นายคุณากร พฤกษากร 6610422020

Report (DADS6005)

รายงานอธิบายเกี่ยวกับการทำงานของแต่ละองค์ประกอบตาม requirement

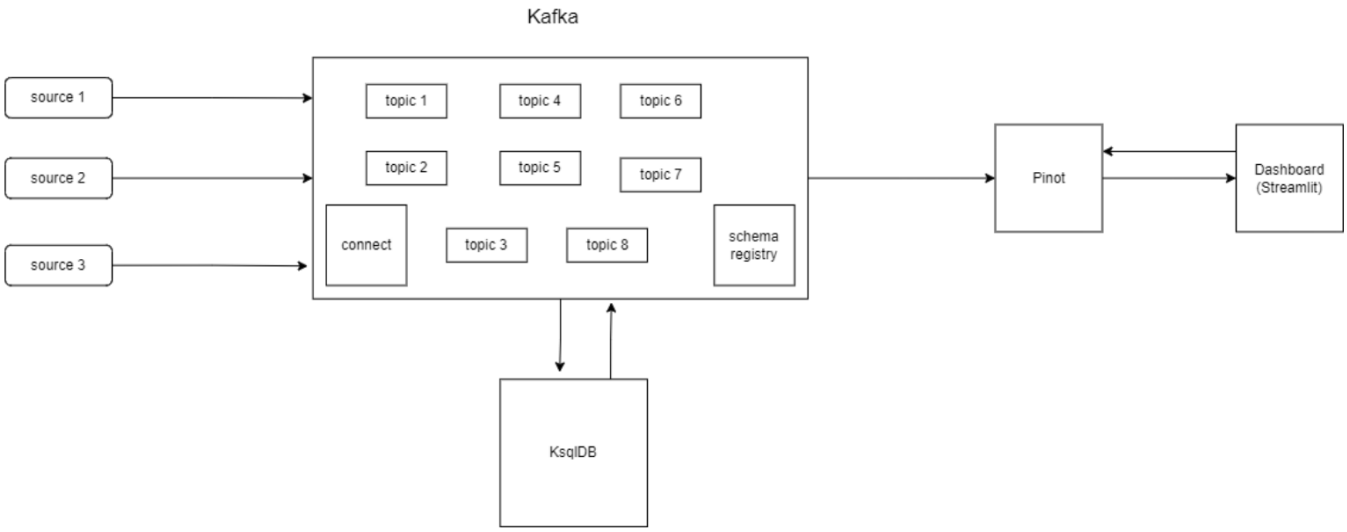

Figure 1 Data Streaming and Real-time Analytics System

1. Data Sources (แหล่งข้อมูล 3 แหล่ง)

a. What องค์ประกอบนี้คืออะไร

Data Sources คือแหล่งข้อมูลที่ใช้ในการส่งข้อมูลเข้าระบบ Kafka ซึ่งใน Midterm Exam นี้มีอยู่ 3 แหล่ง:

- **Source 1 (topic1)**: PageView ข้อมูลจำลองที่บันทึกการเข้าชมหน้าเว็บ
- **Source 2 (topic2)**: Users ข้อมูลเกี่ยวกับผู้ใช้ เช่น User ID, ภูมิภาค และเพศ
- **Source 3 (topic3)**: Relational database ที่เราออกแบบเองซึ่งผมได้ออกแบบให้มีข้อมูลเกี่ยวกับภูมิภาค เช่น ชื่อจังหวัด, จำนวนประชากร และขนาดพื้นที่

b. Why หน้าที่และความสำคัญขององค์ประกอบนี้ในระบบ

Data Sources เป็นข้อมูลที่ระบบสร้างขึ้นมา หรือ เราสร้างขึ้นมาเอง ซึ่งข้อมูลเหล่านี้จะถูกส่งต่อไปยัง Kafka แบบ real-time ในองค์ประกอบนี้ถือเป็นส่วนที่สำคัญมาก ถ้าไม่มีข้อมูลเข้ามาแบบ real-time เราก็ไม่จำเป็นต้องใช้ระบบนี้ แต่ถ้าข้อมูลเข้ามาแบบreal-time เราก็จะส่งไปยัง Kafka ตาม topic ต่างๆ เพื่อให้ส่วนอื่นๆ ในระบบ เช่น ksqlDB และ Apache Pinot นำไปใช้งานต่อ รวมถึงนำไปทำ Real-time Dashboard

