

Kullanılan Teknolojiler

Projeyi geliştirirken aşağıdaki teknolojilerden yararlandım:

- Apache Kafka (v3.5.0)
- Java (v17)
 - kafka-clients (v3.5.1)
 - jackson (v2.15.2)

Projenin derlenmesi ve çalıştırılması

- Proje localhost'a çalışacak şekilde ayarlanmıştır.

Eğer kafka makinaları, localhost üzerinde çalışmayacaksa, Common projesindeki `KafkaConstant` sınıfını güncelleyebilirsiniz.

Apache Kafka kurulumları

Apache Kafka sürümünü belirlediğiniz dizine çıkardıktan (Örneğin: `/home/my-desktop/kafka_2.13-3.5.0/`):

- Zookeeper servisini başlatın:

```
[@desktop]$ pwd
/home/my-desktop/kafka_2.13-3.5.0/

./bin/zookeeper-server-start.sh config/zookeeper.properties
# veya
sh bin/zookeeper-server-start.sh config/zookeeper.properties
```

- Kafka broker servisini başlatın:

```
./bin/kafka-server-start.sh config/server.properties

# veya
sh bin/kafka-server-start.sh config/server.properties
```

- Topic'lerin yaratılması. (Üç farklı topic yaratmanız gerekiyor):

```
# topic-1
./bin/kafka-topics.sh --create --topic new-item-event --bootstrap-server
localhost:9092

# topic-2
./bin/kafka-topics.sh --create --topic radar-event --bootstrap-server
localhost:9092
```

```
# topic-3
./bin/kafka-topics.sh --create --topic send-target-to-mainframe-event --
bootstrap-server localhost:9092
```

Java projesini kafka ve zookeeper ayarları yapılmadan da çalışabilir. Fakat bu ayarları yapmadan çalıştırmak bir işe yaramayacaktır.

Java projesinin çalıştırılması

Java 17 ile yazılmıştır.

Toplamda 3 ayrı proje bulunmaktadır(Sensor, MainFrame ve SimulateWorld). Fakat ayrı ayrı maven projeleri oluşturup kodları kopyalamaktan ziyade multi-module proje oluşturdum.

Her bir projenin main sınıfını çalıştırabilirsiniz. **Common** projesi ortak sınıfları barındırmaktadır.

Sıra gözetmeden aşağıda belirtilen main sınıflarını çalıştırın:

- SimulateWorld projesinin içinde: **WorldApp.java** sınıfını
- Sensor projesinin içinde: **SensorApp.java** sınıfını
- Main projesinin içinde: **MainFrameApp.java** sınıfını çalıştırın

Veya ana proje dizinine gidip (**parent-hvl-project**), **mvn clean install -DskipTests** komutu ile bütün projeler için jar dosyalarını oluşturup çalıştırabilirsiniz.

Hedef bilgisinin gönderilmesi

Apache Kafka ve Java programlarını çalıştırdıktan sonra **new-item-event** topic'i kullanarak hedef bilgilerini gönderebilirsiniz. Örnek bir veri:

```
./bin/kafka-console-producer.sh --topic new-item-event --bootstrap-server
localhost:9092
>{"itemName":"test","itemType":"TARGET","xCoordinate":-1,"yCoordinate":5}
```

Göndereceğiniz verilerin örnekteki gibi olması gerekiyor, aksi halde geçeriz veri girişi yapıldığı varsayılacaktır

Probleme yaklaşım

Toplamda 3 farklı Java uygulaması geliştirdim.

- Her uygulama kendisine verilen kafka topic'ni dinliyor.
- Topic'den gelen veriye göre başka bir topic'e mesaj gönderiyor veya işlemi yapıp sonucu konsol üzerinden basıyor.

Problem için tasarladığım akış şu şekildedir:

- Dünyayı temsil etmek amaçlı oluşturduğum **SimulateWorld** projesi dünyaya gelen bütün istekleri dinlemektedir. Bir istek geldiğinde **KafkaTopicName.RADAR_TOPIC** topic'e istek göndermekte yeni

bir cismin geldiğini belirtmektedir.

- Sensörleri temsil etmek için oluşturduğum **Sensor** projesinde öncesinde iki tane sensörün yerleştirilmiş olduğunu varsaydım.(Bu varsayımı ilgili projenin main metodunda gerçekleştirdim)
 - Sensor projesi `KafkaTopicName.RADAR_TOPIC` isimli topic'leri dinliyor. Bir aksiyon geldiğinde gelen aksiyona göre kerteriz bilgisini hesaplıyor ve `KafkaTopic.SEND_TARGET_TO_MAINFRAME` topic'ine kerteriz bilgisini gönderiyor.
- Merkezi birimi temsil etmek amacıyla oluşturduğum **MainFrame** projesi, `KafkaTopic.SEND_TARGET_TO_MAINFRAME` gelen istekleri dinliyor gelen istek Sensor üzerinden geliyorsa verilen kerteriz bilgisine göre hedefin tespit etmeye çalışıyor.

Kerteriz bilgisine göre hedef tespiti

Burası en çok zorlandığım bölüm idi. Verilen iki noktadan üçüncü noktayı bulmak için linkte belirttiğim <https://everything2.com/title/Triangulate> yazıdan yararlandım(**Angles to Two Points Known**). Yazıdaki matematiksel işlemlerin yardımı için ChatGPT'den yararlandım.

Fakat bu çözüm sadece iki noktadan gelen veriler için üçüncü nokta bulunmak istendiğinde çalışmaktadır. Üç sensor üzerinden hedef belirlemesi yapmamaktadır