

MONITORIZACIÓN DE ESCENARIO



Grafana

EVOLUCIÓN DEL ESCENARIO

- Comenzamos con un CMS y una base de datos



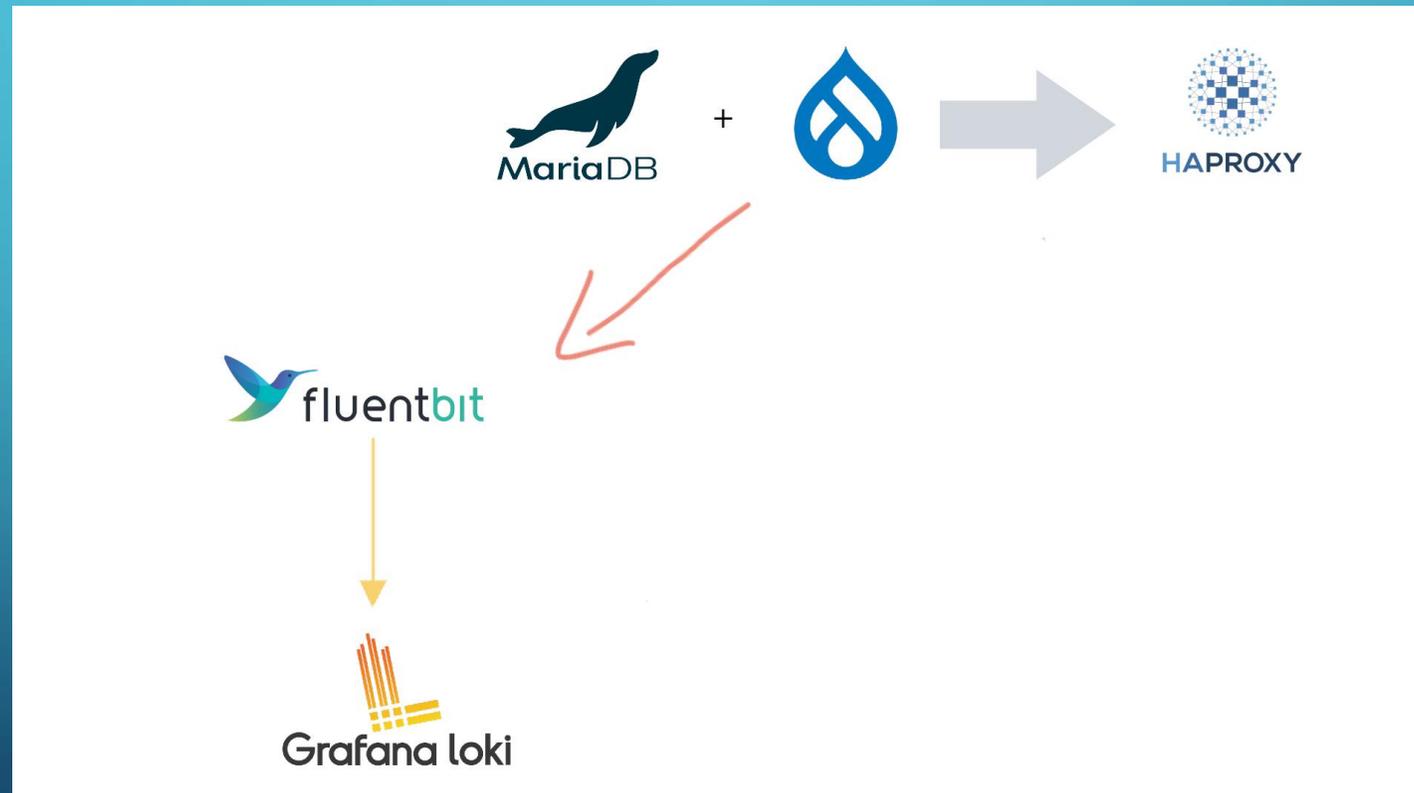
EVOLUCIÓN DEL ESCENARIO

- La primera mejora será añadir HAProxy, añadiendo un balanceo round-robin para asegurar alta disponibilidad



EVOLUCIÓN DEL ESCENARIO

- Una vez que tenemos alta disponibilidad, sería conveniente centralizar los logs de los contenedores, por lo que añadiremos Fluent-Bit y Loki



¿QUÉ ES FLUENT-BIT?

- Fluent-Bit es un recolector, procesador y reenviador de registros de código abierto, destacando en su bajo consumo y alto rendimiento.
- Se propone como solución a unificar la gran cantidad de registros almacenados.
- Nace en 2014 bajo el paraguas de Fluentd.

