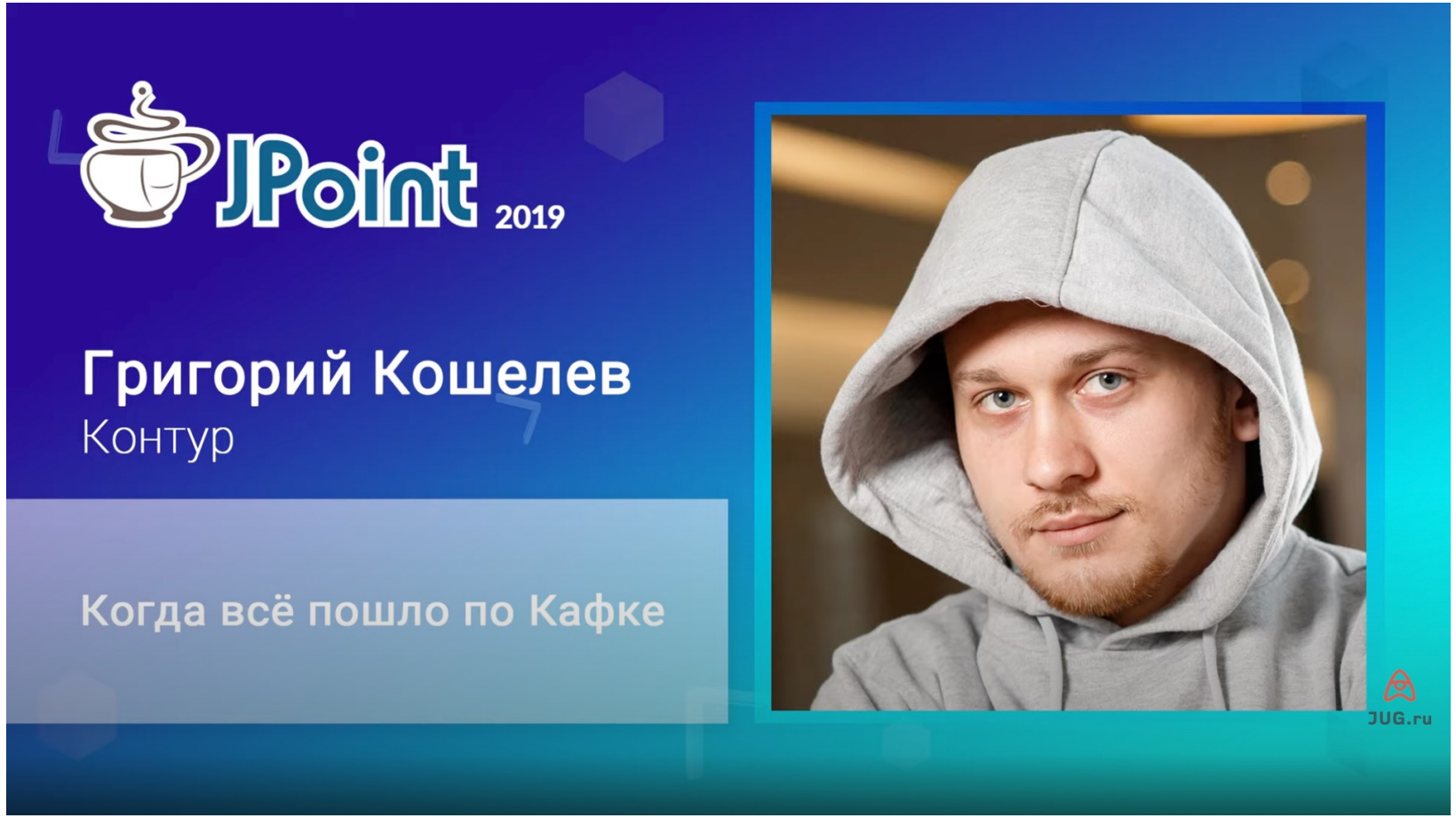
Когда всё пошло по Кафке 3: Kafka Consumer

Григорий Кошелев

Контур

JPoint 2023

Когда всё пошло по Кафке

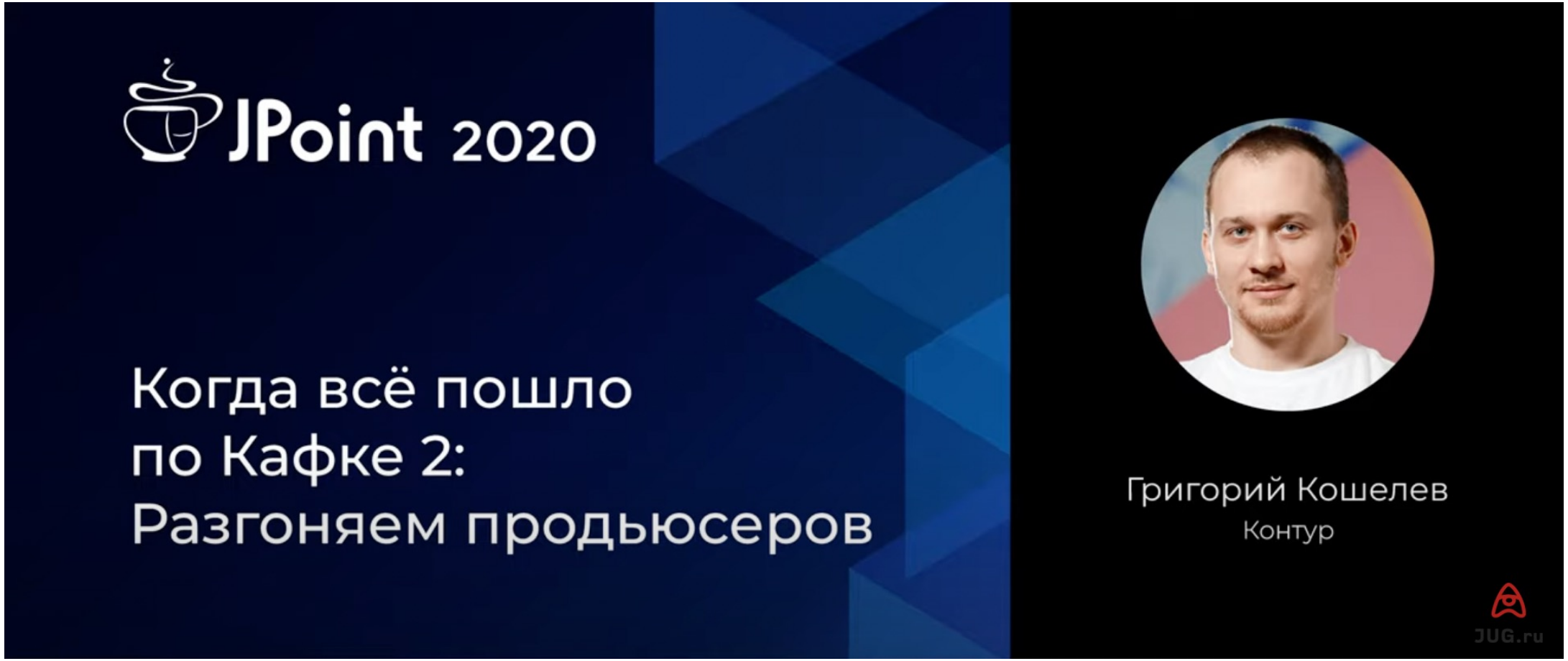


https://www.youtube.com/watch?v=A_yUaPARv8U

Как готовить Кафку, чтобы не пригорало

https://www.youtube.com/watch?v=M3HTM81P-Sg

Когда всё пошло по Кафке



https://www.youtube.com/watch?v=zMLfxztAVlo

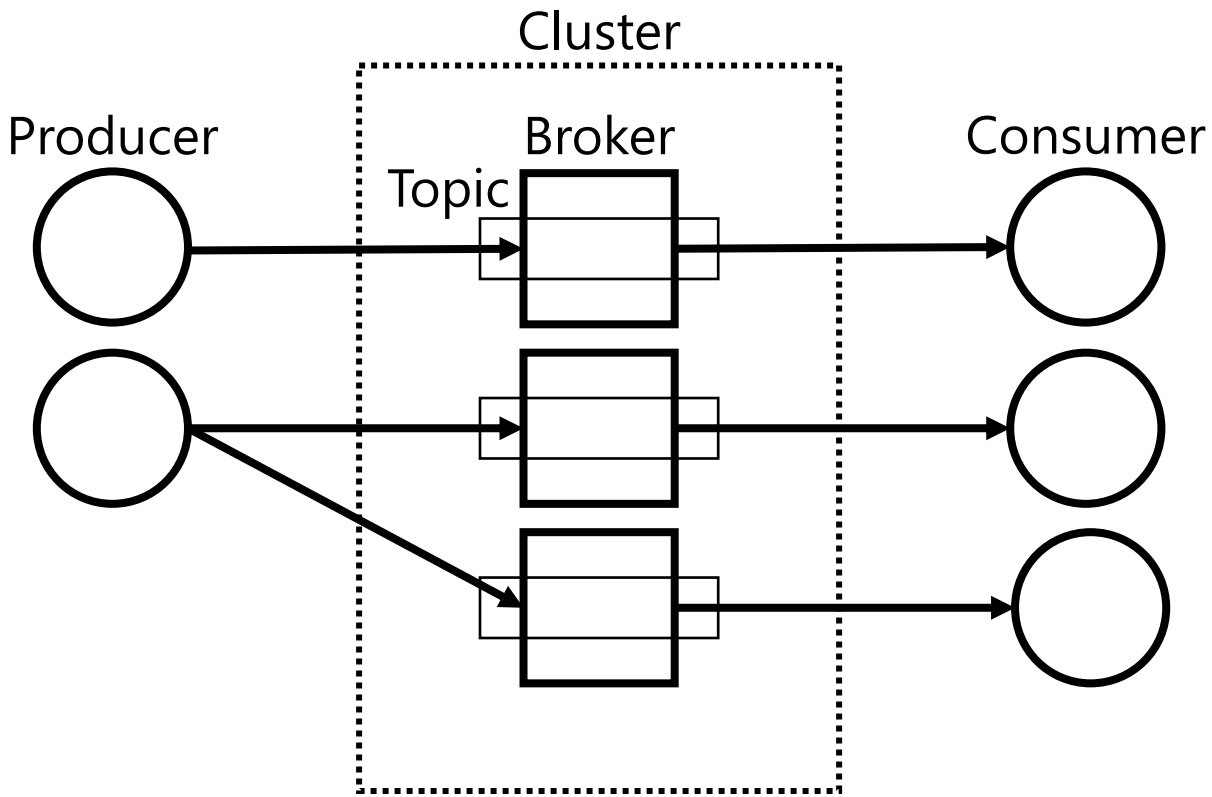
Когда всё пошло по Кафке

— https://www.youtube.com/watch?v=A_yUaPARv8U
Когда всё пошло по Кафке (JPoint 2019)

