
Data Konferences

IoT, Kafka & k8s: Midiendo la
temperatura a lo grande

lot es Big Data?

IoT es Big Data?



IoT es Big Data?



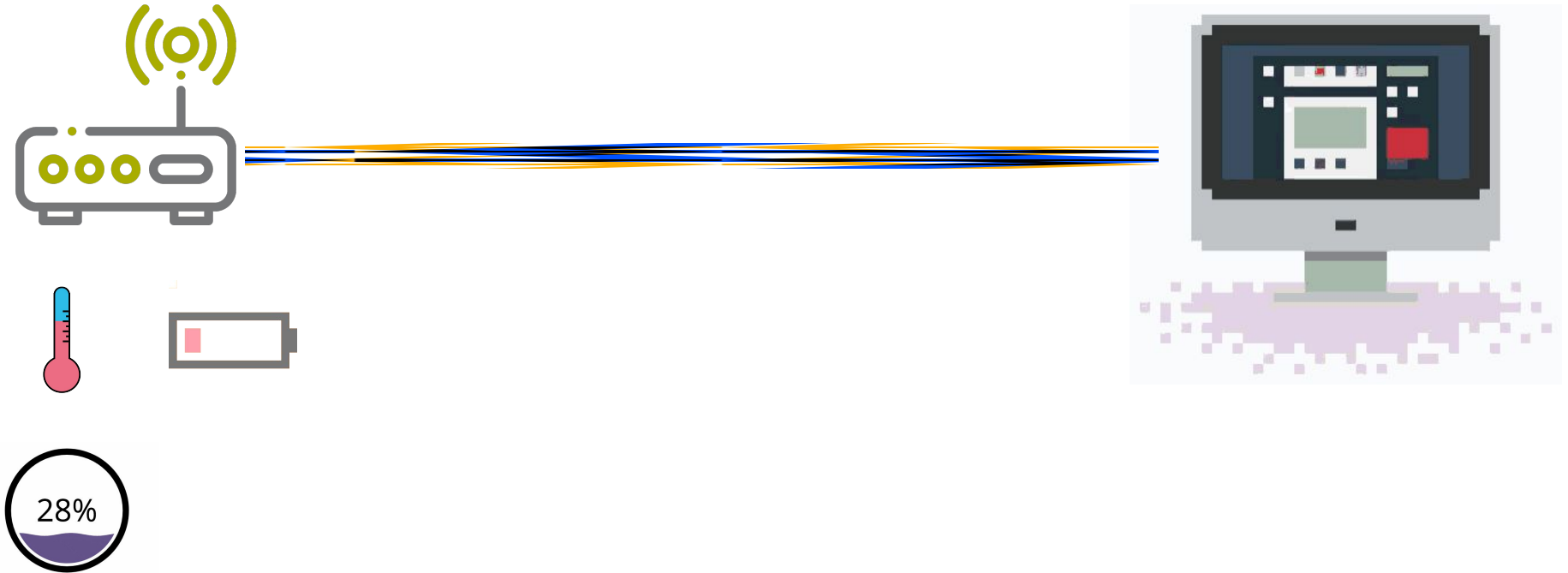
IoT es Big Data?



IoT es Big Data?