	FLUENTD	FLUENT-BIT
Alcance	Contenedores/ Servidores	Linux integrado/ Contenedores / Servidores
Lenguaje	Escrito en C y Ruby	Escrito en C
Rendimiento	Alto rendimiento	Alto rendimiento
Dependencias	Al estar construido con Ruby necesita más dependencias	No necesita dependencias
Complementos	+1000	~=70

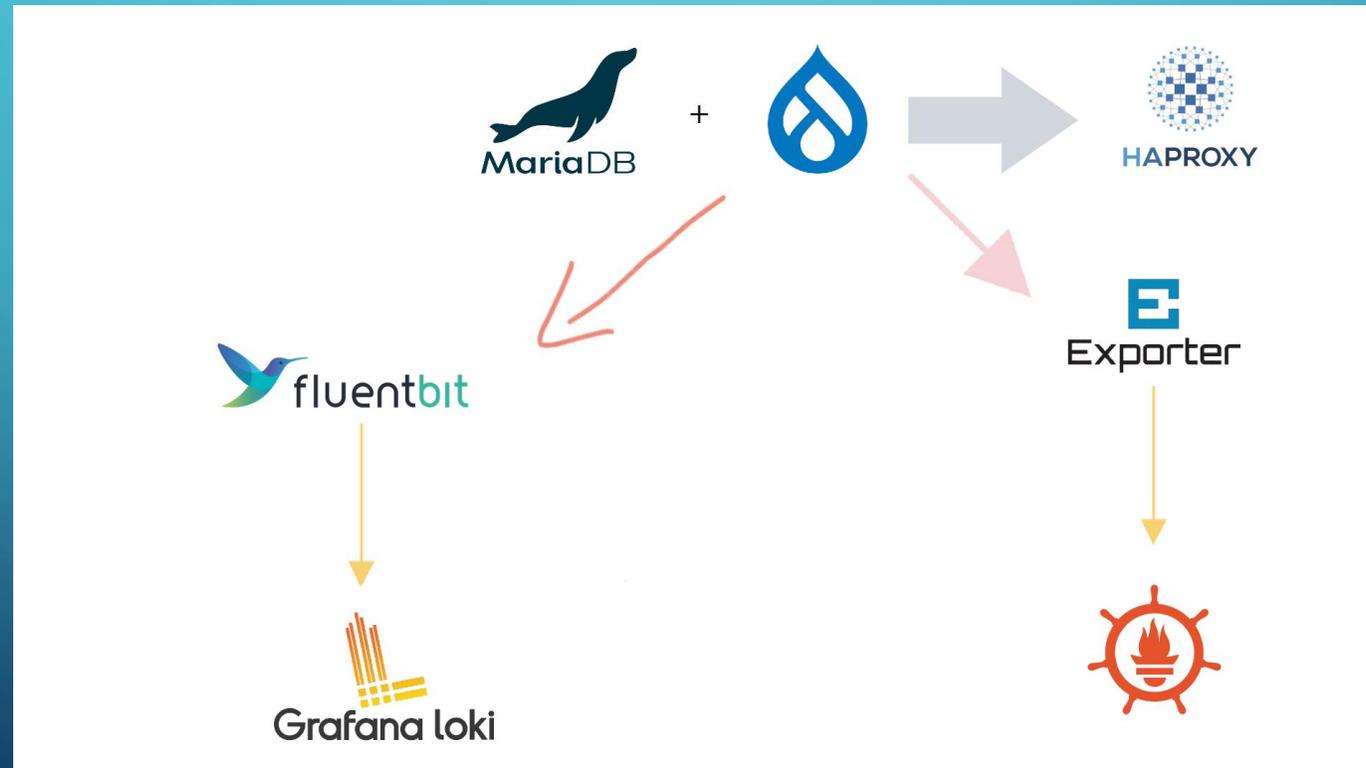
¿QUÉ ES LOKI?

- Es un sistema de registro de código abierto enfocado en la agregación y almacenamiento de registros de aplicaciones.
- Desarrollado por Grafana Labs, Loki se presenta como una solución moderna para el manejo de registros en entornos distribuidos.
- Lo usaremos como almacenamiento de registros, los que Loki nos proporcione.



EVOLUCIÓN DEL ESCENARIO

- Aún teniendo alta disponibilidad y los logs centralizados en loki, sería conveniente conocer las métricas de los hosts que tengamos en nuestro escenario, por lo que podríamos añadir Node Exporter + Prometheus



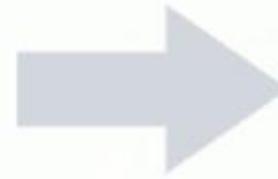
¿QUÉ ES NODE EXPORTER?

- Es una herramienta a cargo de la obtención de estadísticas de un nodo bajo el formato que implementen, por ejemplo JSON o XML.
- Fue desarrollado por la fundación Prometheus.
- Sus principales ventajas son la implementación y la integración

¿QUÉ ES PROMETHEUS?