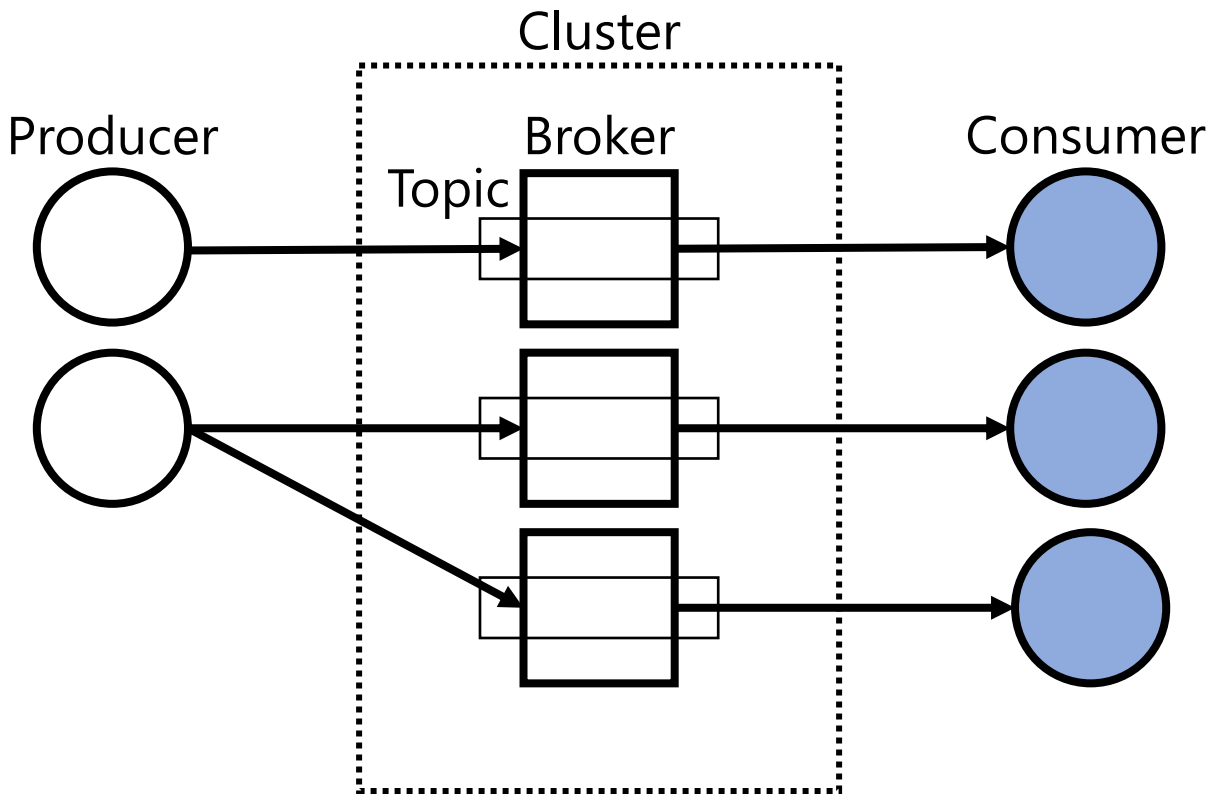
— https://www.youtube.com/watch?v=zMLfxztAVlo
Когда всё пошло по Кафке 2:
разгоняем продьюсеров (Jpoint 2020)

— https://www.youtube.com/watch?v=M3HTM81P-Sg
Как готовить Кафку, чтобы не пригорало
(DevOops 2019)

Apache Kafka



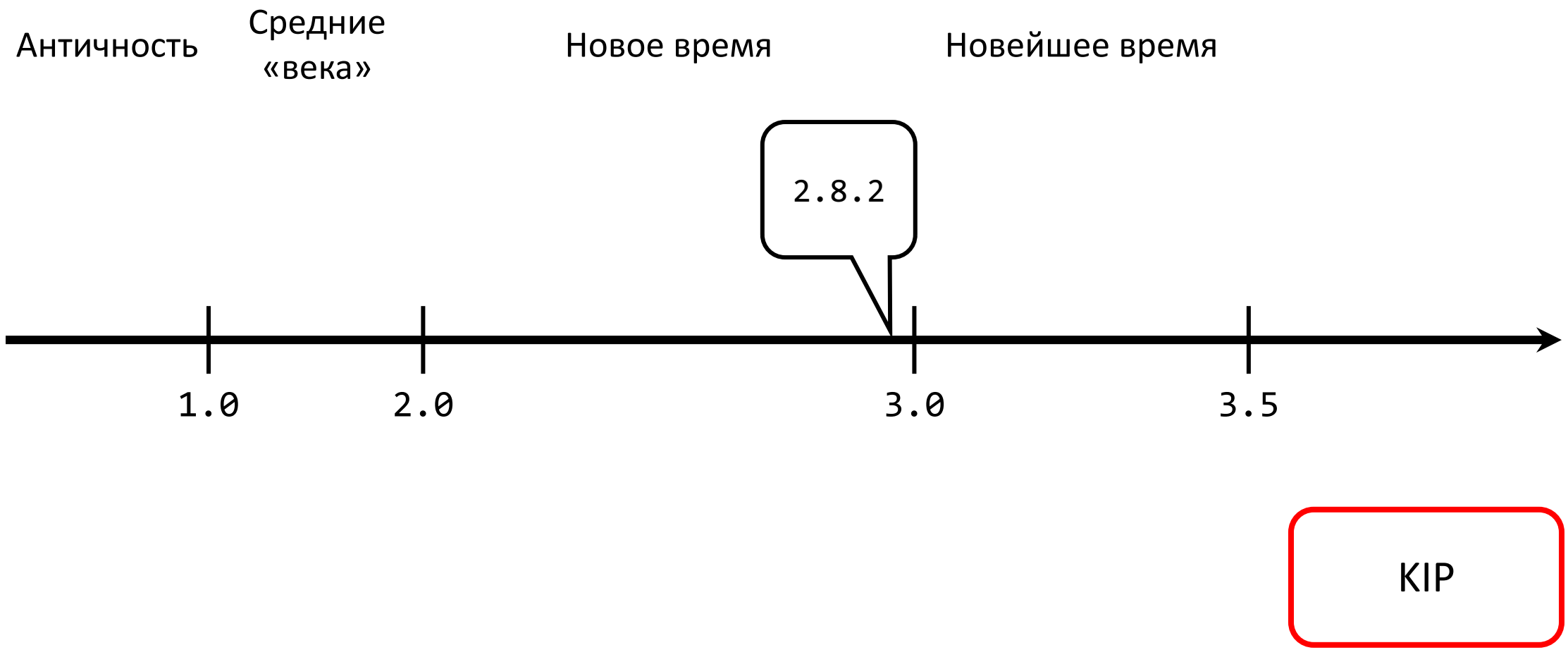
Apache Kafka



Чего ещё не будет в докладе

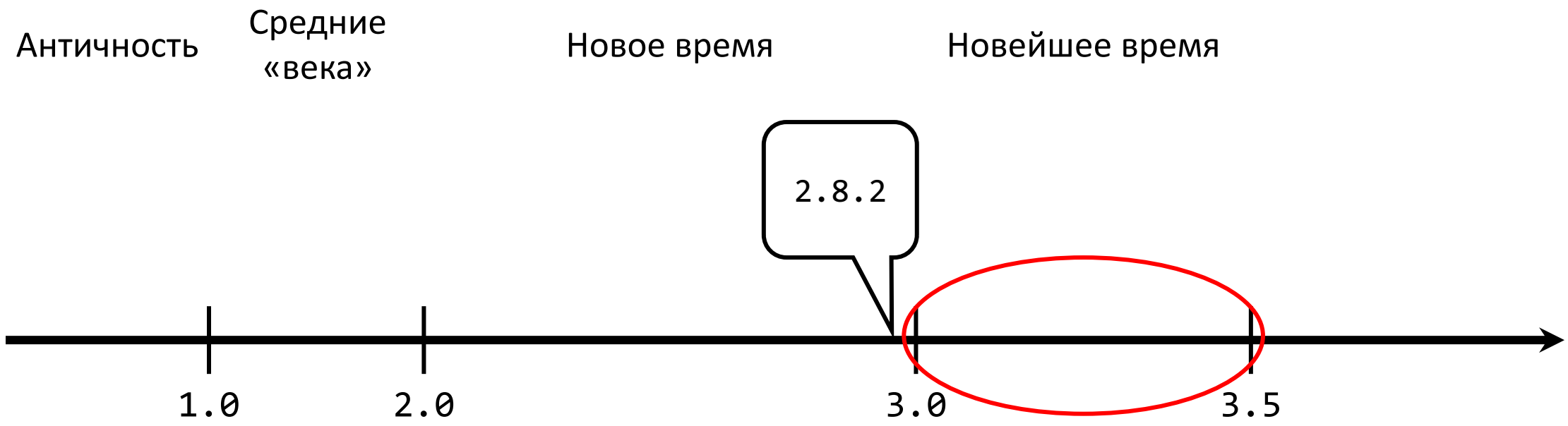
— Kafka Streams
— Kafka Connect
— Mirror Maker
— Transactions

Apache Kafka — С древнейших времён и до наших дней

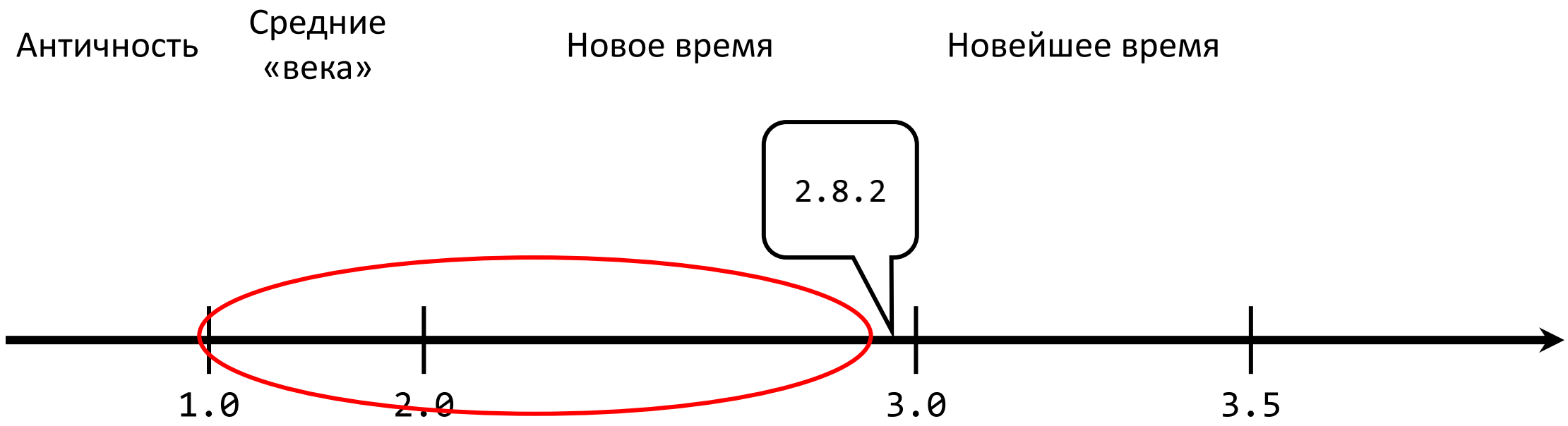


https://cwiki.apache.org/confluence/display/kafka/kafka+improvement+proposals

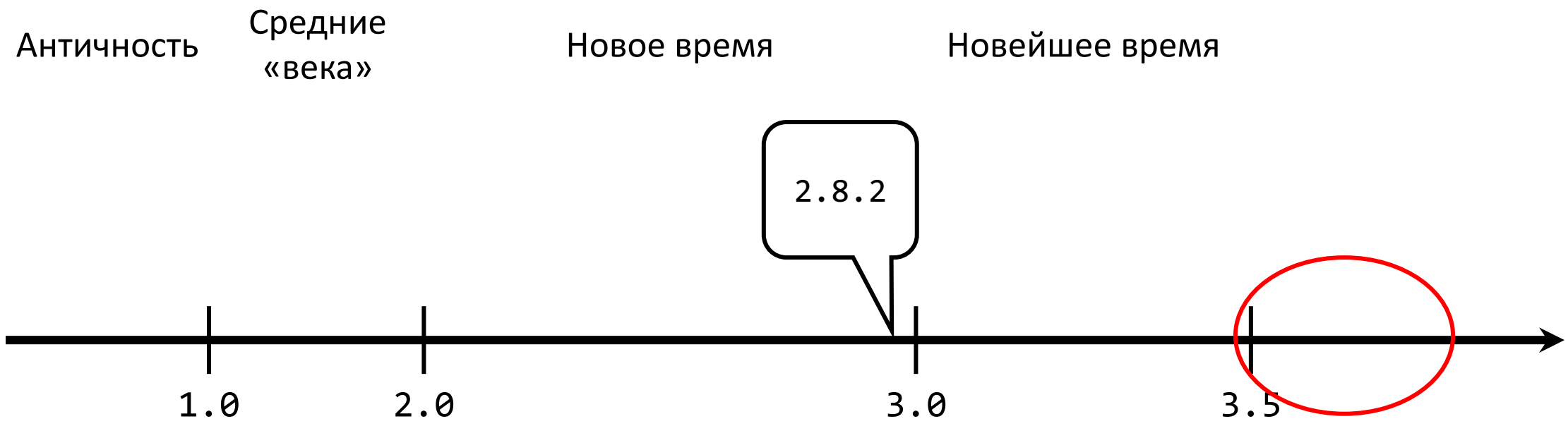
Apache Kafka — С древнейших времён и до наших дней



