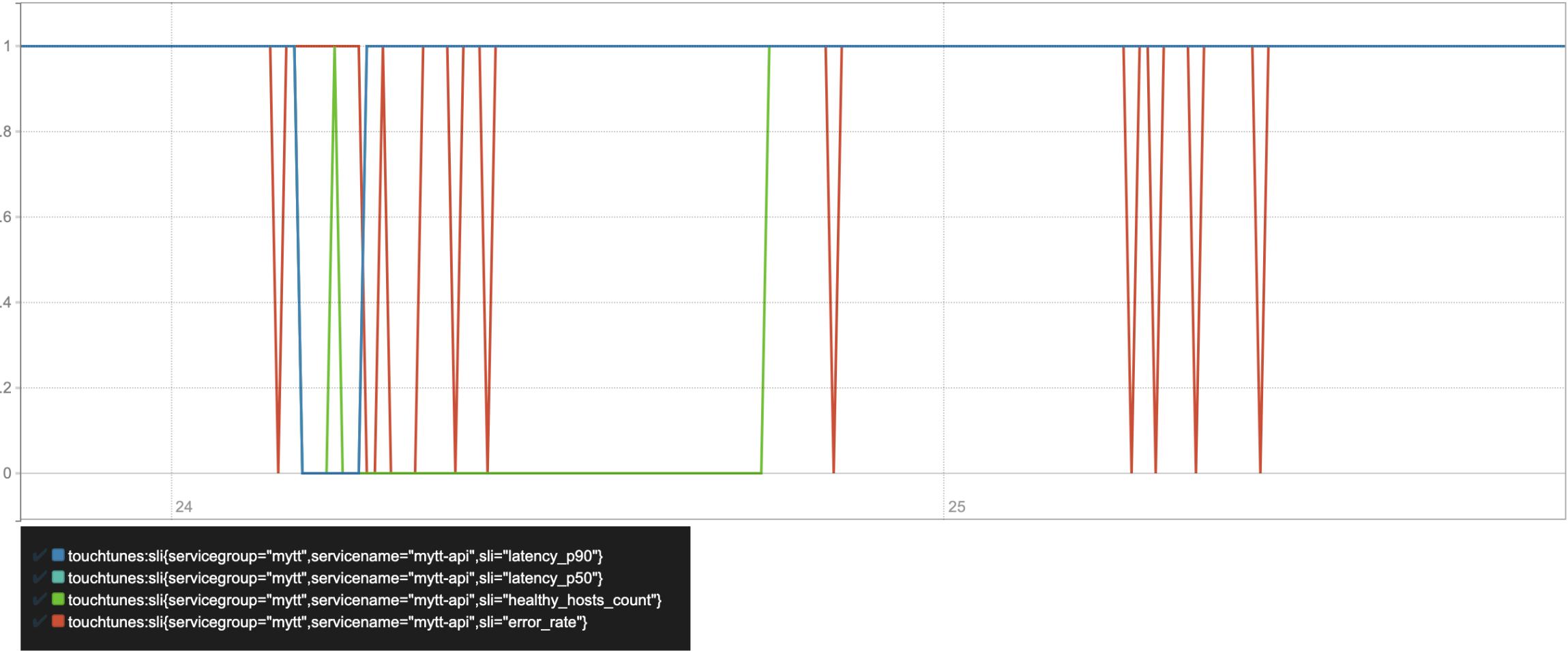


Métrique



SLI

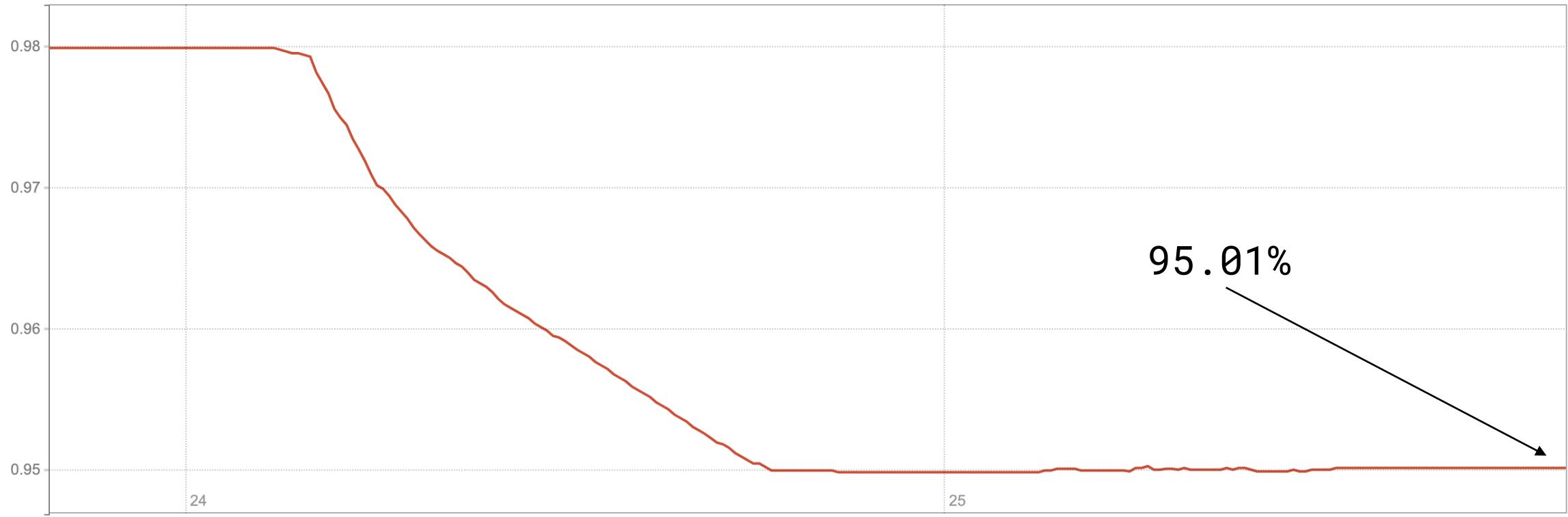
# Indicateurs



# Niveau de service (1/2)



# Niveau de service (2/2)





# Atelier

# TouchTunes

Rock on.