- Es una plataforma de monitoreo y alerta diseñada para recopilar métricas de sistemas y servicios
- Permite almacenar y consultar datos de manera eficiente y flexible, y ofrece un poderoso lenguaje de consulta para analizar y visualizar métricas
- Sus ventajas incluyen una arquitectura escalable, capacidad de descubrimiento automático de servicios y una activa comunidad de desarrollo.

INTEGRACIÓN DE NODE EXPORTER Y PROMETHEUS



EVOLUCIÓN DEL ESCENARIO

- Hemos conseguido recolectar métricas y logs de máquinas, pero de forma distribuida, ya que ésta información se encuentra cada una en su correspondiente contenedor, por lo que añadiremos Grafana al escenario



¿QUÉ ES GRAFANA?

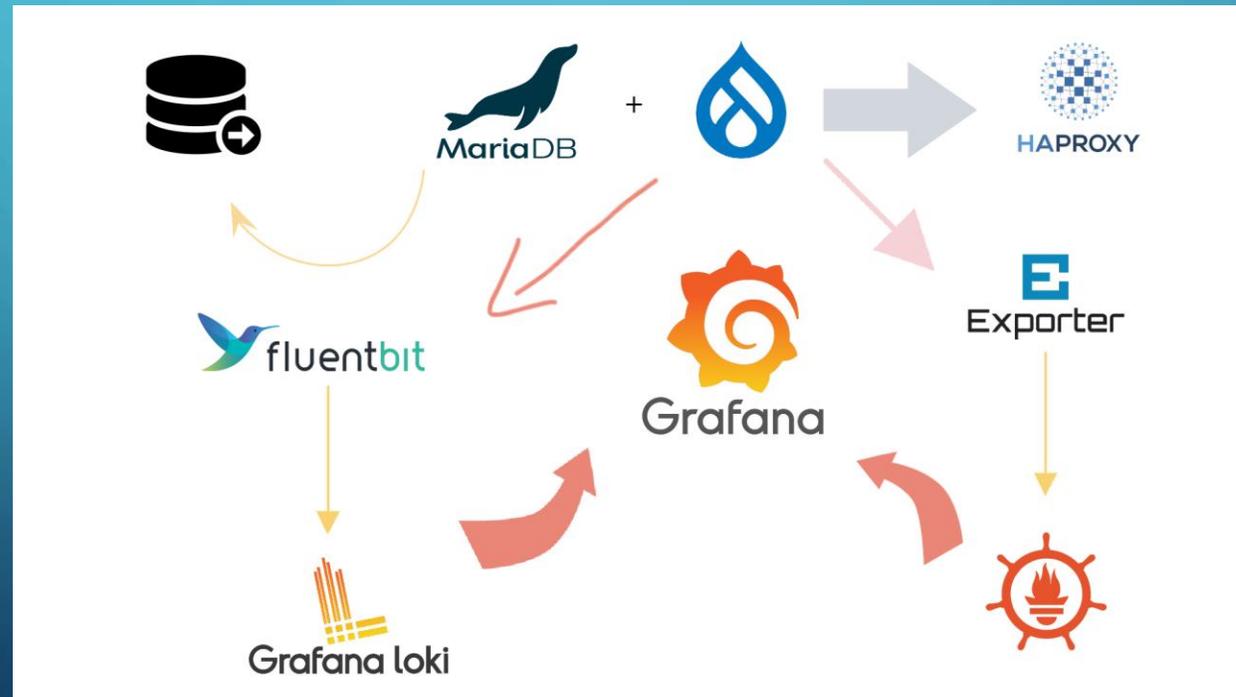
- Grafana es una plataforma de visualización de datos y paneles de control que se integra con otras fuentes de datos
- Permite crear visualizaciones gráficas interactivas y personalizadas a partir de las métricas recopiladas por Loki y/o Prometheus
- Sus ventajas incluyen una amplia variedad de paneles preconfigurados, una extensa biblioteca de complementos y la capacidad de crear alertas basadas en umbrales personalizados

INTEGRACIÓN DE GRAFANA



EVOLUCIÓN DEL ESCENARIO

- Siguiendo con la evolución del escenario, es cierto que la máquina Drupal se encuentra más controlada que al principio pero rápidamente podemos darnos cuenta que hemos descuidado el contenedor MariaDB, así que añadiremos el contenedor MySQL Exporter con el que visualizaremos información sobre la base de datos



¿QUÉ ES MYSQL EXPORTER?