Apache Kafka — С древнейших времён и до наших дней

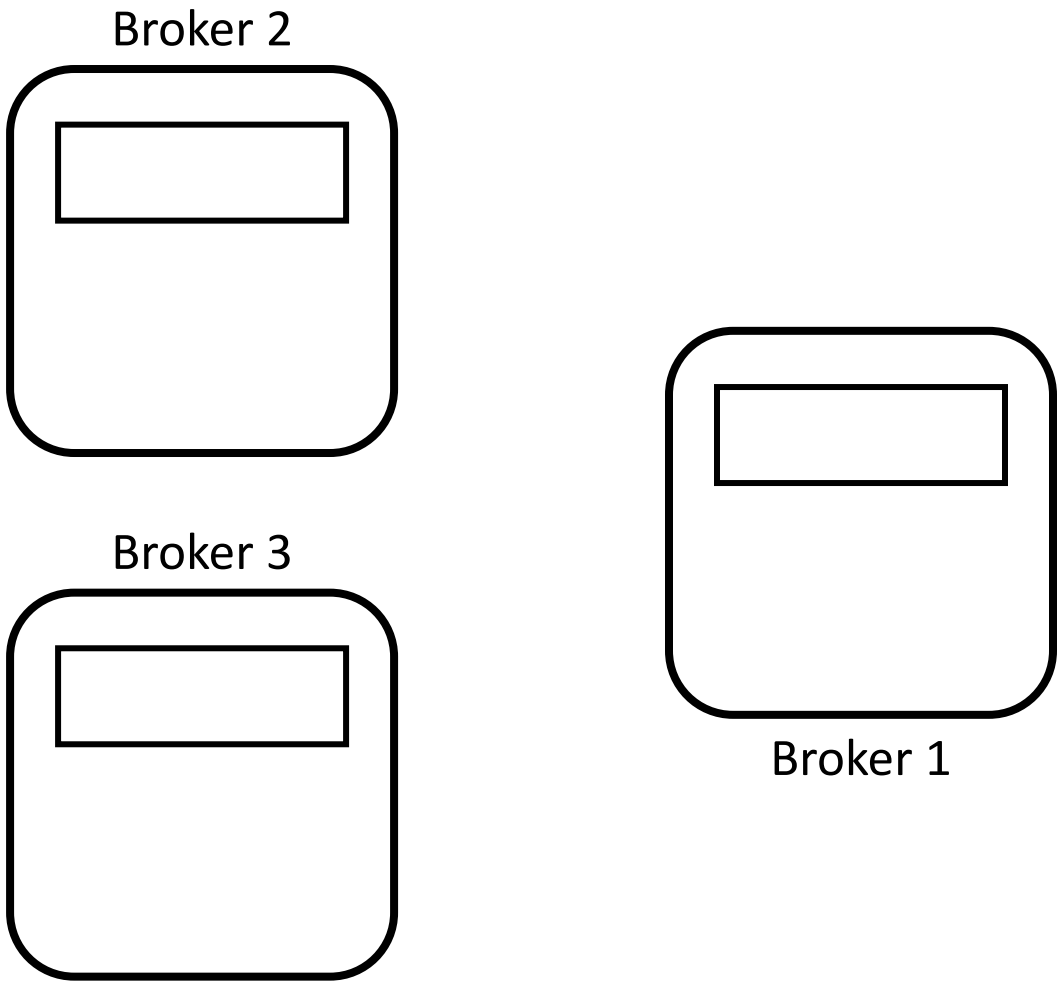


Apache Kafka — С древнейших времён и до наших дней

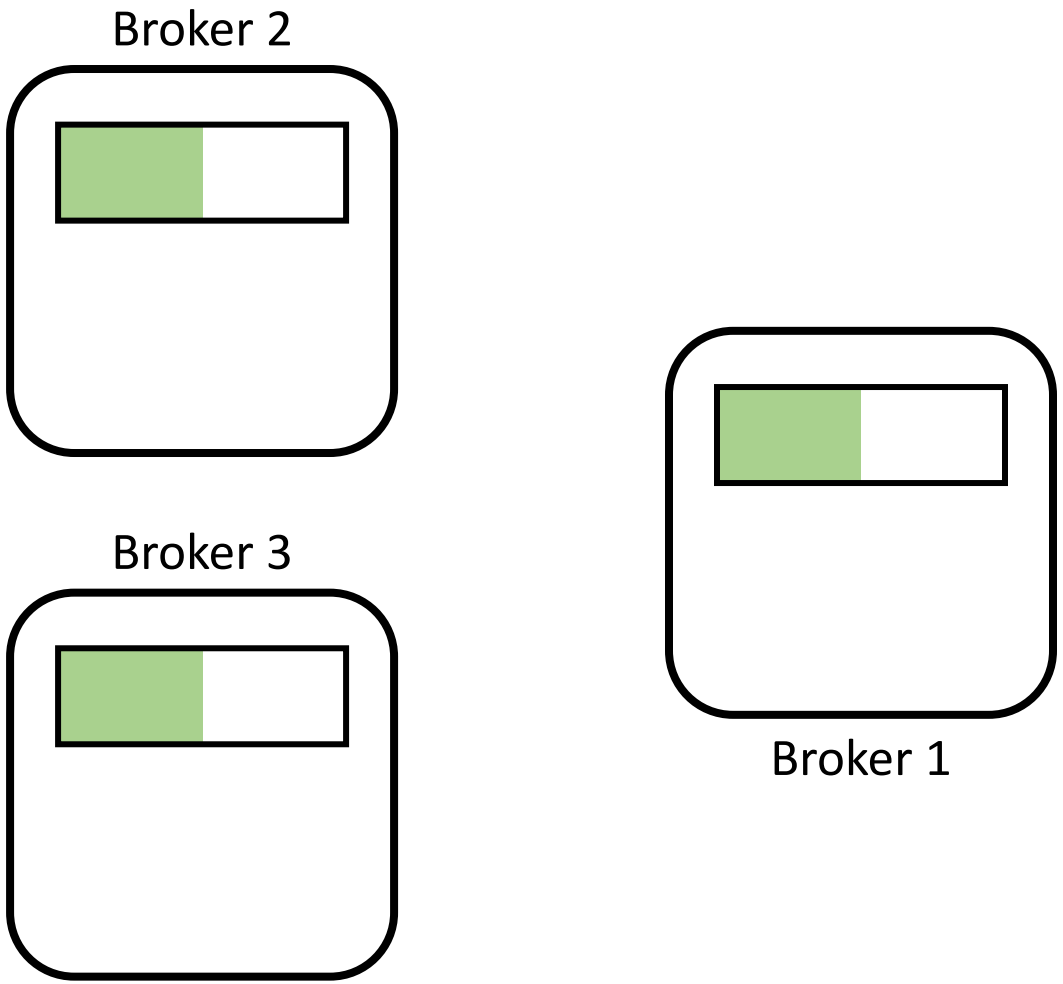


Репликация и чтение из Apache Kafka

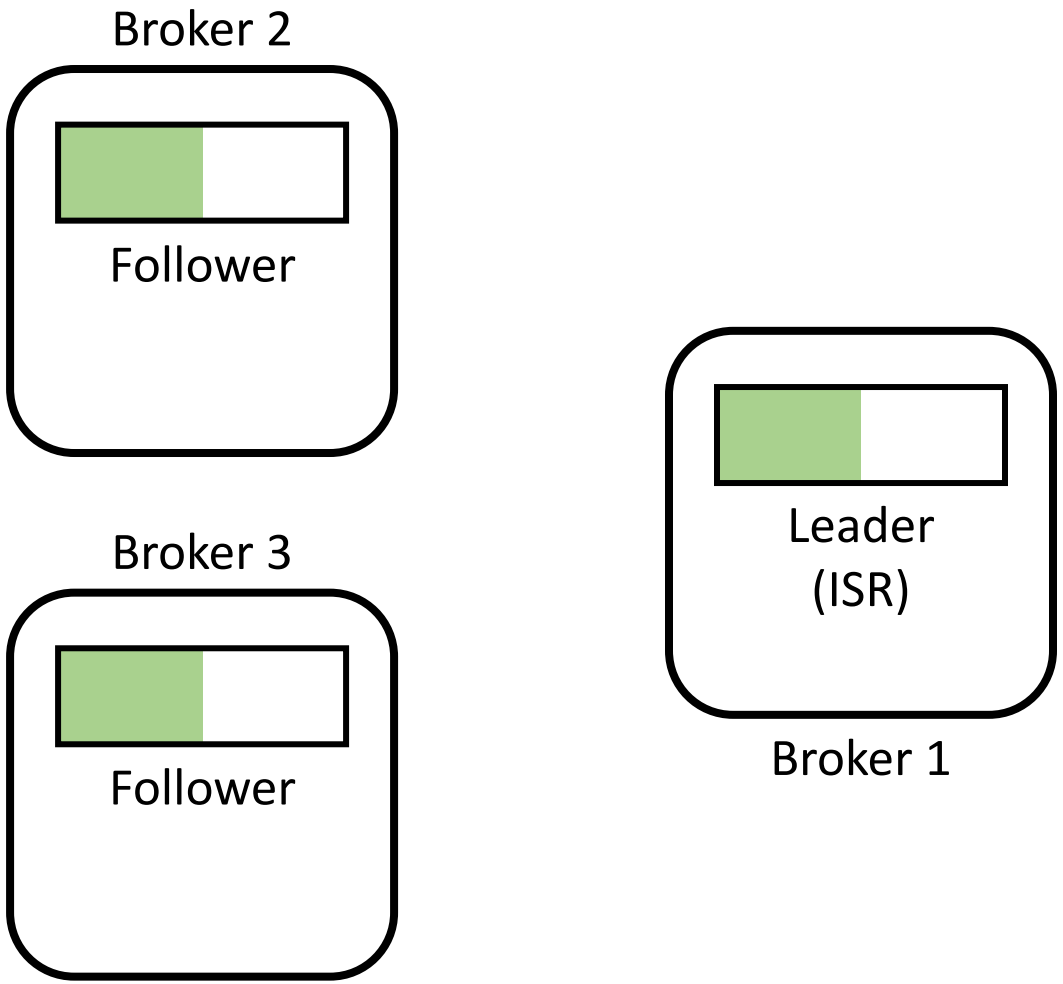
Репликация и чтение из Apache Kafka



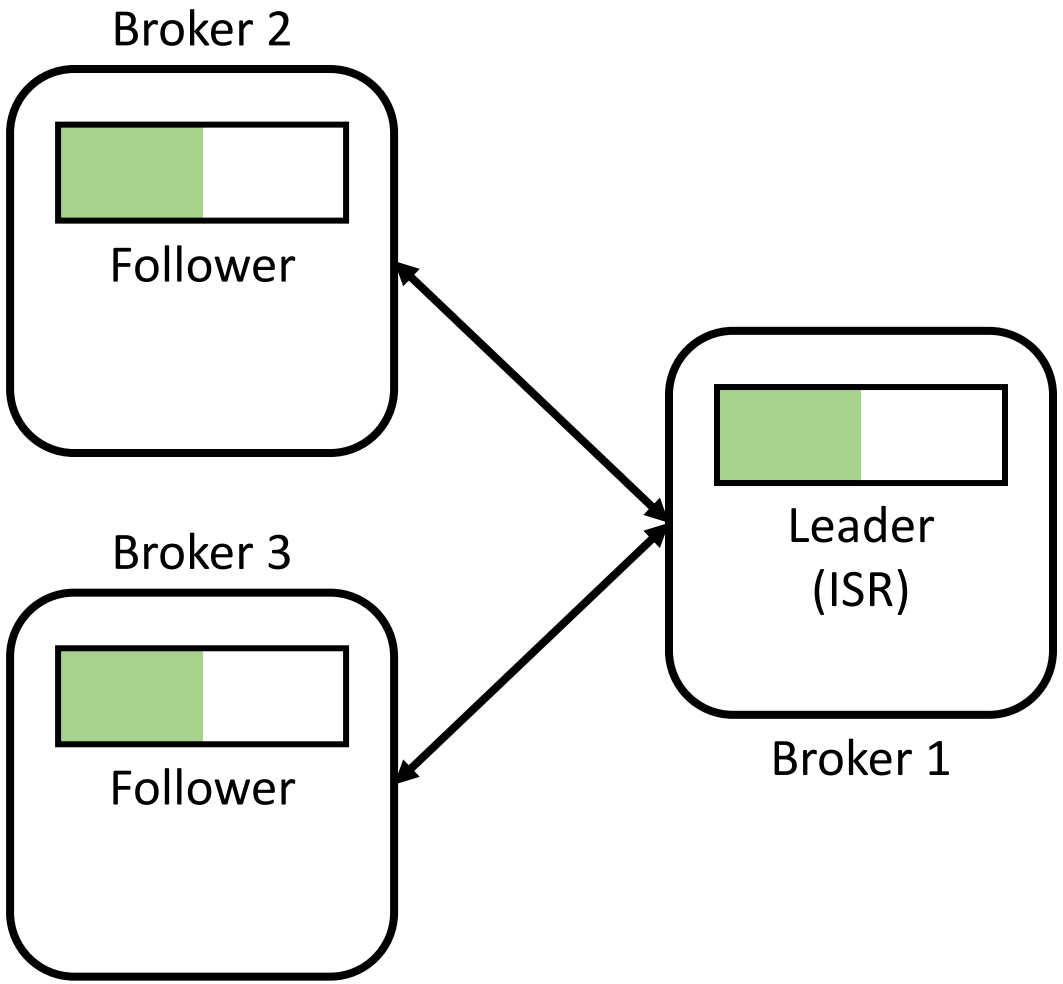