c. How องค์ประกอบทำงานอย่างไร โดยให้อธิบายตาม flow ดังนี้

- **Input (data)**: ไม่มีข้อมูลเข้ามา แต่เป็นคำสั่งใช้ในการ Generate data
  - Source1(PageView): ข้อมูลนี้จะถูกสร้างโดยการสุ่ม ประกอบด้วย viewtime,userid,pageid
  - Source2(Users): ข้อมูลผู้ใช้จากการสุ่ม ประกอบด้วย registertime,userid,regionid,gender
  - Source3(design) : มีข้อมูล Region_id, ชื่อจังหวัด (Region_name), ข้อมูลจำนวนประชากรในจังหวัด (Population), และขนาดพื้นที่ (Area_size)
- **Process**:
  - ข้อมูลจาก Source 1 และ Source 2 จะถูกสร้างโดยใช้เครื่องมือ datagen ที่ช่วยให้ข้อมูลสร้างขึ้นโดยอัตโนมัติ และส่งเข้าสู่ Kafka อย่างต่อเนื่อง
  - Source 3 ใช้ script Python (source3_data.py) เพื่อสร้างข้อมูล และส่งเข้ามาใน Kafka topic_3
- **Output (results)**: ข้อมูลทั้งหมดจะถูกรวบรวมอยู่ใน Kafka topics โดย topic1 คือ PageView (stream datagen), topic2 คือ Users (stream datagen), และ topic3 คือ ข้อมูลที่เราสร้างขึ้นมาเอง ซึ่งจะถูกนำไปใช้ในการประมวลผลต่อไป

2. Kafka System (ระบบจัดการ Kafka)

a. What องค์ประกอบนี้คืออะไร

Kafka System เป็นระบบที่ทำหน้าที่รับ-ส่ง และจัดเก็บข้อมูลแบบ streaming ที่ใช้ Apache Kafka ทำงานในรูปแบบ real-time ช่วยให้การเชื่อมต่อข้อมูลจากหลายๆ แหล่งมาอยู่ในที่เดียวกัน ภายใน Kafka Cluster ที่ประกอบด้วยหลายๆ โหนด (nodes) จัดการผ่าน brokers โดยจะมี topic กำกับ คนที่ส่ง(Publisher)ก็จะส่งด้วย topic หนึ่งๆ คนที่มารับ(Subscriber)ก็ต้องรับด้วย topic นั้นๆ และมี partitions เพื่อเพิ่มความเร็วในการประมวลผล เพื่อให้การประมวลผลเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ

b. Why หน้าที่และความสำคัญขององค์ประกอบนี้ในระบบ

Kafka เป็นตัวกลางที่ทำให้ข้อมูลจาก Data Sources ไปถึงผู้ใช้ได้ง่ายและเร็วขึ้น เป็นเหมือนจุดรวบรวมและแจกจ่ายข้อมูลไปตามแต่ละ topic ถ้าไม่มี Kafka ก็อาจจะต้องส่งแบบ1ต่อ1 ซึ่งถ้าผู้ส่งหรือผู้รับมีหลายคน ก็จะทำ