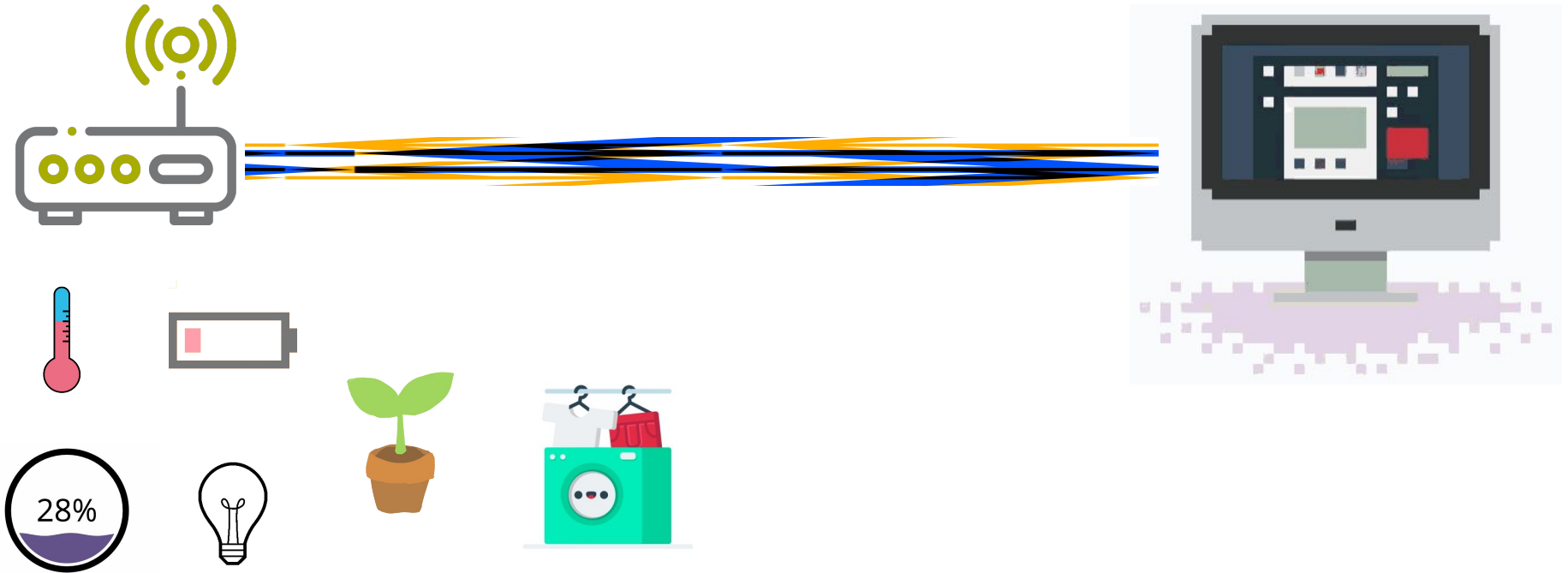
IoT es Big Data?



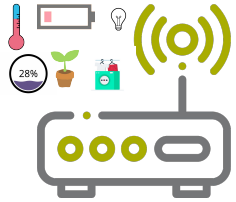
IoT es Big Data?



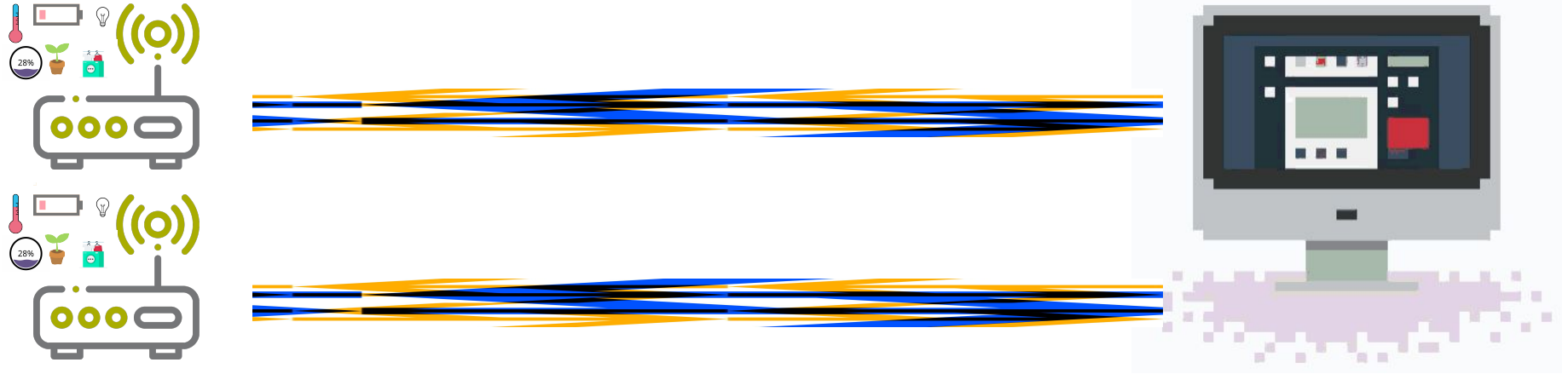
IoT es Big Data?