Репликация и чтение из Apache Kafka



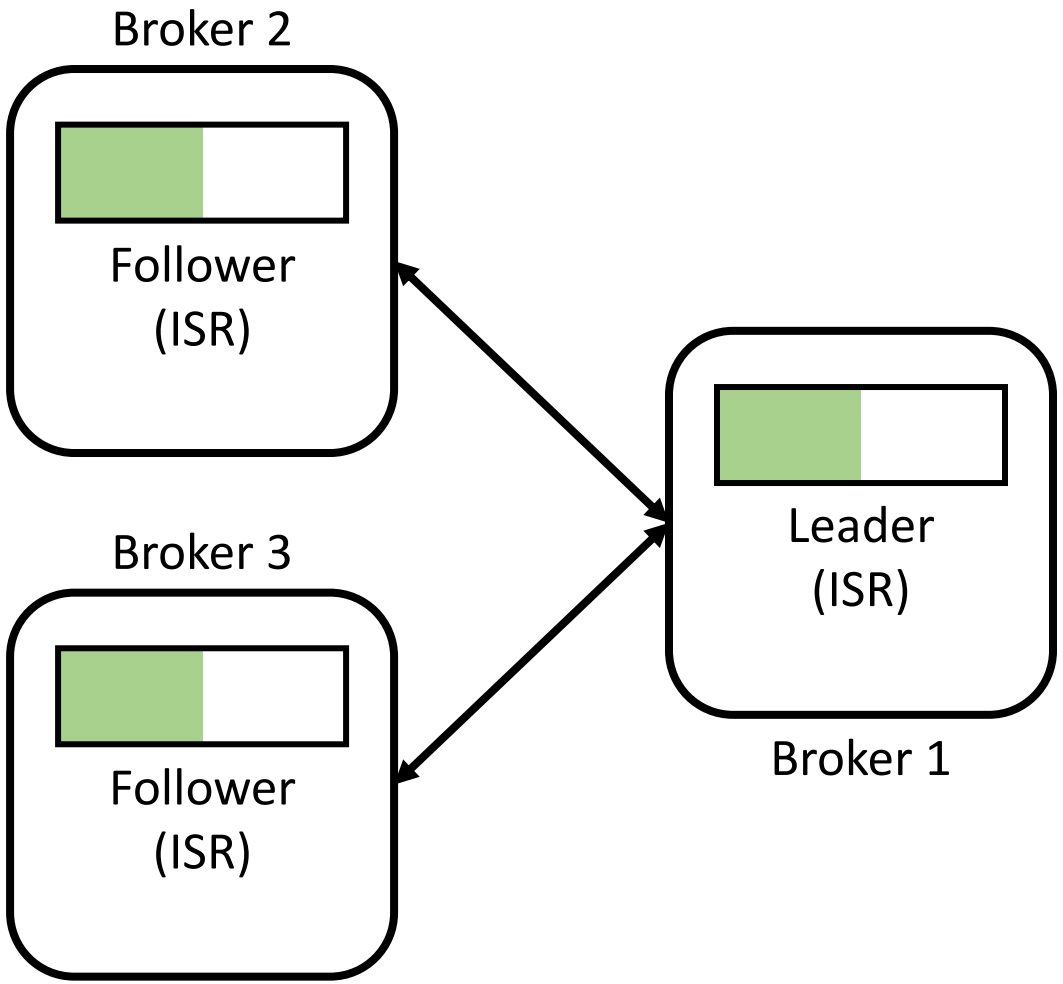
Репликация и чтение из Apache Kafka



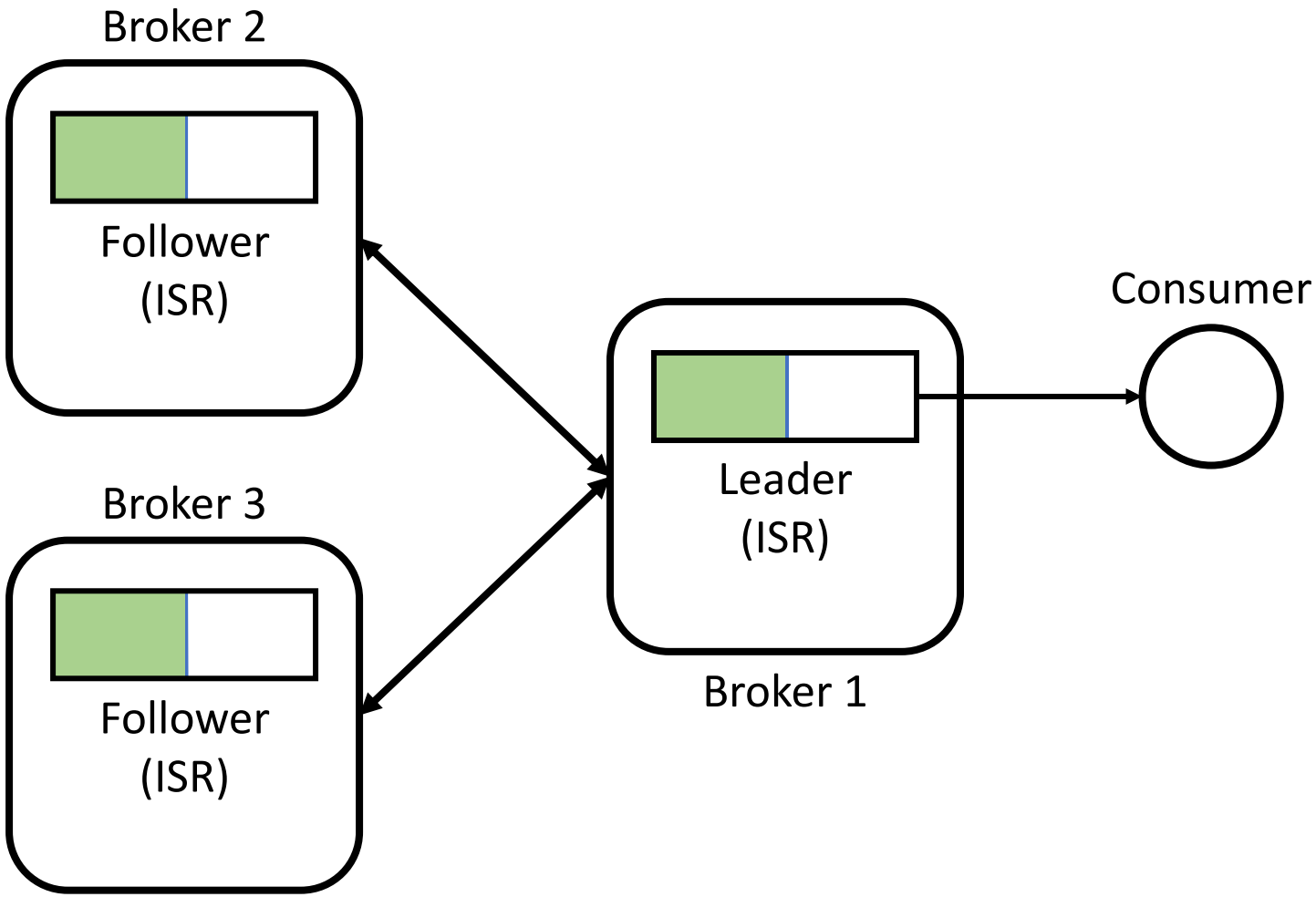
Репликация и чтение из Apache Kafka



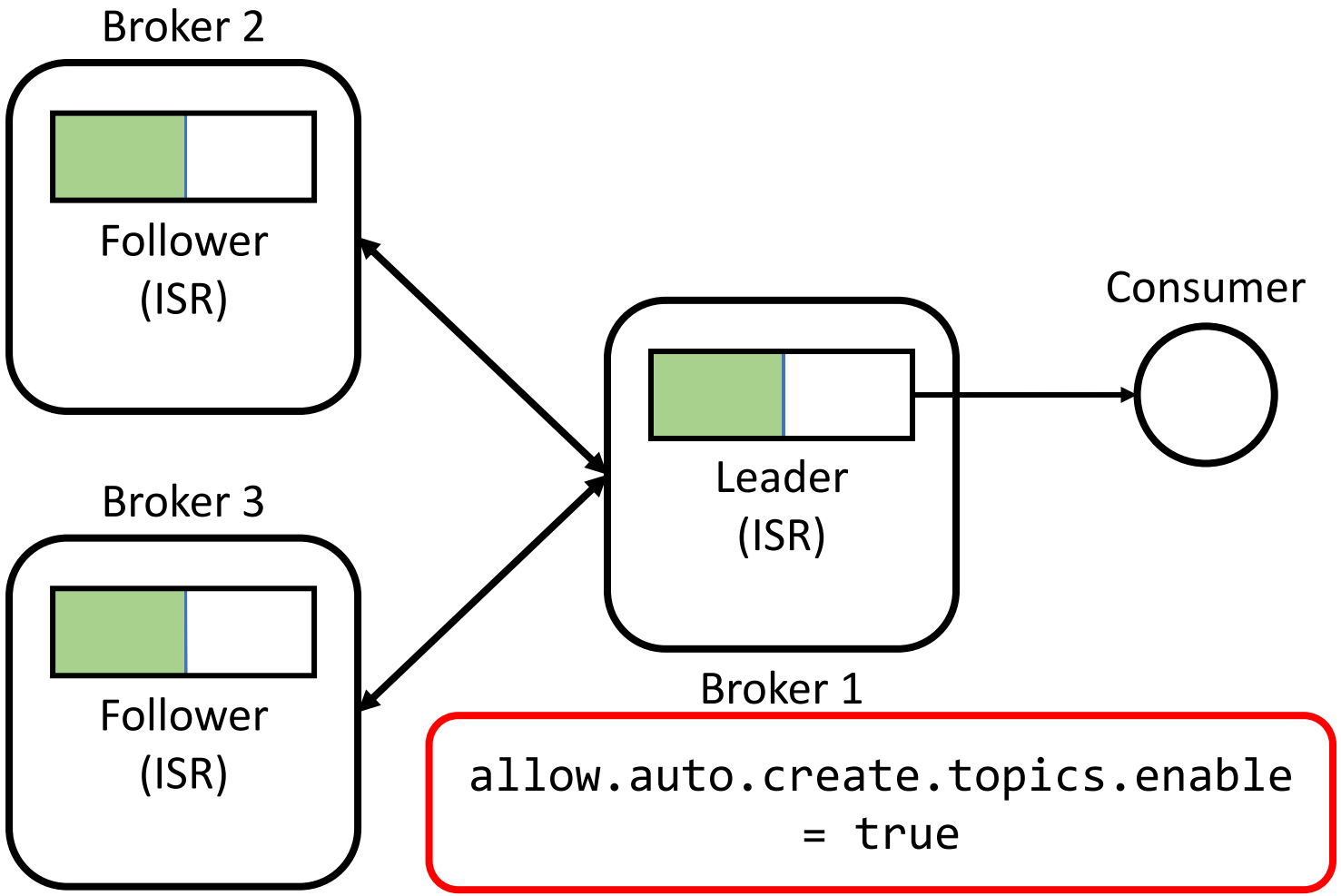
Репликация и чтение из Apache Kafka



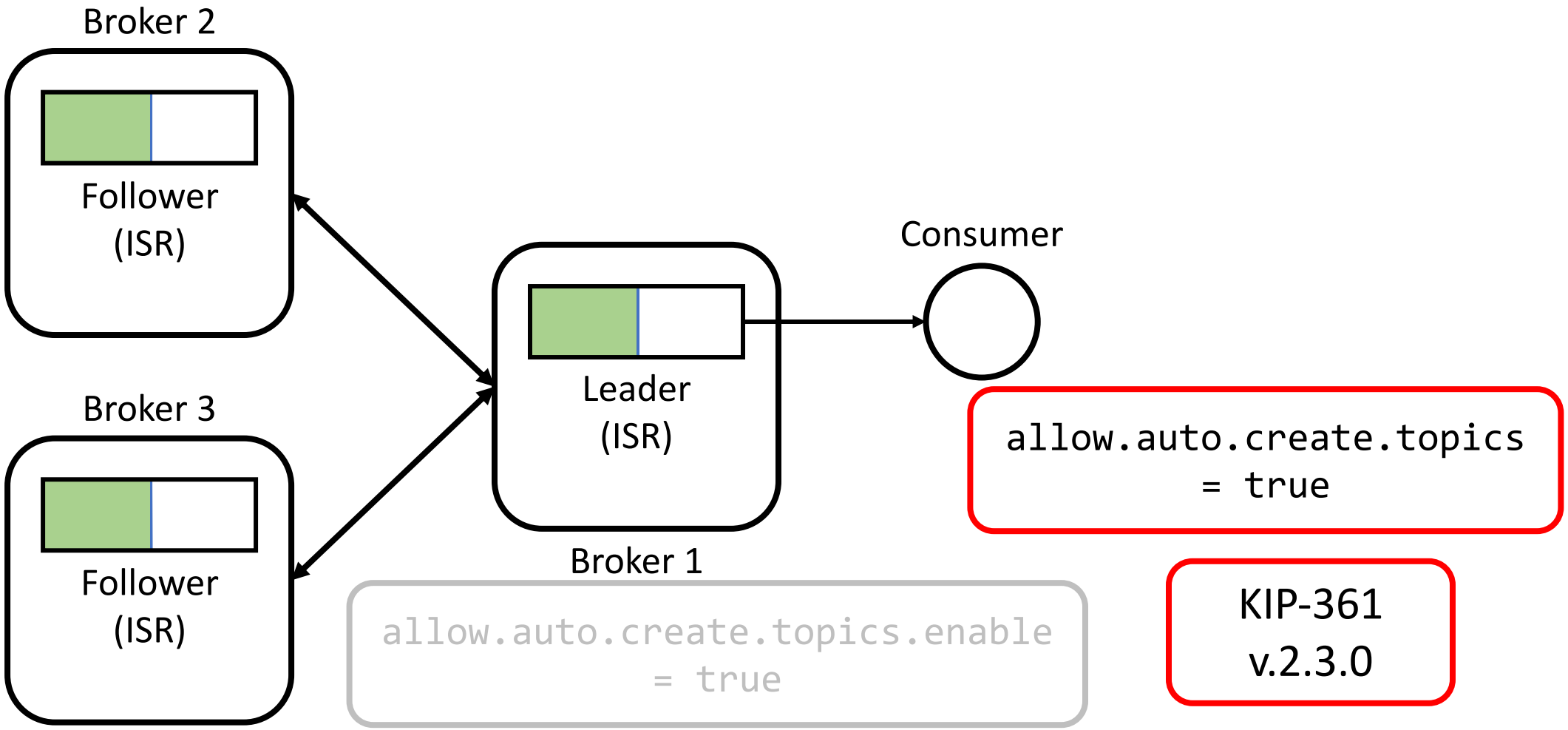
