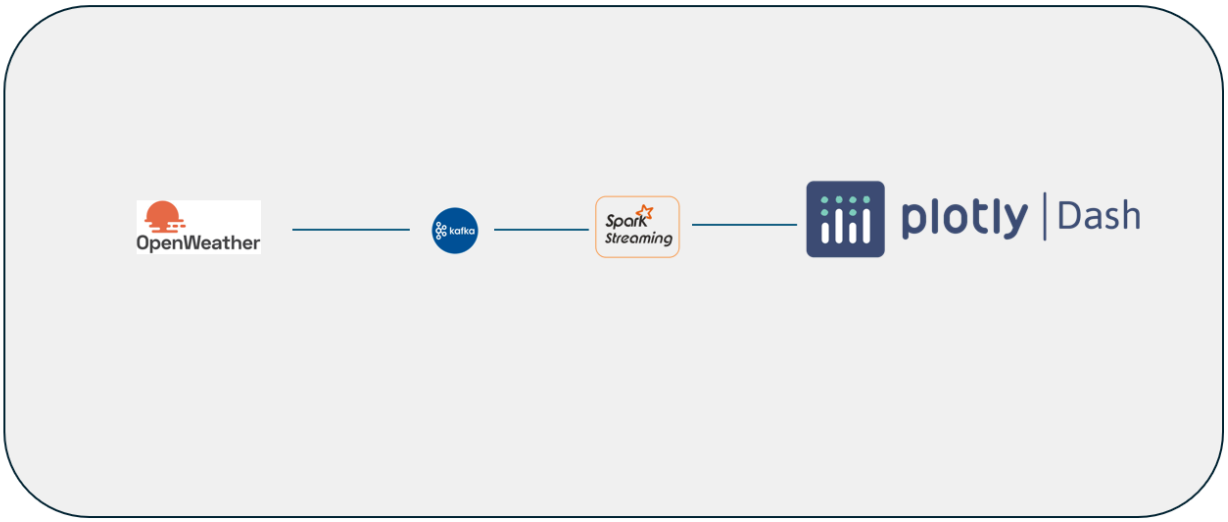
Isaac Laaouaj - Gardo en Ciencia de Datos e IA
UAX

Streaming de Datos Climáticos en Tiempo Real

En este proyecto se implementa un sistema para la ingesta, procesamiento y visualización de datos climáticos en tiempo real utilizando herramientas de tecnologías modernas como Apache Kafka, Apache Spark y Dash.

Flujo a alto nivel



---

1. Apache Kafka

Apache Kafka es una plataforma de mensajería distribuida diseñada para manejar flujos de datos en tiempo real con alta escalabilidad y tolerancia a fallos. Es ampliamente utilizada en sistemas donde se requiere transmitir grandes volúmenes de datos de forma continua.

- **Concepto de Topic:** Kafka organiza los datos en "topics", que son canales donde los productores publican mensajes y los consumidores los leen.
- **Producers y Consumers:** Los productores envían datos al sistema Kafka, mientras que los consumidores recuperan esos datos. Estos roles permiten una arquitectura desacoplada.
- **Logs distribuidos:** Kafka almacena los datos en forma de logs distribuidos replicados, lo que asegura la durabilidad y confiabilidad de los mensajes.
- **Particiones:** Cada topic está dividido en particiones, permitiendo la distribución de la carga entre varios nodos y logrando mayor rendimiento.
- **Casos de uso:** Kafka es ideal para sistemas de streaming, análisis de datos en tiempo real, monitoreo y procesamiento de eventos.

Para el proyecto, Kafka actúa como un intermediario para ingerir datos desde una API climática y ponerlos a disposición de las aplicaciones downstream, como Spark y Dash.

---

2. Apache Spark

Apache Spark es un motor de procesamiento de datos en clústeres que permite realizar cálculos rápidos y distribuidos. Es una herramienta clave para el procesamiento de datos en tiempo real y por lotes debido a su velocidad, escalabilidad y facilidad de uso.

- **Arquitectura Resilient Distributed Dataset (RDD):** Spark utiliza los RDDs, una estructura inmutable y distribuida que facilita las operaciones paralelas y tolerantes a fallos.
- **Spark Streaming:** Un componente que permite procesar datos en tiempo real provenientes de fuentes como Kafka, HDFS o sockets. Convierte los datos en "micro-batches" que luego procesa como si fueran lotes.
- **Integración con Kafka:** Spark puede consumir datos directamente desde topics de Kafka, procesarlos y producir resultados procesados en tiempo real o casi en tiempo real.
- **Transformaciones y Acciones:** Spark aplica transformaciones (map, filter) y acciones (count, collect) sobre los datos para generar los resultados deseados.
- **Ventajas:**
    - Alta velocidad: Procesa datos en memoria.
    - Flexibilidad: Compatible con varios lenguajes (Python, Scala, Java).
    - Integración: Compatible con sistemas como Hadoop y bases de datos.

En este proyecto, Spark se encarga de consumir los datos publicados en Kafka, procesarlos en tiempo real y preparar la información para su visualización.

---

3. Dash y Plotly

Dash es un framework de Python utilizado para construir dashboards interactivos. Se basa en Flask para la parte del servidor, Plotly para las visualizaciones y React.js para los componentes interactivos.

- **Componentes principales:**
    - `dcc.Graph`: Permite incluir visualizaciones avanzadas como gráficos de líneas, mapas de calor y más.
    - `html.Div`, `html.H1`, etc.: Elementos de HTML que estructuran la aplicación.
    - `dcc.Interval`: Facilita la actualización automática de los datos en el dashboard.

- **Callbacks:** Dash utiliza un modelo reactivo basado en callbacks para actualizar los componentes de la interfaz cada vez que los datos cambian.
- **Personalización:** Dash permite adaptar completamente el diseño y las visualizaciones, integrando estilos personalizados mediante CSS.
- **Plotly:** Este paquete es el motor gráfico de Dash y ofrece gráficos interactivos de alta calidad que mejoran la experiencia del usuario.

Para el proyecto Dash proporciona un dashboard interactivo que visualiza las temperaturas, precipitaciones y otros datos climáticos en tiempo real. Esto facilita el monitoreo y análisis dinámico de los datos procesados.

---

4. Integración y Flujo de Datos

El sistema integra las herramientas mencionadas en un flujo continuo:

1. **Kafka:** Recoge datos en tiempo real desde la API de OpenWeatherMap y los publica en un topic.
2. **Spark:** Consume los datos del topic, los procesa para extraer información relevante (por ejemplo, temperatura y humedad) y genera resultados procesados.
3. **Dash:** Muestra los datos procesados en un dashboard interactivo, actualizándose periódicamente para reflejar los datos en tiempo real.

---

5. Ventajas del Sistema

- **Escalabilidad:** Gracias a Kafka y Spark, el sistema puede manejar grandes volúmenes de datos y escalar horizontalmente.
- **Tiempo real:** Permite visualizar información climática actualizada al minuto, lo cual es útil para aplicaciones de monitoreo.
- **Interactividad:** El dashboard de Dash ofrece una experiencia de usuario intuitiva y personalizable.
- **Desacoplamiento:** Cada componente está diseñado para ser independiente, facilitando el mantenimiento y la extensión del sistema.

Conclusión

Este proyecto me ha dado las bases para construir sistemas de procesamiento de datos en tiempo real. La combinación de Apache Kafka, Apache Spark y Dash permite crear una solución robusta, escalable y flexible para monitorear datos climáticos en tiempo real. Esta arquitectura puede adaptarse a otros casos de uso, como análisis financiero, IoT y más.