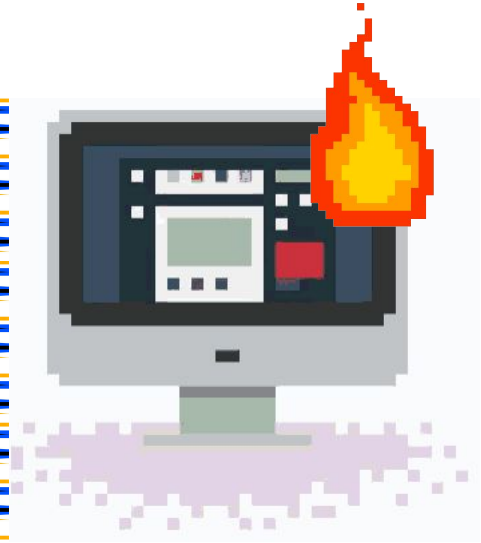
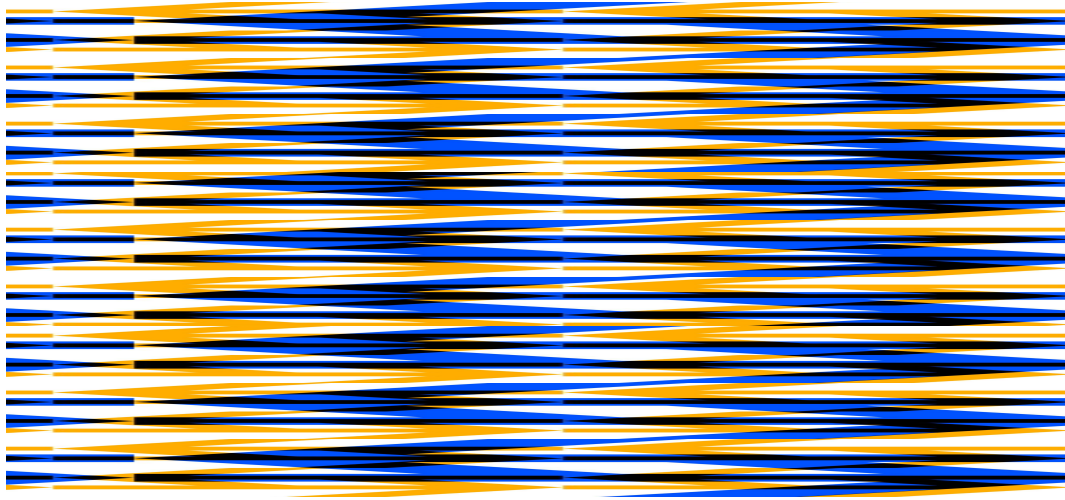
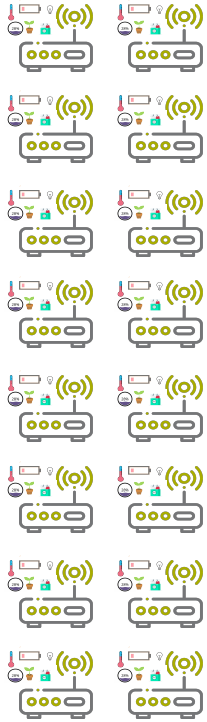
IoT es Big Data?



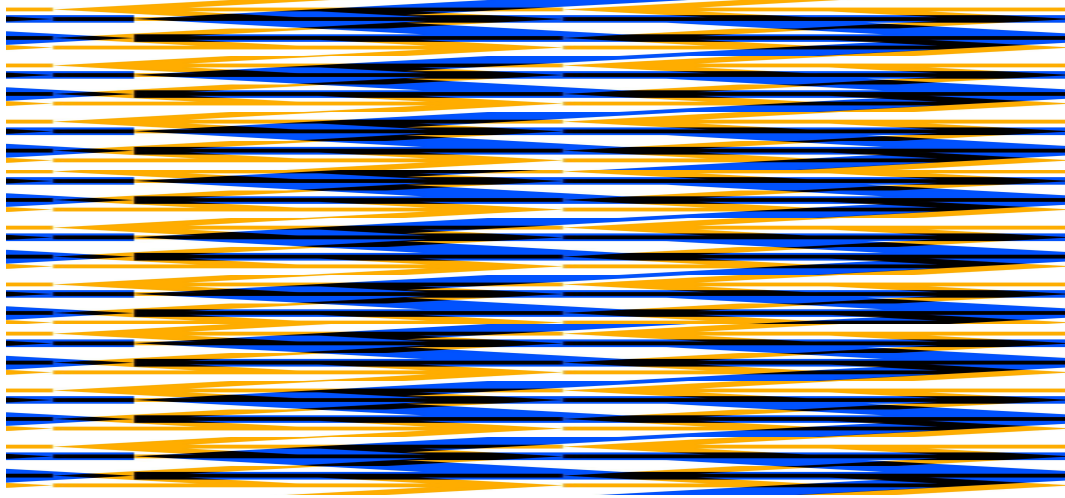
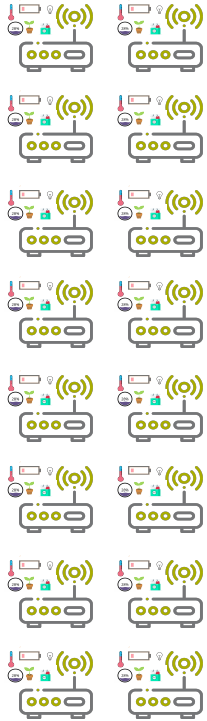
IoT es Big Data?