Репликация и чтение из Apache Kafka



Репликация и чтение из Apache Kafka

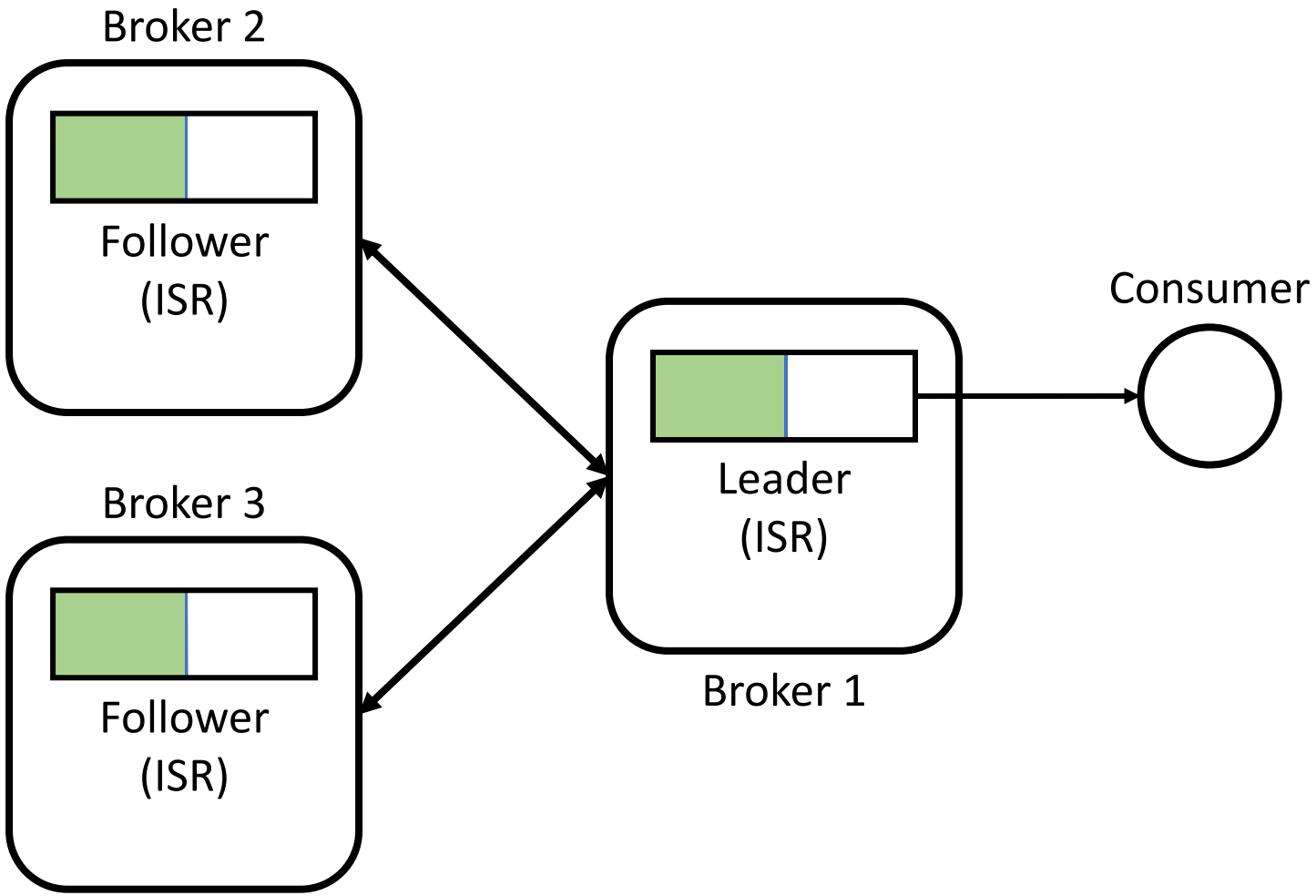


Репликация и чтение из Apache Kafka

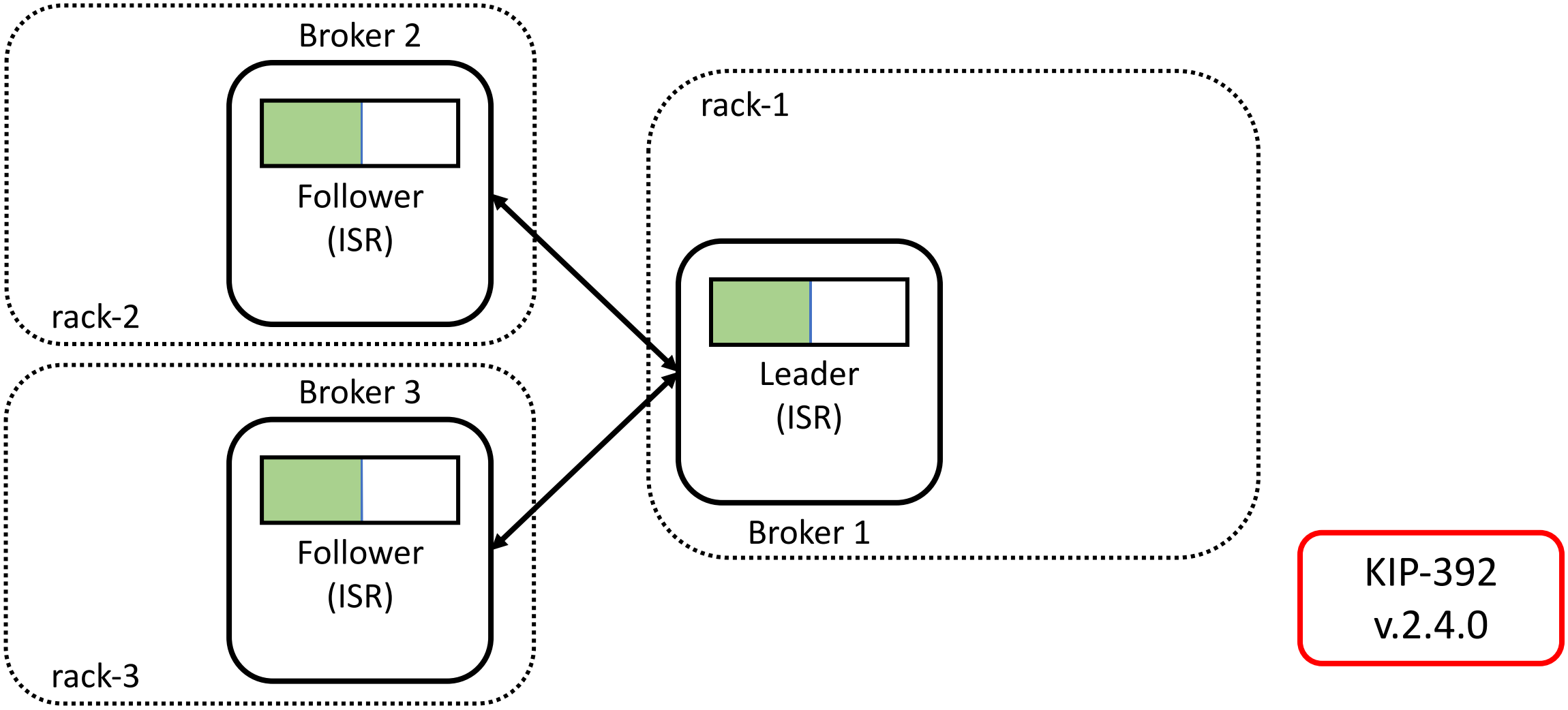


https://cwiki.apache.org/confluence/display/KAFKA/KIP-361%3A+Add+Consumer+Configuration+to+Disable+Auto+Topic+Creation

