



# **PROJECT 1**

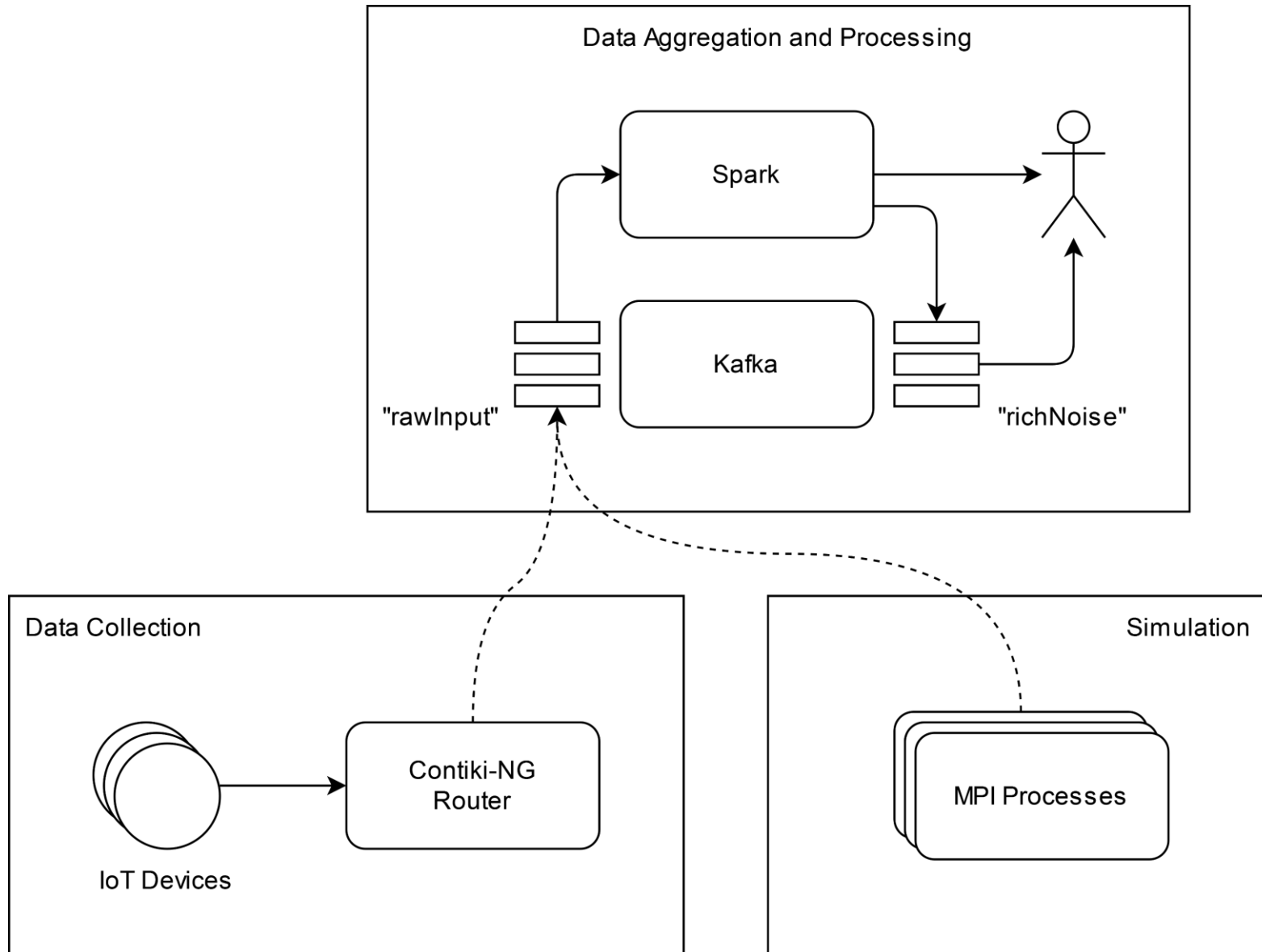
Simulation and Analysis of Noise Level

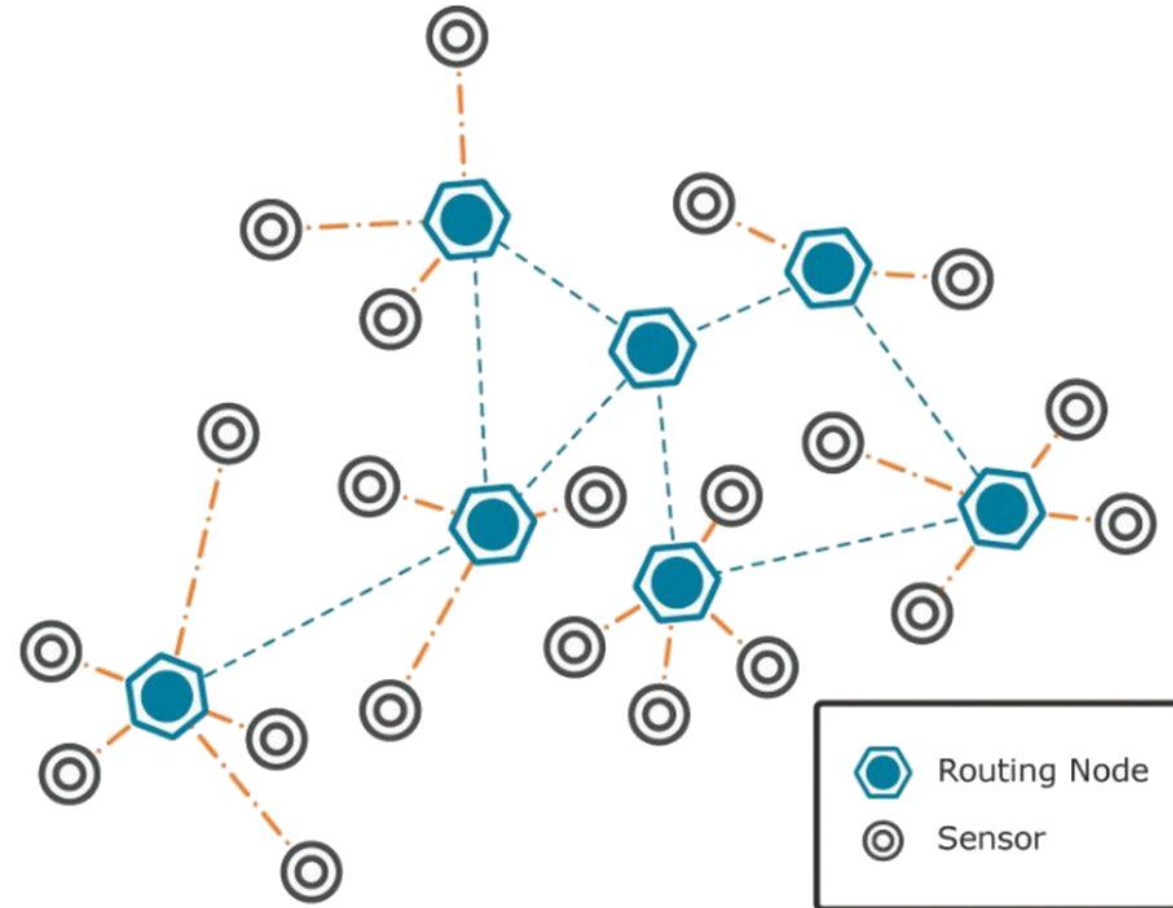