IoT es Big Data?



IoT es Big Data?



IoT es Big Data?

```
{  
  "temperature": 22,  
  "humidity": 50,  
  "sensor": "AAAA",  
  "light": "on",  
  "light_2": "on",  
  "light_3": "off",  
  "volume_control": 0.50,  
  "timestamp": 1234567890,  
  "metadata": {  
    "version": 1.0.0,  
    "model": "HW-MD202",  
    "location": "garden"  
  }  
}
```

321 bytes/mensaje 1 mensaje segundo

1 sensor		1.000.000 sensores (~2% población española)	
1 minuto	~ 18 KB	1 minuto	~ 16 GB
1 hora	~ 1 MB	1 hora	~ 976 GB
1 día	~ 24 MB	1 día	~ 22 TB
1 mes	~ 720 MB	1 mes	~ 686 TB
1 año	~ 8.5 GB	1 año	~ 8 PB

Tecnología

Tecnología: Stack



kubernetes



Google Cloud

Tecnología: Google Cloud

- Toda la funcionalidad de la cloud: instancias, red, almacenamiento, balanceadores...
- **Google Kubernetes Engine (GKE)**
 - Kubernetes cluster as a service.
 - Instancias VM con imagen de k8s.
 - Autoescalado automático de nodos.
 - Volúmenes en red.
 - Balanceadores de carga.
 - mucho más...!

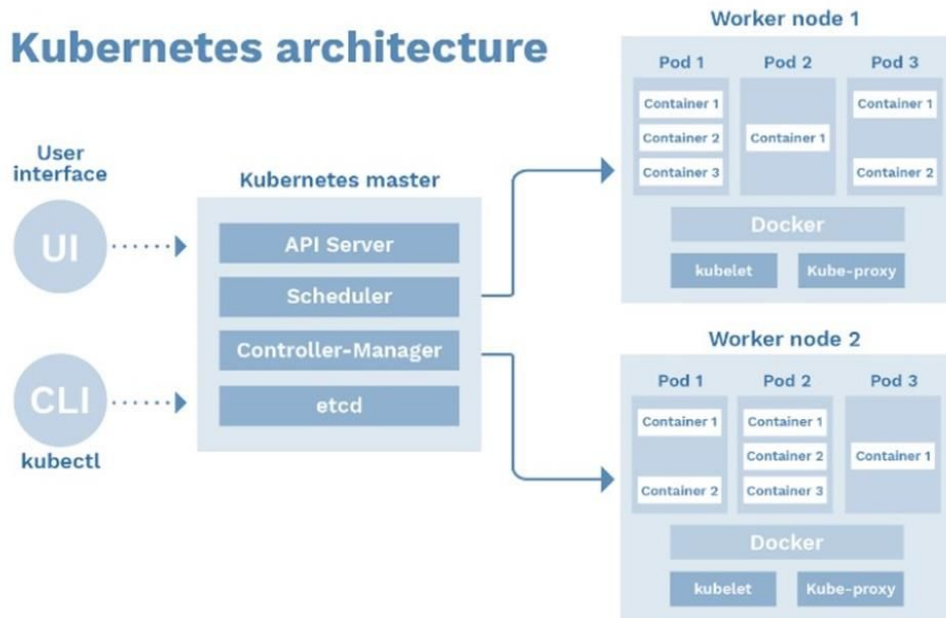


Tecnología: Kubernetes

Simplificado: Orquestador de contenedores distribuido.

- Contenedores con estado.
- Control de réplicas.
- Escalado de contenedores en base a utilización de recursos.
- Balanceadores a nivel de servicio.
- Tolerancia a fallo.
- mucho más...!