Репликация и чтение из Apache Kafka



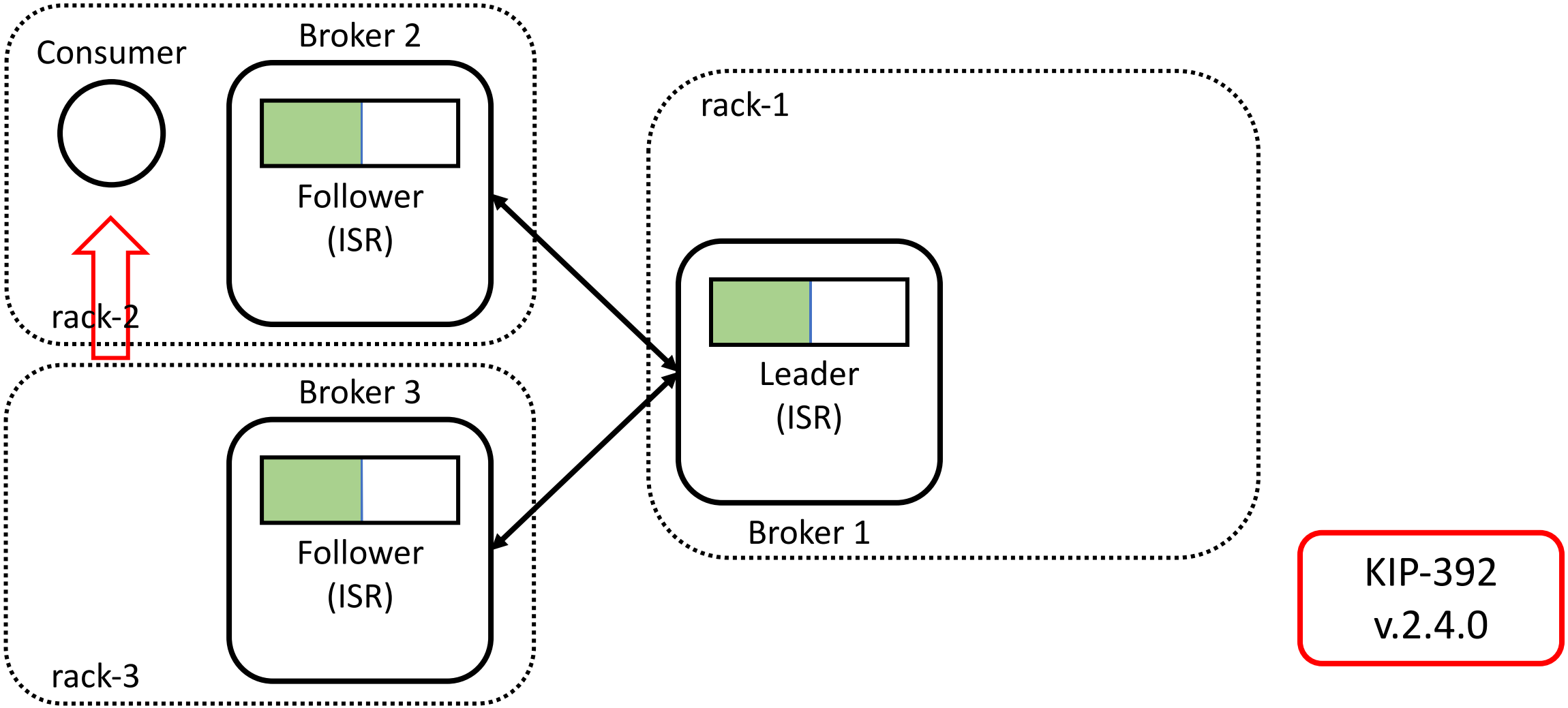
Репликация и чтение из Apache Kafka



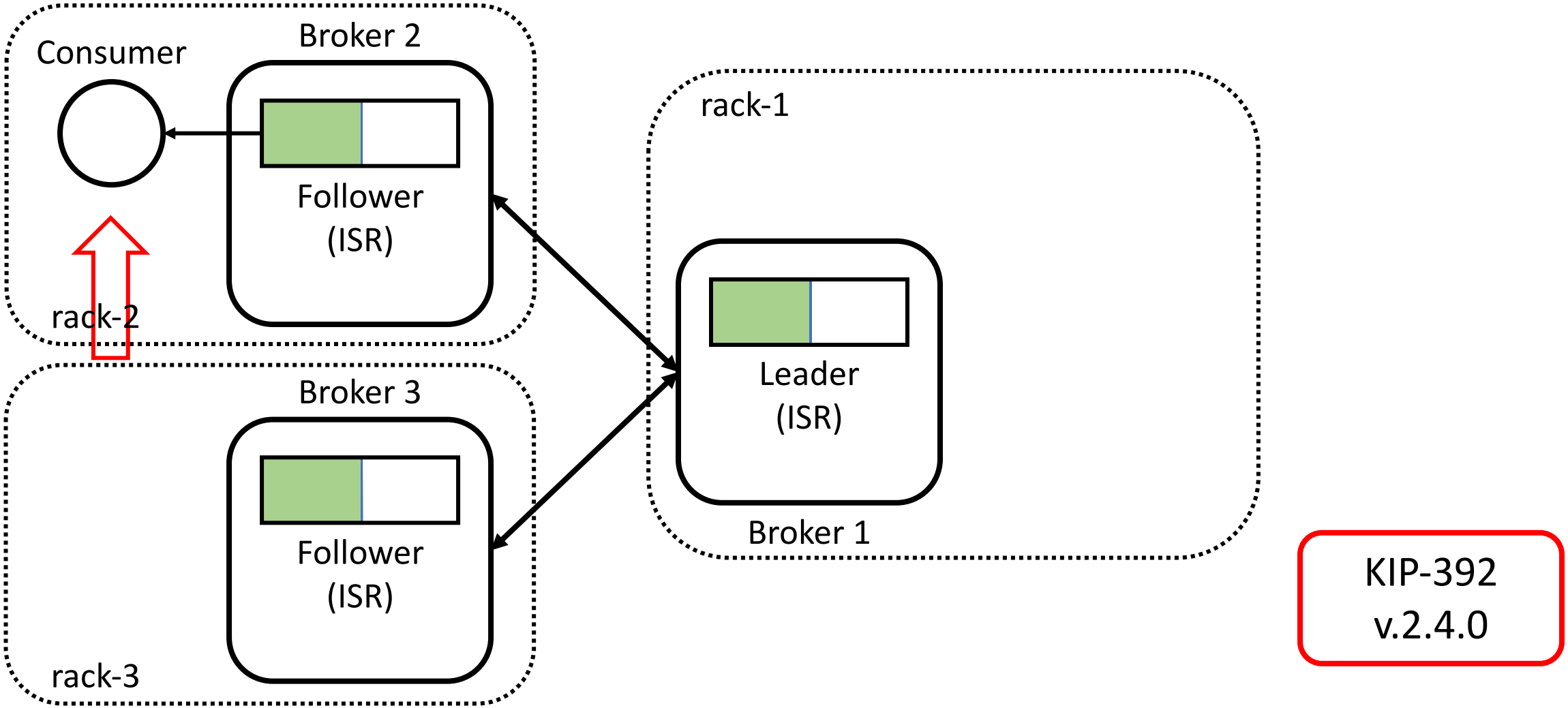
KIP-392
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-392%3A+Allow+consumers+to+fetch+from+closest+replica