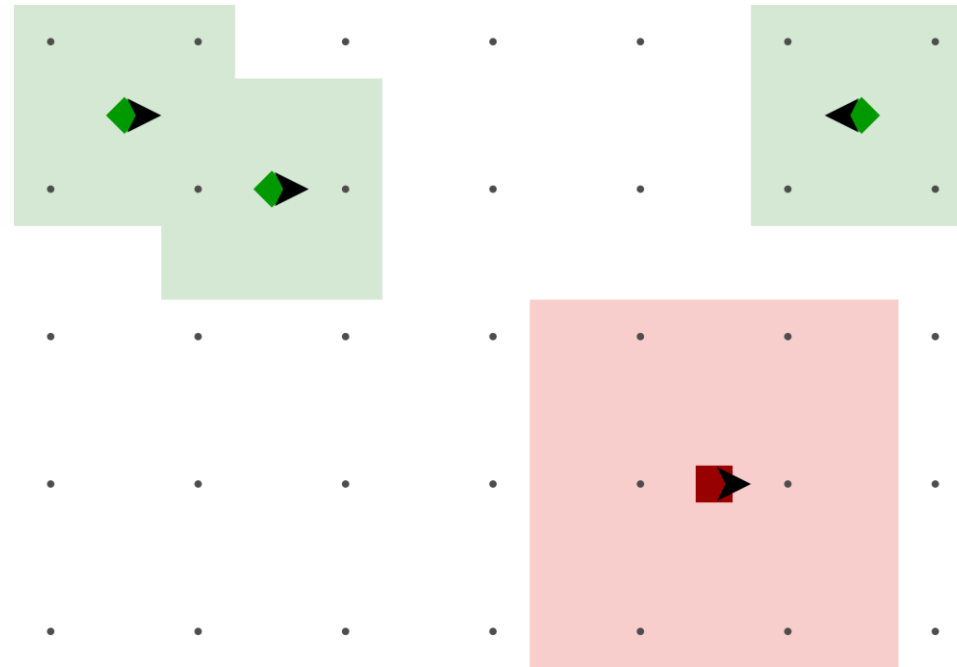
## GOALS

- **Collect** noise data
- **Simulate** when not possible
- **Clean** input data
- **Analyze** and compute metrics