Kubernetes architecture



Tecnología: Kafka

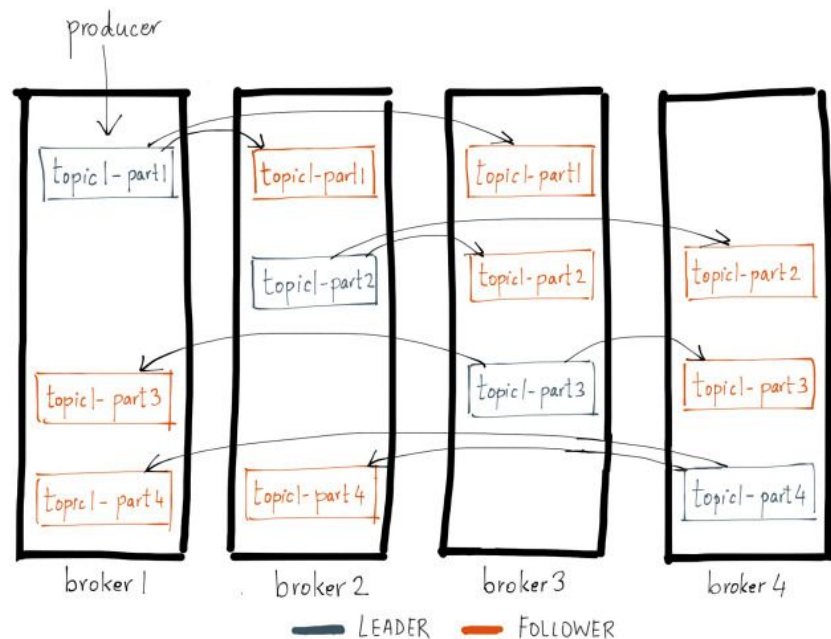
- Sistema de cola distribuido.
- Colas → Topics → Particiones (múltiples réplicas, 1 leader).
- Particiones distribuidas en brokers.
- Brokers forman un cluster.

Kafka Rest

- Servicio que proporciona una capa HTTP, para producir y consumir mensajes de Kafka.

Kafka Streams (iot-engine)

- Tecnología de procesamiento streaming.
- Proporciona: escalabilidad, stateful processing, tolerancia a fallos, windowing, etc



Implementación

Implementación

1. Google Load Balancer

- Puntos de entrada HTTP al sistema.
- Enviar datos de los sensores.
- Consultar métricas agregadas.
- Gestionar las reglas.
- Consumir alertas.

2. Kafka Rest:

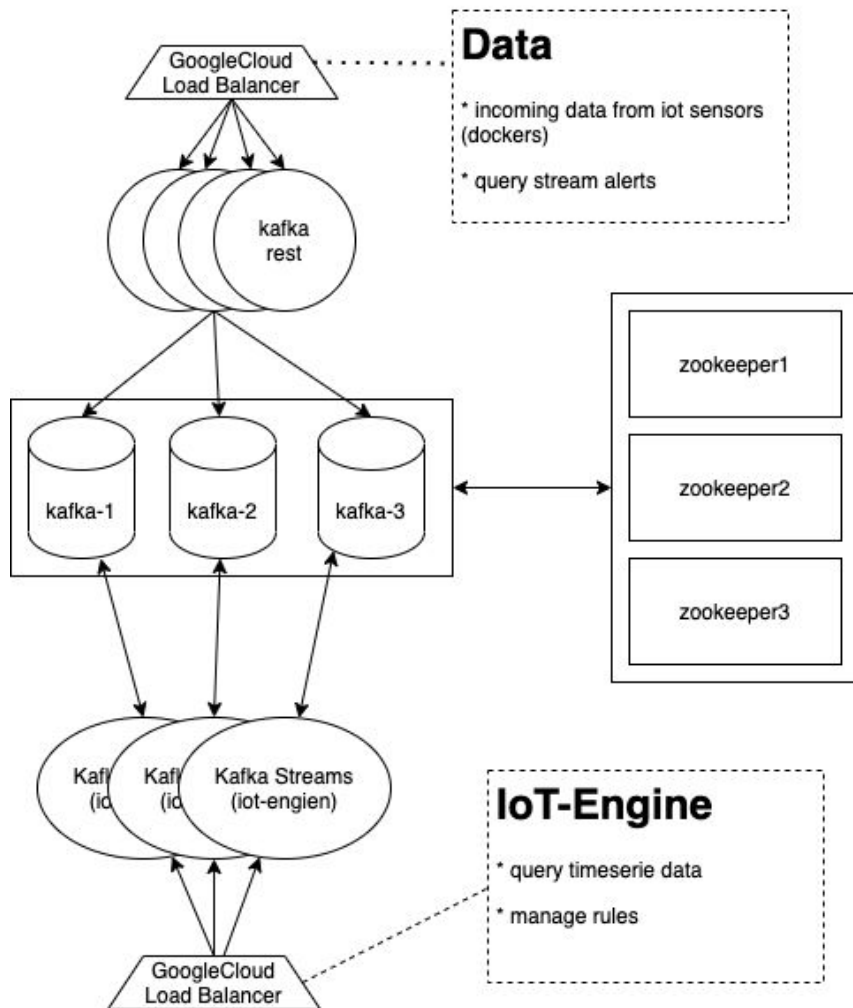
- Convertidor HTTP<->KAFKA.