ให้ระบบช้า หรือถ้าผู้รับหลายคนอยากมารับ(Subscribe) topic เดียวกันนี้อาจจะทำให้ต้องส่งหลายรอบ จึงมี Kafka มาเป็นตัวกลางในการกระจาย broadcast ข้อมูล พูดง่ายๆคือ เราส่งให้ Kafka อย่างเดียว ผู้รับก็มารับจาก Kafka โดยใช้ topic เดียวกัน ทำให้การไหลของข้อมูลมีความต่อเนื่อง รวดเร็ว และตรวจสอบความถูกต้องได้

c. How องค์ประกอบทำงานอย่างไร โดยให้อธิบายตาม flow ดังนี้

- **Input (data)**: ข้อมูลที่ถูกส่งจาก Data Sources (Publisher) เข้ามาที่ Kafka โดยใน Midterm Exam นี้ ประกอบไปด้วย 3 source ใช้ 3 topic ประกอบด้วย topic1 คือ PageView (stream datagen), topic2 คือ Users (stream datagen), และ topic3 คือ ข้อมูลที่เราสร้างขึ้นมาเอง
- **Process**:
    - Kafka จัดเก็บข้อมูลด้วยการแบ่งเป็น 5 partitions ทำให้สามารถกระจายการประมวลผลและทำให้เร็วขึ้น
    - ข้อมูลที่เข้ามาผ่าน 3 brokers เพื่อความเสถียร ถ้า broker ใดมีปัญหา อีก brokers ยังสามารถช่วยจัดการข้อมูลได้
    - Schema Registry ใช้ในการจัดการ schema ของข้อมูลทุกชุดในระบบ เพื่อให้แน่ใจว่ารูปแบบข้อมูลตรงกัน
    - Topic คือ Stream ของข้อความที่ถูกจัดกลุ่มเข้าด้วยกัน (เปรียบเทียบได้กับ Table ใน database) ใช้เป็นตัวเชื่อมระหว่างผู้ส่ง(Publisher) และผู้รับ(Subscriber) ว่าส่งมา topic นี้ ถ้าผู้รับอยากได้ข้อมูลชุดนี้ ต้อง subscribe มาด้วย topic เดียวกัน (ชื่อ Topic จะต้องไม่ซ้ำกัน)
- **Output (results)**: ข้อมูลในรูปแบบต่างๆ เช่น JSON ที่ตรงตาม topic ที่ Subscriber ต้องการ เพราะว่า ถ้าจะเอาข้อมูลออกจาก Kafka ต้องมา Subscribe ด้วย topic นั้นๆ

3. ksqlDB Operations (การจัดการข้อมูลใน ksqlDB)

a. What องค์ประกอบนี้คืออะไร

ksqlDB เป็นแพลตฟอร์มที่ทำให้เราสามารถใช้ SQL query เพื่อจัดการข้อมูลใน Kafka เท่าที่ผมทราบคือ ksqlDB เกิดขึ้นเพราะ Kafka streams ใช้กับ Java ซึ่งเขียนยาก จึงทำ Kafka streams มาในรูปแบบของ database สามารถใช้ SQL query ได้เลย ทำให้เขียนโค้ดได้ง่าย รวดเร็ว ตัวอย่างเช่น การ transform ข้อมูล, การรวมข้อมูล

(aggregate), การ Join และการสร้าง window ที่ทำให้สามารถดูการเปลี่ยนแปลงในช่วงเวลาที่ต้องการ โดย ksqlDB ทำให้การเขียน SQL แล้วสามารถ operate ได้ เหมือน Java บน Kafka Streams เลย

b. Why หน้าที่และความสำคัญขององค์ประกอบนี้ในระบบ

ksqlDB ช่วยให้เราสามารถจัดการข้อมูลแบบ real-time (Kafka Streams) ได้ง่ายๆ โดยใช้ SQL query ได้เลย เราสามารถ Process/Transform data แบบ real-time/streams จริงๆได้ภายใน Kafka ก่อนออกจาก Kafka ด้วยซ้ำ โดยทำผ่านksqlDB ทำให้สามารถ Clean/Transform/Join data หรืออื่นๆ ก่อนออกจาก Kafka ตัวอย่างเช่น การทำ aggregate และการสร้าง window เพื่อตรวจสอบพฤติกรรมของผู้ใช้ในช่วงเวลาต่างๆ