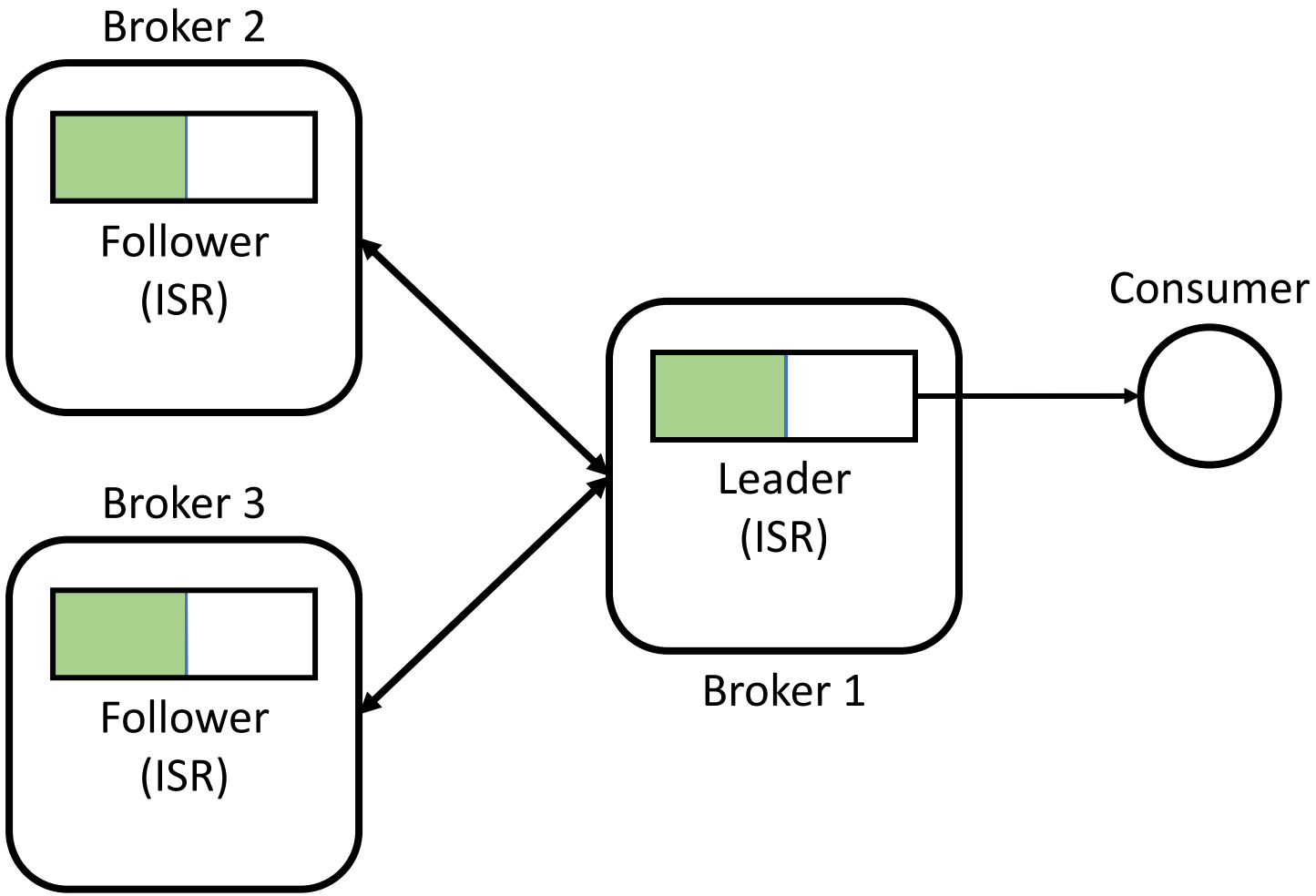
Репликация и чтение из Apache Kafka



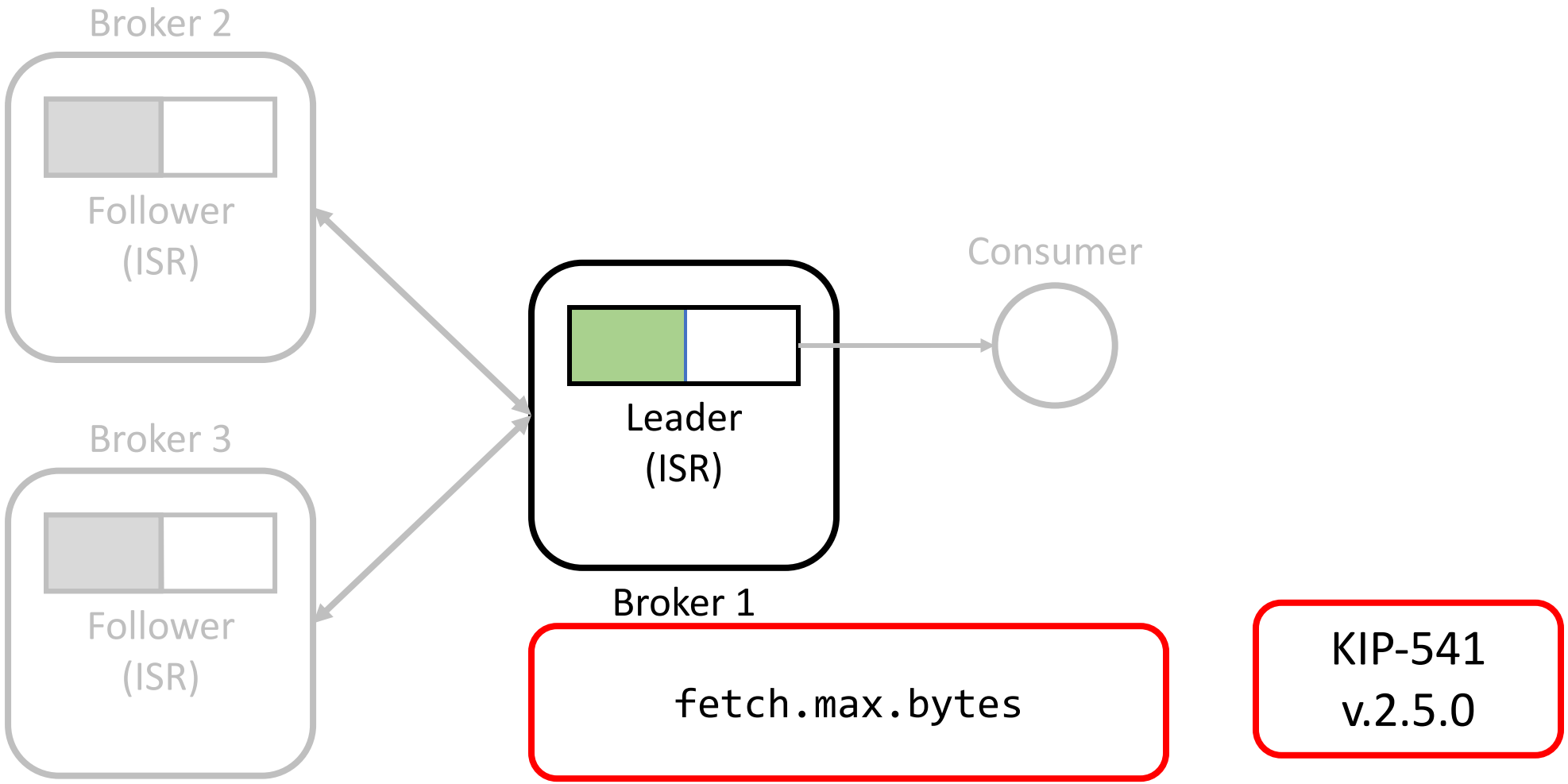
Репликация и чтение из Apache Kafka



Репликация и чтение из Apache Kafka

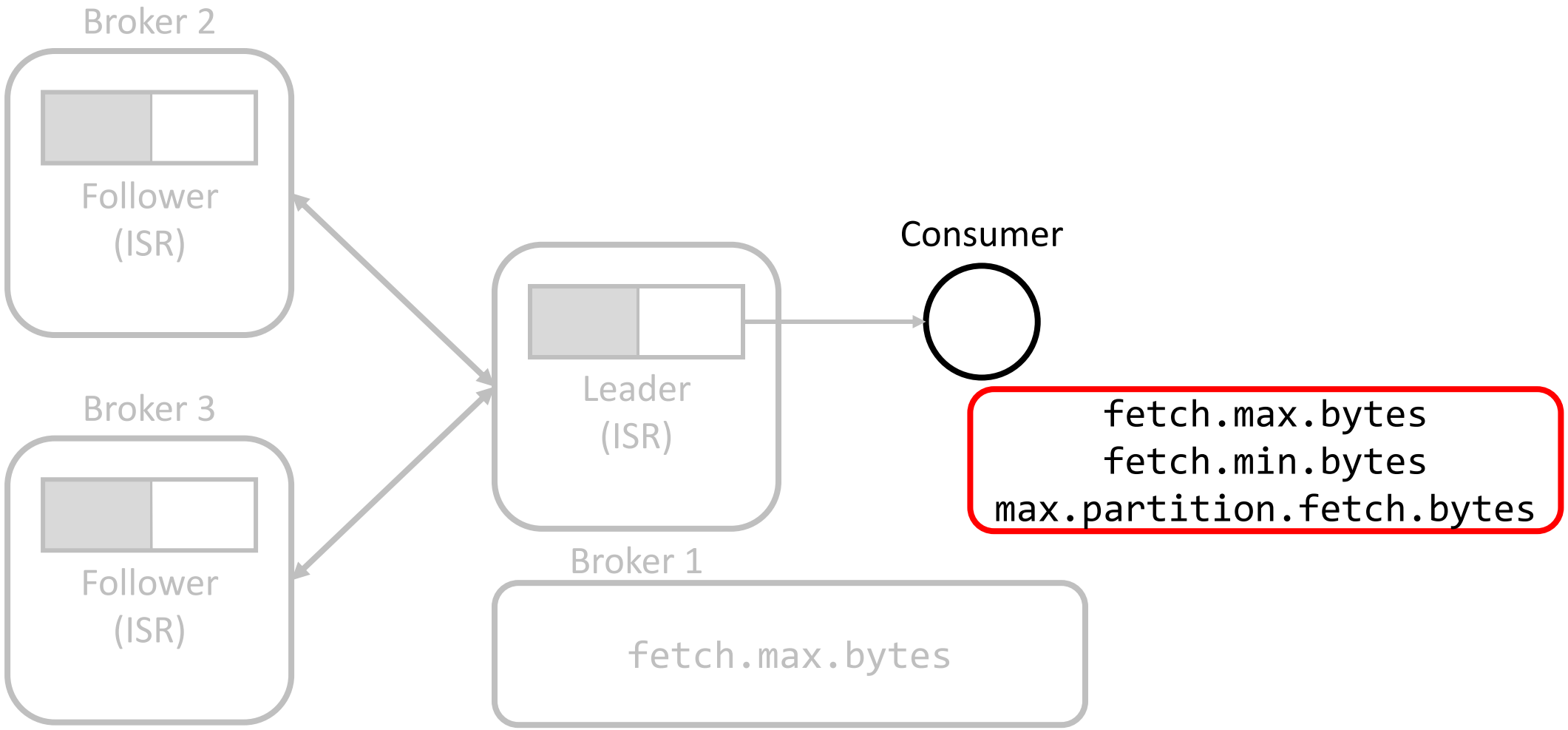


Репликация и чтение из Apache Kafka

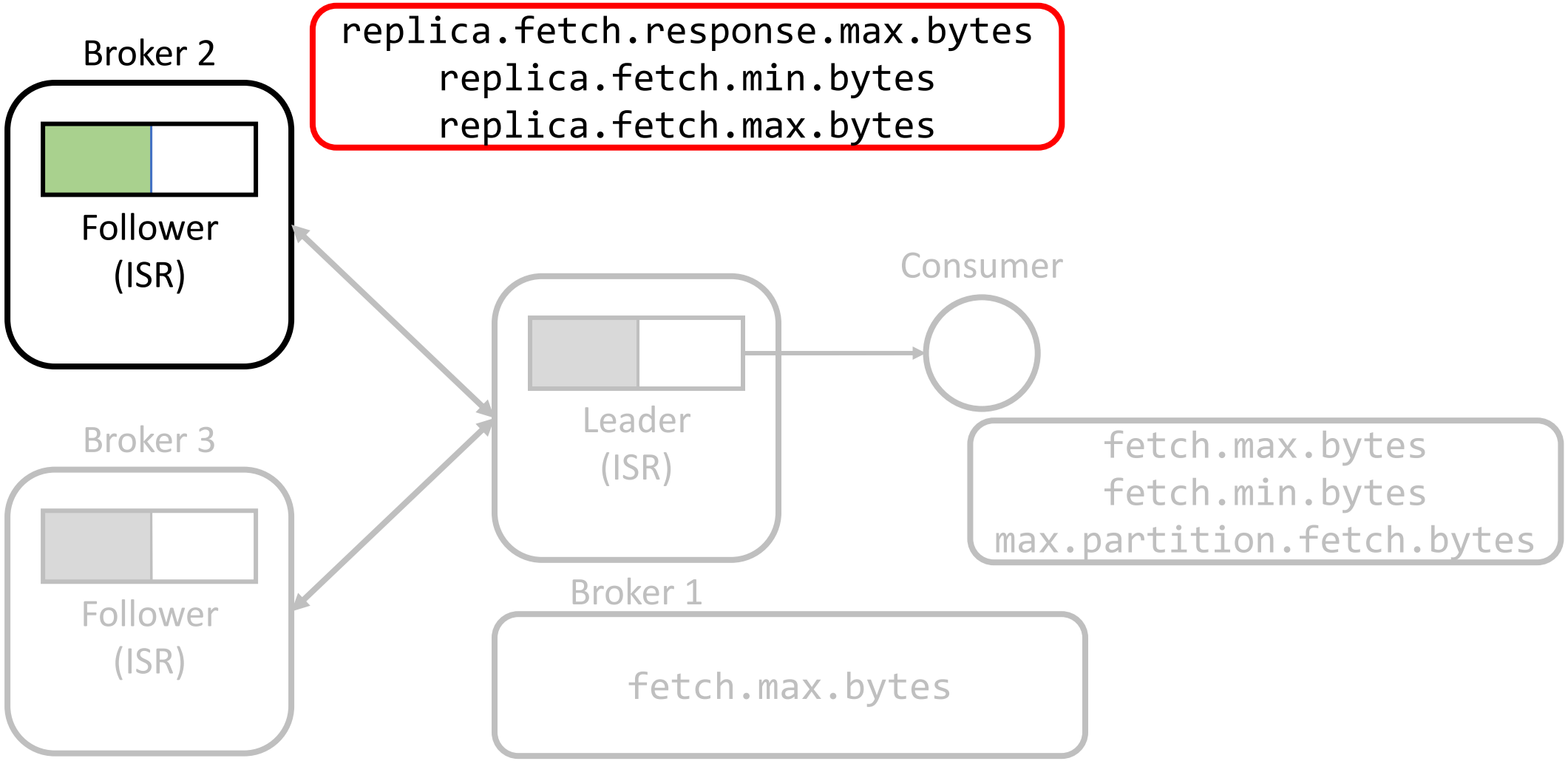


https://cwiki.apache.org/confluence/display/KAFKA/KIP-541%3A+Create+a+fetch.max.bytes+configuration+for+the+broker