3. Kafka:

- Almacenamiento del sistema.
- RealTime + Batch

4. IoT Engine:

- Aplicación Kafka Streams.
- Realiza procesamiento de mensajes.
- Agrega datos, verifica reglas, genera alertas.
- Expone métricas.



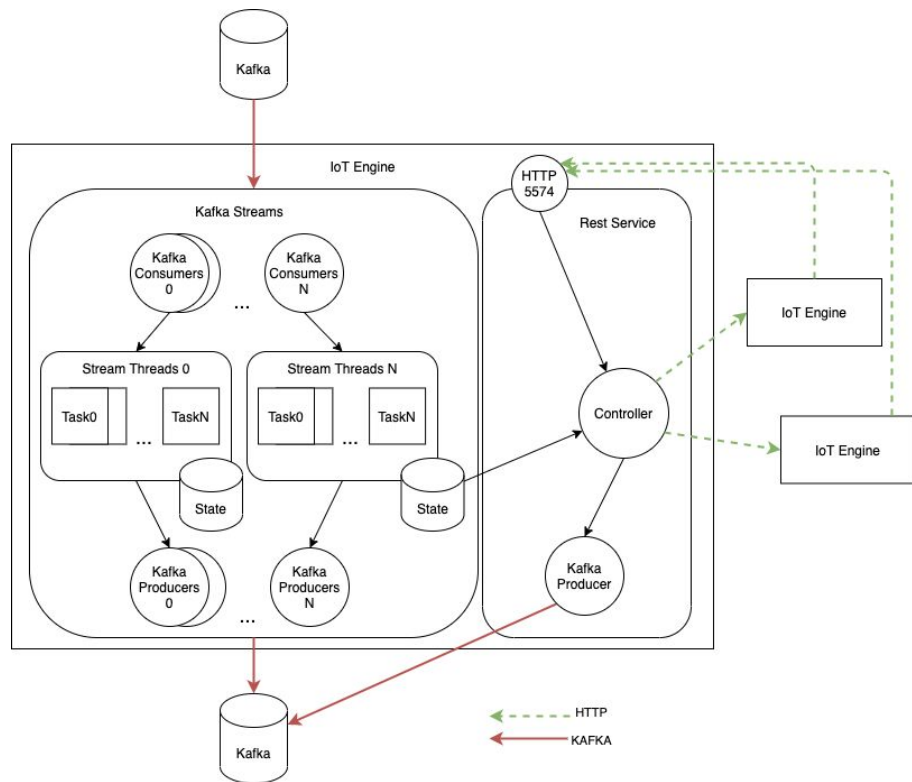
Implementación: IoT-Engine

Kafka Streams

- Consumir/deserializar datos JSON desde Kafka.
- Agrega las métricas por ID y minuto.
- Expone las métricas en los states.
- Consume las reglas desde Kafka.
- Valida las reglas con los datos.
- Genera alertas y datos agregados en Kafka.

Rest Service

- Proporciona API REST: Gestión de reglas, consultas de datos en timeseries, consulta metadatos cluster iot-engine.
- Consulta la información de los states de kafka streams.
- Redirige las peticiones a otros motores de iot-engine, si contienen la información solicitada.



Demo Time

https://github.com/andresgomezfrr/data_konferences_2021

Links

- Google Cloud Kubernetes : <https://cloud.google.com/kubernetes-engine>
- Weavescope : <https://www.weave.works/oss/scope/>
- Kafka : <https://kafka.apache.org/>
- Zookeeper : <https://zookeeper.apache.org/>
- Kafka-Rest : <https://github.com/confluentinc/kafka-rest>
- IoT-Engine : <https://github.com/andresgomezfrr/iot-engine>
- Data Simulator : <https://hub.docker.com/r/andresgomezfrr/data-simulator>

Who?

Andrés Gómez Ferrer

Big Data Infrastructure Developer at Telefónica



<https://github.com/andresgomezfrr>



<https://www.linkedin.com/in/andresgomezfrr/>



<https://twitter.com/andresgomezfrr>