c. How องค์ประกอบทำงานอย่างไร โดยให้อธิบายตาม flow ดังนี้

- **Input (data)**: ข้อมูลจาก Kafka topics เช่น topic1, topic2, และ topic3
- **Process**:
  - **Transform (topic4)**: ใช้ SQL เปลี่ยนแปลงรูปแบบข้อมูล เช่น เปลี่ยน timestamp ให้เป็นวันที่ที่อ่านง่ายขึ้น ผลลัพธ์เก็บใน Stream users_formatted
  - **Aggregate (topic5)**: ทำการ join ข้อมูลจากหลาย topic เช่น รวมข้อมูลการเข้าชมจาก topic1 กับข้อมูลผู้ใช้จาก topic2 และข้อมูลจังหวัด,ประชากรจาก topic3
  - **Windows (topic6 ถึง topic8)**: ใช้ windows ในการเก็บข้อมูลที่แบ่งตามเวลา เช่น
    - **Tumbling Window** (topic6): เก็บจำนวน page views ที่เกิดในทุก 1 นาที
    - **Hopping Window** (topic7): นับ page views ทุกๆ 5 วินาที พร้อมกับแสดงการเปลี่ยนแปลง
    - **Session Window** (topic8): เก็บข้อมูล session ที่บอกถึงระยะเวลาเข้าชมในแต่ละครั้งของผู้ใช้แต่ละคน
- **Output (results)**: ข้อมูลที่ถูกจัดการจะถูกบันทึกใน Kafka topics ใหม่ (topic4 ถึง topic8) สำหรับใช้ในการ query ต่อไป และ เพื่อรอ Subscriber มา subscribe ด้วย topic เหล่านี้

4. Apache Pinot (ระบบเก็บและ query ข้อมูล)

a. What องค์ประกอบนี้คืออะไร

Apache Pinot คือระบบฐานข้อมูลที่ช่วยให้สามารถ query ข้อมูลขนาดใหญ่ได้อย่างรวดเร็ว ถูกออกแบบมาเพื่อรองรับการวิเคราะห์ข้อมูลแบบ real-time โดยเฉพาะ เหมาะสำหรับการจัดเก็บและ query ข้อมูลจำนวนมากได้อย่างรวดเร็ว โดยข้อมูลเหล่านี้มาจากระบบ data streaming (Kafka) ที่ไหลเข้ามาเรื่อย ๆ ในปริมาณมาก และมีการเปลี่ยนแปลงตลอดเวลา โดยตัว Apache Pinot สามารถเก็บข้อมูลจาก Kafka และแสดงผลแบบ real-time analytics ได้อย่างรวดเร็ว

b. Why หน้าที่และความสำคัญขององค์ประกอบนี้ในระบบ

Apache Pinot ช่วยให้การ query ข้อมูลจาก Kafka เป็นไปอย่างรวดเร็วและมีประสิทธิภาพ สามารถใช้ในการวิเคราะห์ข้อมูล สามารถเก็บ real-time data ได้ ซึ่งมาจาก Kafka และเราสามารถเขียน query ได้อย่างง่าย (ผมเคยใช้ Database ที่คล้ายๆกัน คือ InfluxDB ซึ่งเก็บข้อมูลง่าย แต่ queryยากมาก พอมาใช้ตัวนี้ถึงแม้จะเก็บยาก Schemaต้องตรงเป๊ะๆ แต่เวลา query นั้นง่ายมาก) รวมทั้งสามารถ query ข้อมูล เพื่อนำไปแสดงผลบน dashboard แบบ real-time เช่น Streamlit , Grafana