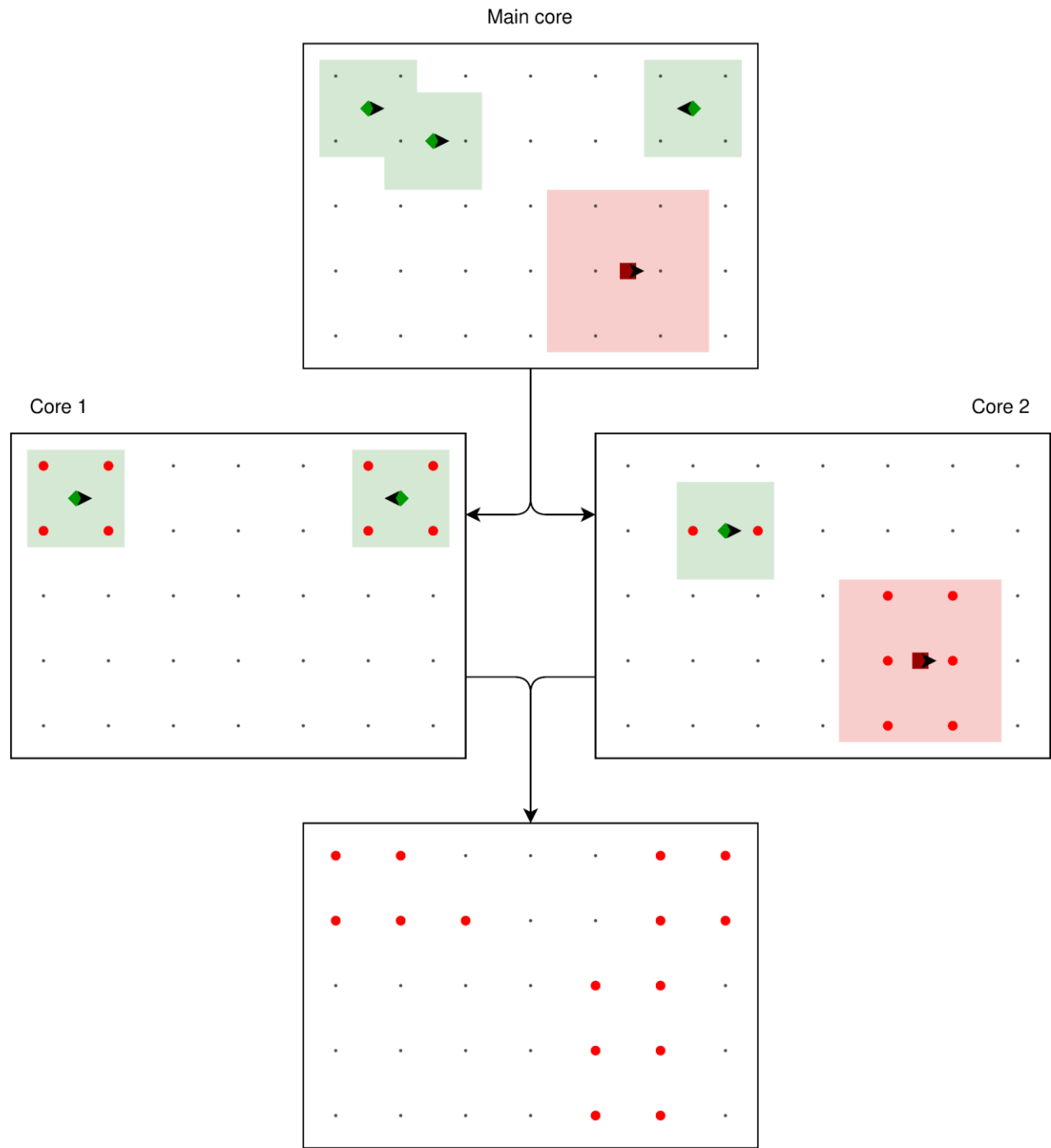


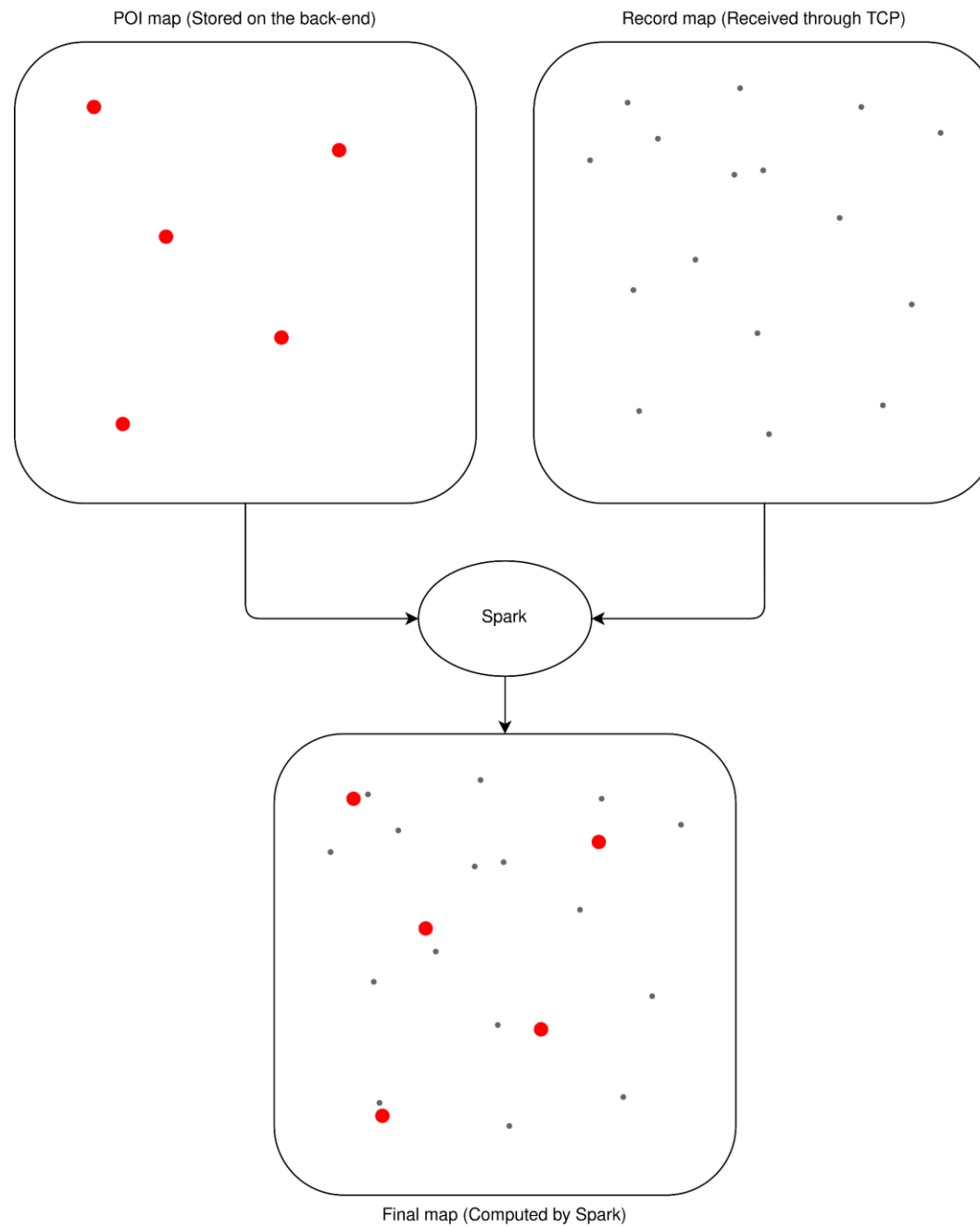
◆ Entity

▲ Direction

■ Noise range

· Sensor









X	Y	VALUE	TIMESTAMP
10.1	15.3	52.7	2022-06-11 16:43:24.000



POI	VALUE	TIMESTAMP
milan_cathedral	52.7	2022-06-11 16:43:24.000





POI	VALUE	TIMESTAMP
milan_cathedral	52.7	2022-06-11 16:43:24.000
trevi_fountain	41.2	2022-06-11 16:45:52.000
milan_cathedral	61.5	2022-06-11 16:46:17.000
pisa_tower	31.2	2022-06-11 16:46:25.000
pisa_tower	76.5	2022-06-11 16:47:44.000
colosseum	49.6	2022-06-11 16:51:13.000



```
hourlyAverage = source
    .where(col("TIMESTAMP")
        .gt(current_timestamp()
            .minus(expr("INTERVAL 1 HOUR")))))
    .select("POI", "VALUE")
    .groupBy("POI")
    .avg("VALUE");
```



```
top10poi = hourlyAverage  
    .orderBy(desc("avg(VALUE)"))  
    .limit(10);
```



```
noiseStreak = source
    .where(col("VALUE").gt(threshold))
    .groupBy("POI")
    .agg(max("TIMESTAMP"))
    .withColumn("STREAK", current_timestamp()
        .minus(col("max(TIMESTAMP)")))
    .orderBy(desc("STREAK"))
    .select("POI")
    .limit(1);
```