

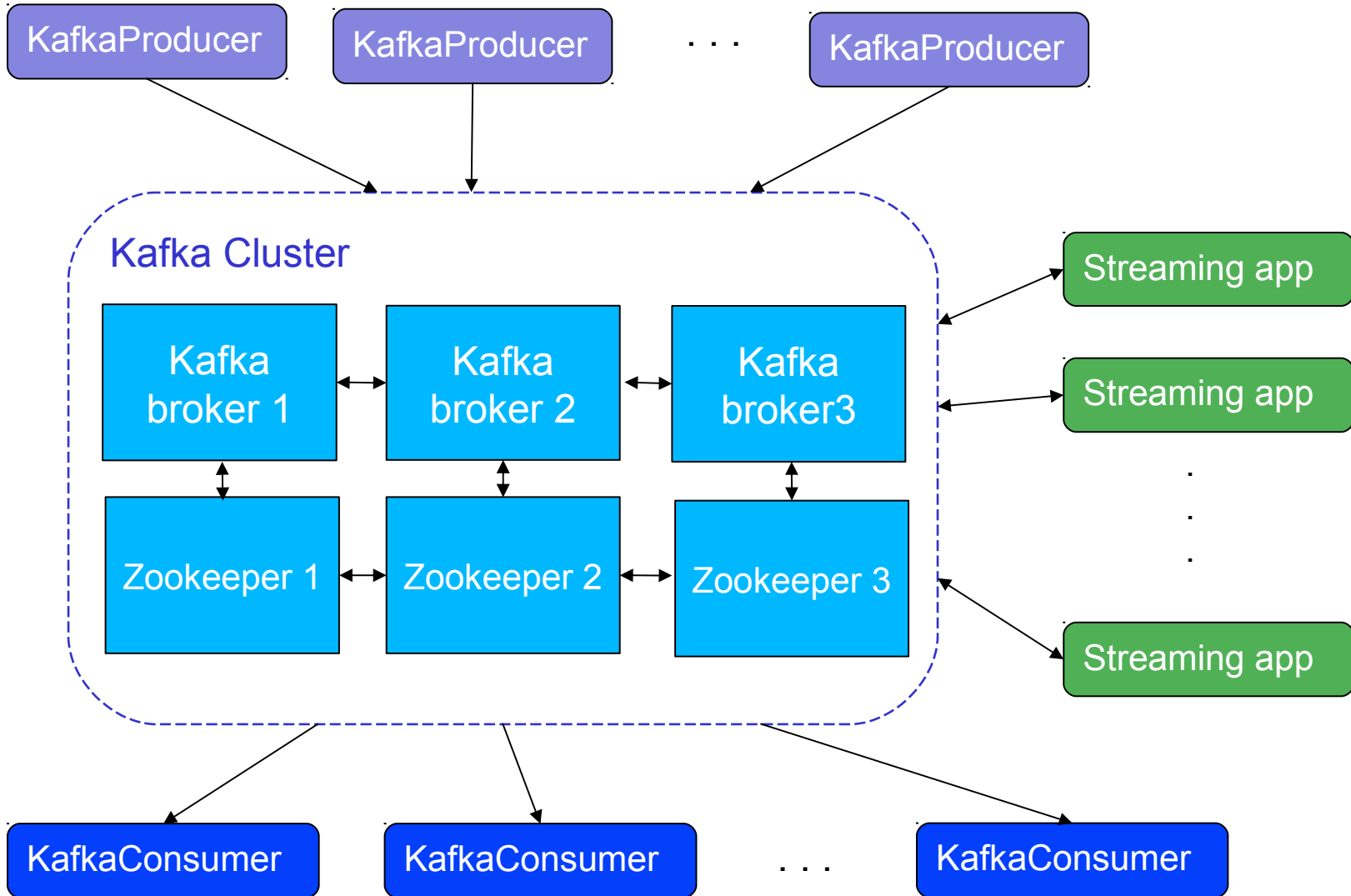
# Kafka Monitoring

Ivan Turčinović

ivan.turcinovic@inovatrend.com



# Kafka ecosystem

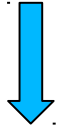


# Kafka Monitoring

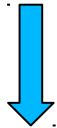
- **Why**
  - Troubleshooting
  - Identifying potential problems
  - Optimize performance
- **What**
  - Logs
    - Kafka logs
    - System logs
  - System resource utilization
  - **Kafka JMX metrics**