- Es una herramienta que permite recopilar y exponer métricas de rendimiento de bases de datos MySQL
- Estas métricas se pueden utilizar para supervisar el rendimiento, detectar cuellos de botella y tomar decisiones informadas sobre la optimización de la base de datos
- MySQL Exporter es una solución de código abierto respaldada por la comunidad, lo que garantiza su constante desarrollo y mejora



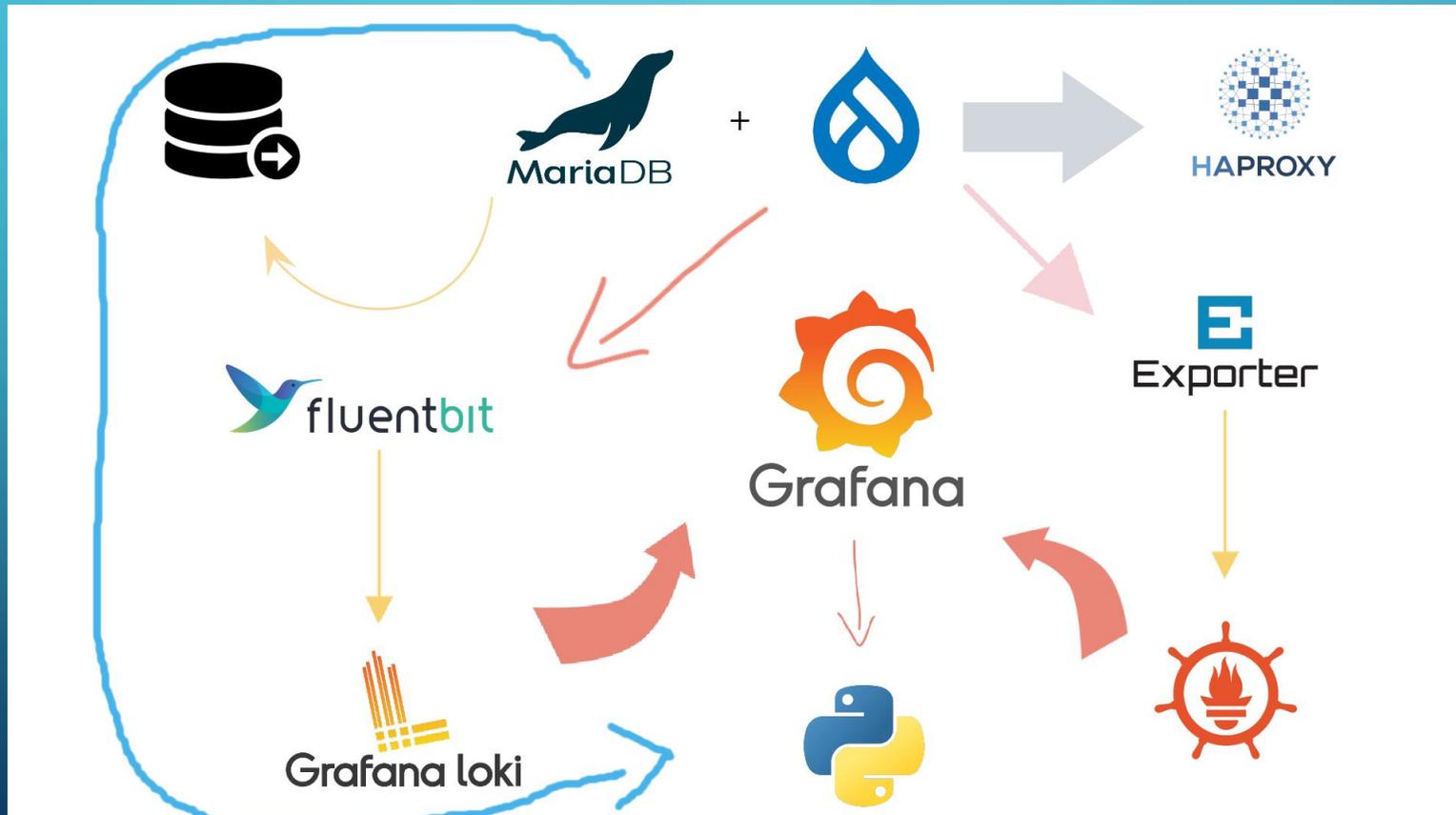
EVOLUCIÓN DEL ESCENARIO

Una vez obtenemos métricas de la base de datos, el escenario podría haber finalizado, en cambio, añadiremos un programa Python con el cual interactuaremos con Grafana a través de su API



EVOLUCIÓN DEL ESCENARIO

Por último, añadiremos al programa Python una última opción con la que testaremos la base de datos



PROGRAMA PYTHON

¿Por qué hacer un programa Python que conecte con Grafana?

- Investigando descubrí que la API de Grafana nos ofrece mucha versatilidad
- Aceleración de tareas rutinarias, como creación de usuarios o consultar información
- Automatización
- Aprovechamiento para otras tareas, como testeo de la BBDD

DEMOSTRACIONES

- HAProxy round-robin
- Recolección de logs
- Filtrado de logs
- Visualización de métricas y testeando RAM y CPU
- Monitorización de MariaDB
- Alertas al suceder determinado evento
- Programa Python con Grafana API