c. How องค์ประกอบทำงานอย่างไร โดยให้อธิบายตาม flow ดังนี้

- **Input (data)**: ข้อมูลที่ผ่านการประมวลผลใน ksqlDB แล้ว (เช่น ข้อมูลใน topic5, topic6 และ topic8)
- **Process**:
  - ข้อมูลจาก Kafka topics ถูกดึงเข้ามาเก็บในตาราง (tables) ที่มี schema ตามที่กำหนด เช่น ตาราง Consolidate_REALTIME สำหรับข้อมูลที่ถูก aggregate จาก topic5
  - ข้อมูลใน Apache Pinot สามารถ query ด้วยคำสั่ง SQL ได้ ทำให้สามารถดึงข้อมูลสรุป เช่น จำนวนการเข้าชมเพจ หรือระยะเวลา session ของผู้ใช้แต่ละคนออกมาแสดงผลได้โดยง่าย
- **Output (results)**: ข้อมูลที่เราต้องการจะแสดงผลบน Dashboard ซึ่งได้จากการ query บน Apache Pinot ซึ่ง Dashboard ก็จะมาดึงข้อมูลจาก Apache Pinot ผ่านคำสั่ง query ตัว Pinot ก็จะส่งข้อมูลไปให้ real-time Dashboard แสดงผล ใน Midterm Exam นี้ใช้ Streamlit

5. Dashboard (การแสดงผลด้วย Streamlit)

a. What องค์ประกอบนี้คืออะไร

Dashboard เป็นส่วนที่ใช้แสดงข้อมูลในรูปแบบ visualizations ที่เราสร้างขึ้นด้วย Streamlit เพื่อให้ผู้ใช้งานสามารถมองเห็นข้อมูลและการวิเคราะห์ได้แบบ real-time เช่น กราฟ หรือ แผนภูมิ ที่แสดงข้อมูลพฤติกรรมการเข้าชมเพจของ users

b. Why หน้าที่และความสำคัญขององค์ประกอบนี้ในระบบ

Dashboard ช่วยให้ข้อมูลทั้งหมดที่เราประมวลผลมานั้นเข้าใจง่ายและสามารถนำไปใช้ประโยชน์ได้ทันที เช่น เห็นการเปลี่ยนแปลงของการเข้าชมเพจในช่วงเวลาต่างๆ แล้วนำไปตัดสินใจอะไรบางอย่าง, การดูข้อมูล session length ของผู้ใช้ หรือ ข้อมูลการเข้าชมแบ่งตามจังหวัด

c. How องค์ประกอบทำงานอย่างไร โดยให้อธิบายตาม flow ดังนี้

- **Input (data)**: ข้อมูลที่ query มาจาก Apache Pinot ซึ่งอาจรวมถึงข้อมูลที่อยู่ใน topic5, topic6, และ topic8
- **Process**:
  - Streamlit ช่วยให้เราสร้าง dashboard ที่เชื่อมต่อกับ Apache Pinot ได้อย่างง่าย โดยการ query ข้อมูลจาก Apache Pinot และนำไปสร้างเป็นกราฟหรือแผนภูมิต่างๆ ตามที่ต้องการ
  - การทำงานใน Streamlit จะสร้าง dashboard ที่มีหลาย panel ซึ่งสามารถแสดงข้อมูลได้แบบ interactive เช่น ผู้ใช้สามารถเลือกดูข้อมูลเฉพาะ session ที่สนใจหรือดูข้อมูลการเข้าชมในช่วงเวลาที่เลือก และ สามารถ filter เลือกจังหวัด, เพศ ได้ด้วย
- **Output (results)**: ผลลัพธ์คือการแสดงผลแบบ real-time ใน dashboard :
  - กราฟแสดงจำนวนการเข้าชมเพจต่อจังหวัด/ภูมิภาค
  - เวลาเฉลี่ยในการเข้าชมเพจแยกตามจังหวัด/ภูมิภาค
  - กราฟแสดงค่าเฉลี่ย session length ของผู้ใช้
  - กราฟเส้นแสดง Total Page Visits 5 นาทีหลังสุด เพื่อดู trend ได้

★ Dashboard (Streamlit)

http://ec2-47-129-89-174.ap-southeast-1.compute.amazonaws.com:8501

★ Confluent

http://ec2-47-129-89-174.ap-southeast-1.compute.amazonaws.com:9021

★ Apache Pinot

http://ec2-47-129-89-174.ap-southeast-1.compute.amazonaws.com:9000