# Demo monitoring setup

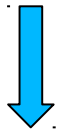
## Kafka JMX Metrics



jmxtrans



Graphite



Grafana

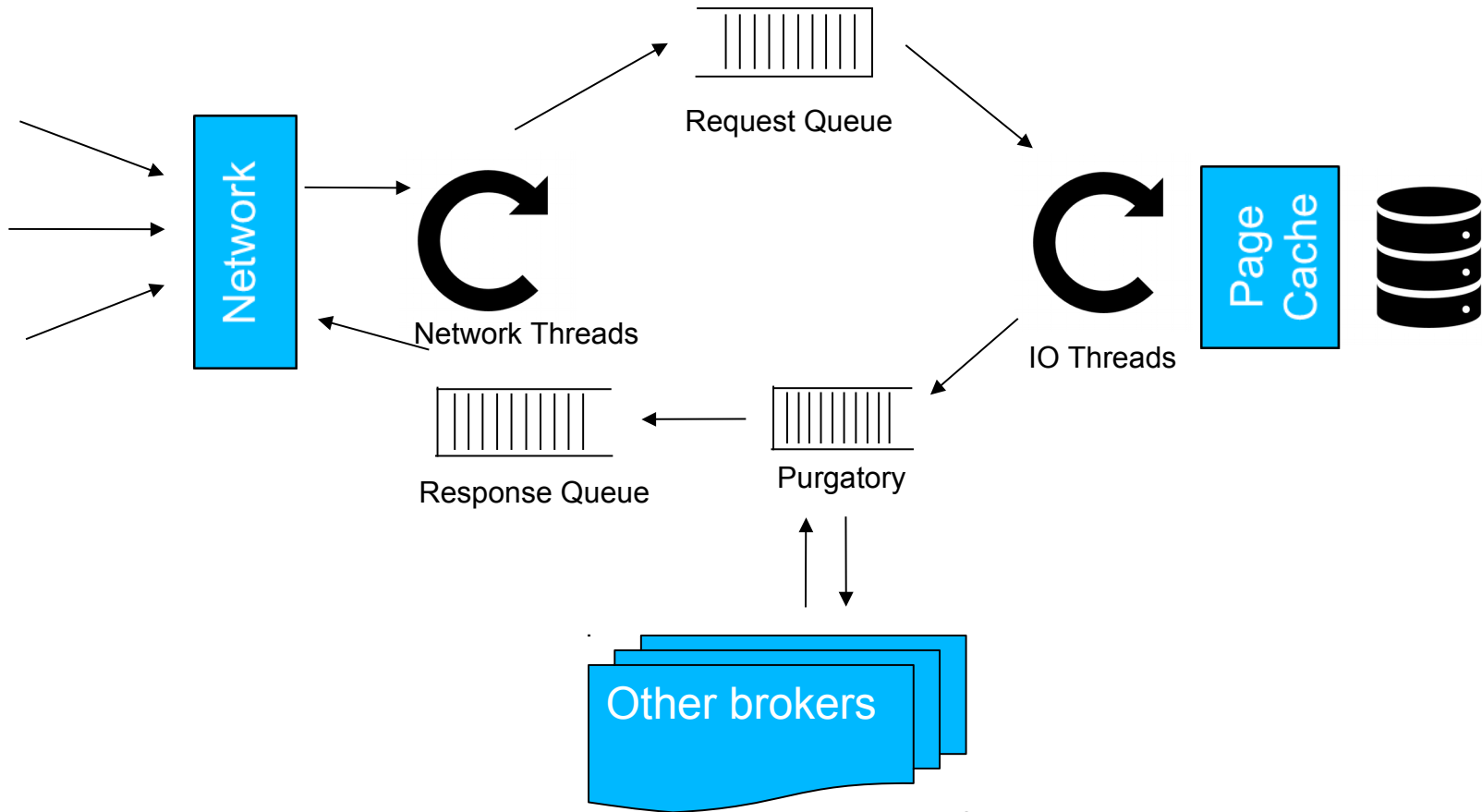


# Broker Metrics

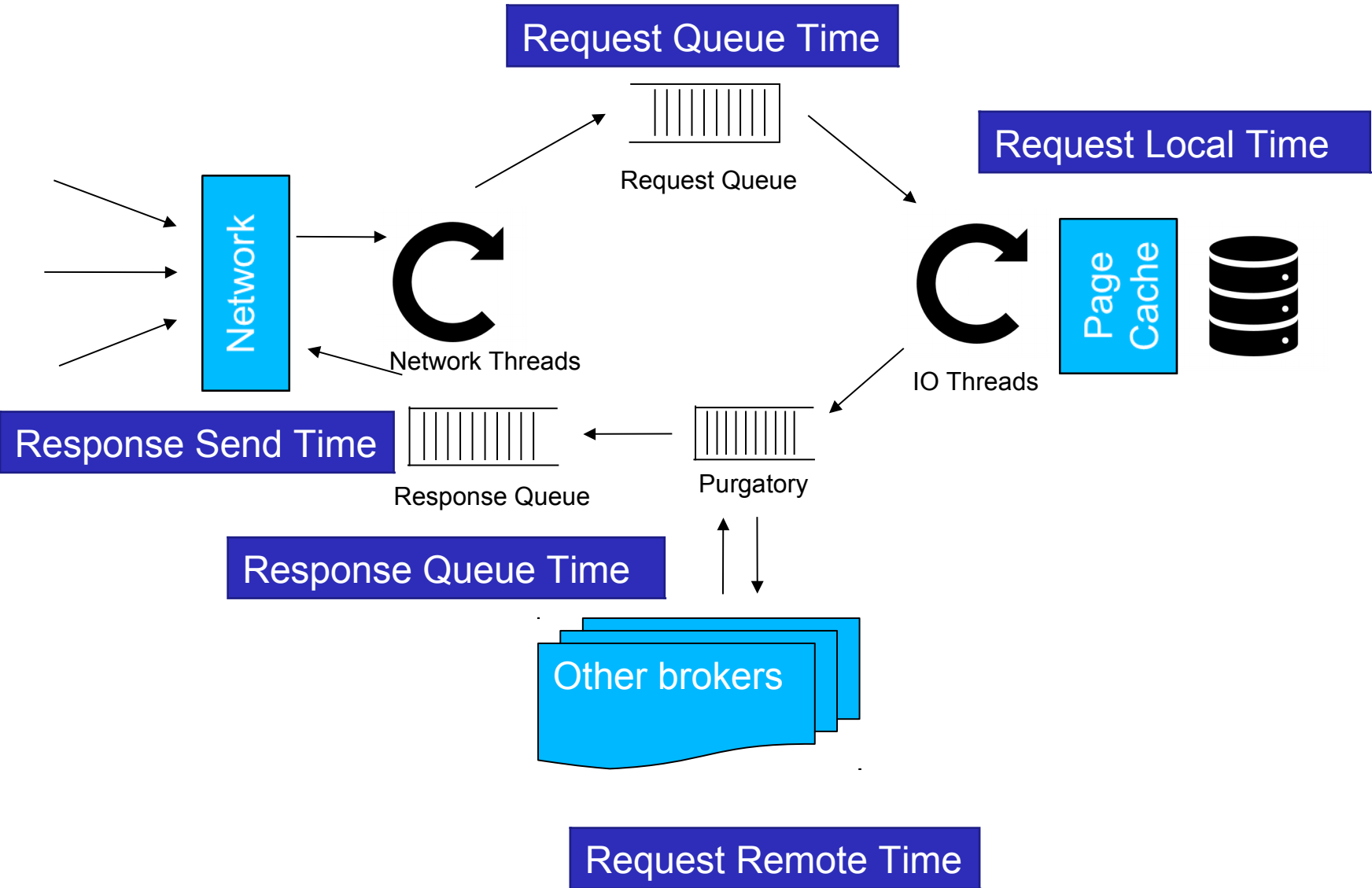
- Broker State
- ActiveControllerCount
- OfflinePartitionsCount
- UnderReplicatedPartitions
- PartitionCount
- LeaderCount
- BytesInPerSec
- MessagesInPerSec
- BytesOutPerSec
- CPU Load
- Memory Usage

# Broker Performance Metrics

Produce/Fetch Requests



# Broker Performance Metrics



# Broker Performance Metrics

## Total Time:

- kafka.network:type=RequestMetrics,name=TotalTimeMs,request=Produce
- kafka.network:type=RequestMetrics,name=TotalTimeMs,request=FetchConsumer
- kafka.network:type=RequestMetrics,name=TotalTimeMs,request=FetchFollower

## Break Down Times:

- RequestQueueTimeMs
- ResponseSendTimeMs
- ResponseQueueTimeMs
- LocalTimeMs
- RemoteTimeMs

## Thread Pools:

- kafka.server:type=KafkaRequestHandlerPool,name=RequestHandlerAvgIdlePercent
- kafka.network:type=SocketServer,name=NetworkProcessorAvgIdlePercent



# Producer Metrics

- **Producer metrics are picked up from producer JVM, not broker**
- **kafka.producer:type=producer-metrics,client-id=client\_id**
  - outgoing-byte-rate
  - record-error-rate
  - record-retry-rate
  - record-send-rate
  - record-size-avg

# Consumer Metrics

- **Consumer metrics are picked up from consumer JVM, not broker**
- **kafka.consumer:type=consumer-fetch-manager-metric,client-id=client\_id**
  - bytes-consumed-rate
  - records-consumed-rate
  - **records-lag**
  - fetch-latency-avg
  - fetch-rate
  - records-per-request-avg

# Consumer Lag Monitoring

- **kafka-consumer-groups** command

TOPIC	PARTITION	CURRENT-OFFSET	LOG-END-OFFSET	LAG
javacro-topic-2	0	8049	15049	7000

- **LinkedIn Burrow**
  - <https://github.com/linkedin/Burrow>
- **Remora**
  - <https://github.com/zalando-incubator/remora>

# Confluent Control Center

- **Commercial product from Confluent**
- **Included in Confluent Platform Enterprise**
- **Monitoring**
  - End-to-end stream monitoring
    - Confluent Monitoring Interceptors
    - Confluent Metrics Reporter
  - System health
  - Alerting

# Thank You!