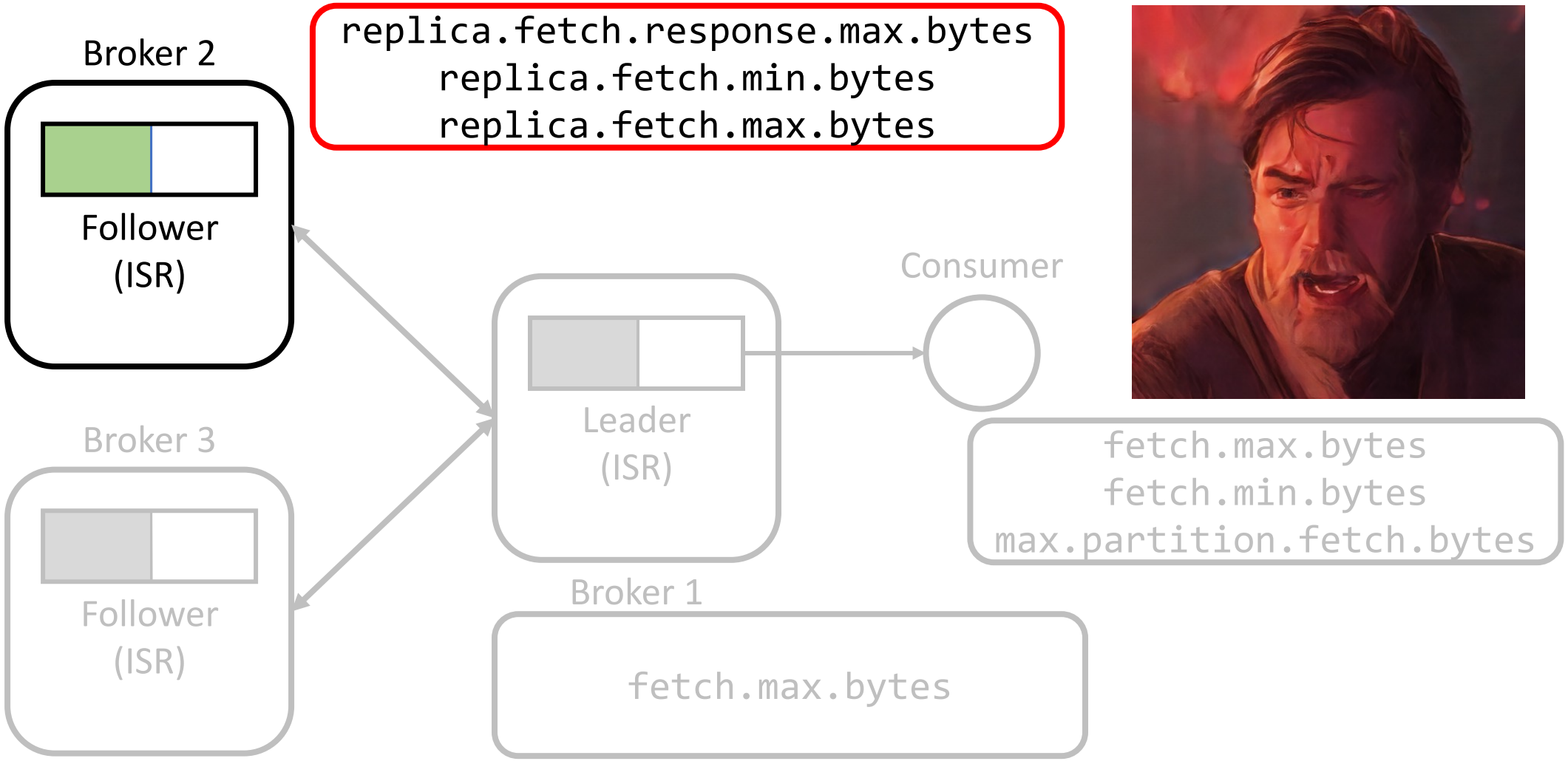
Репликация и чтение из Apache Kafka



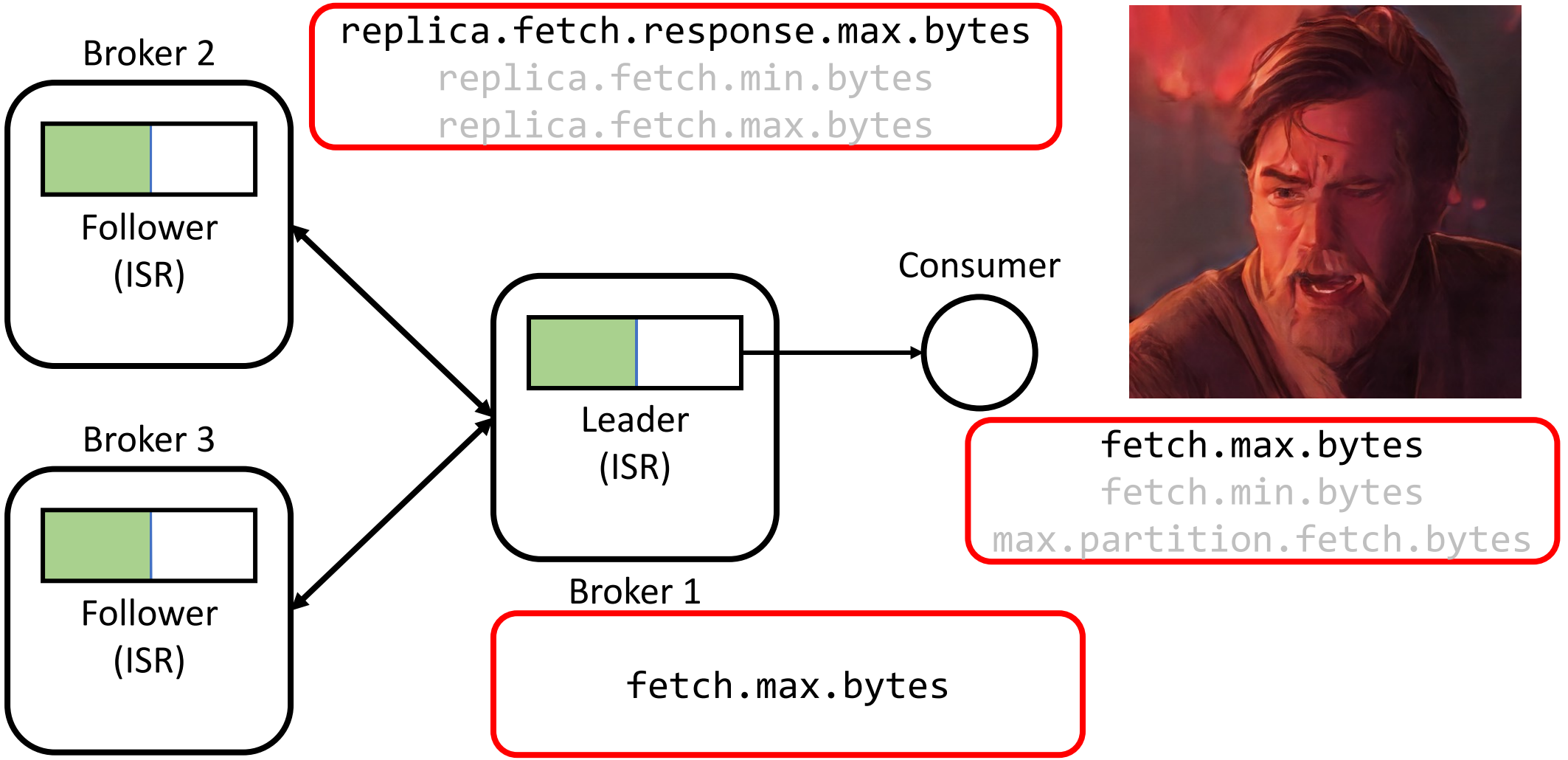
Репликация и чтение из Apache Kafka



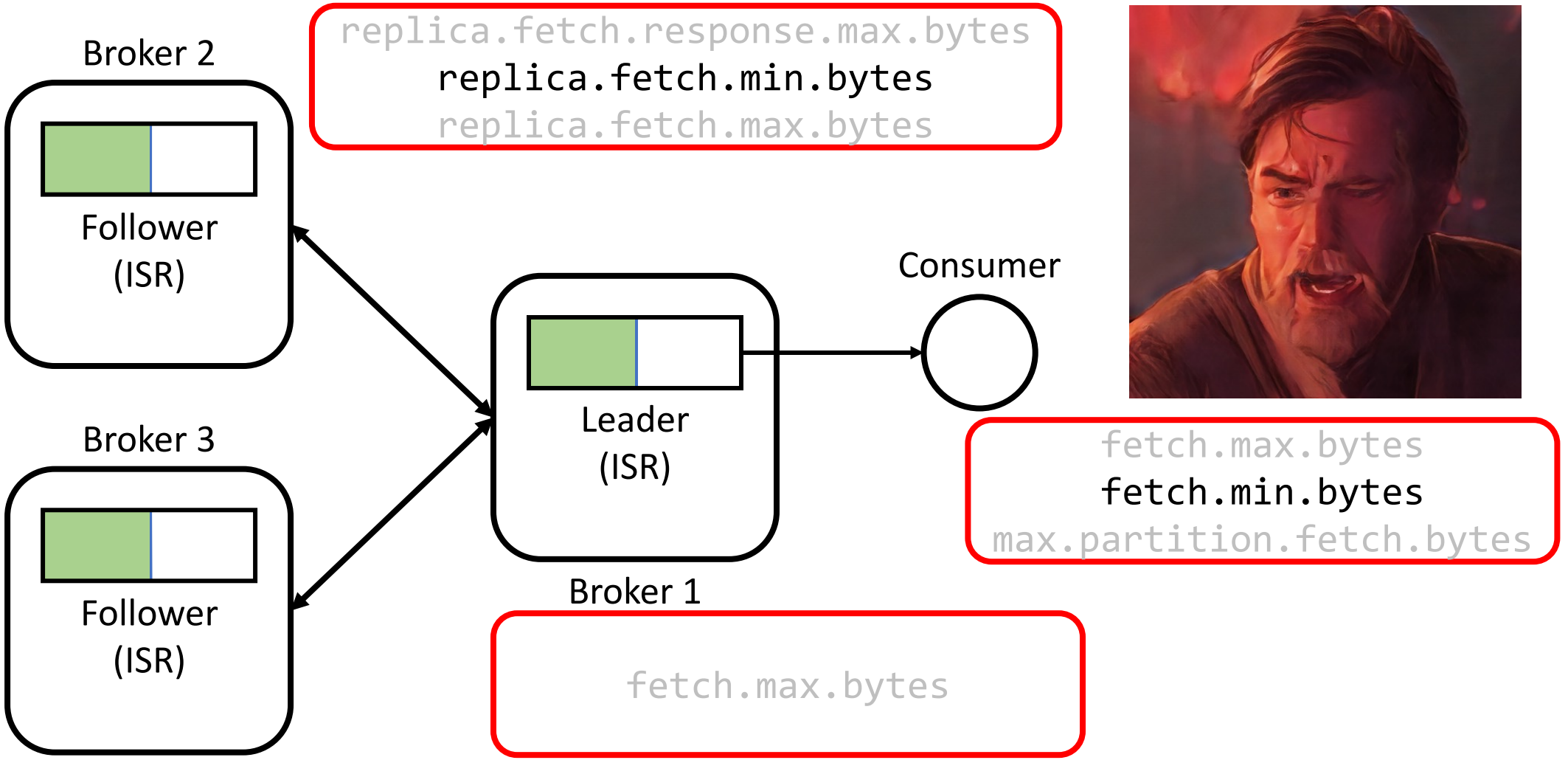
Репликация и чтение из Apache Kafka



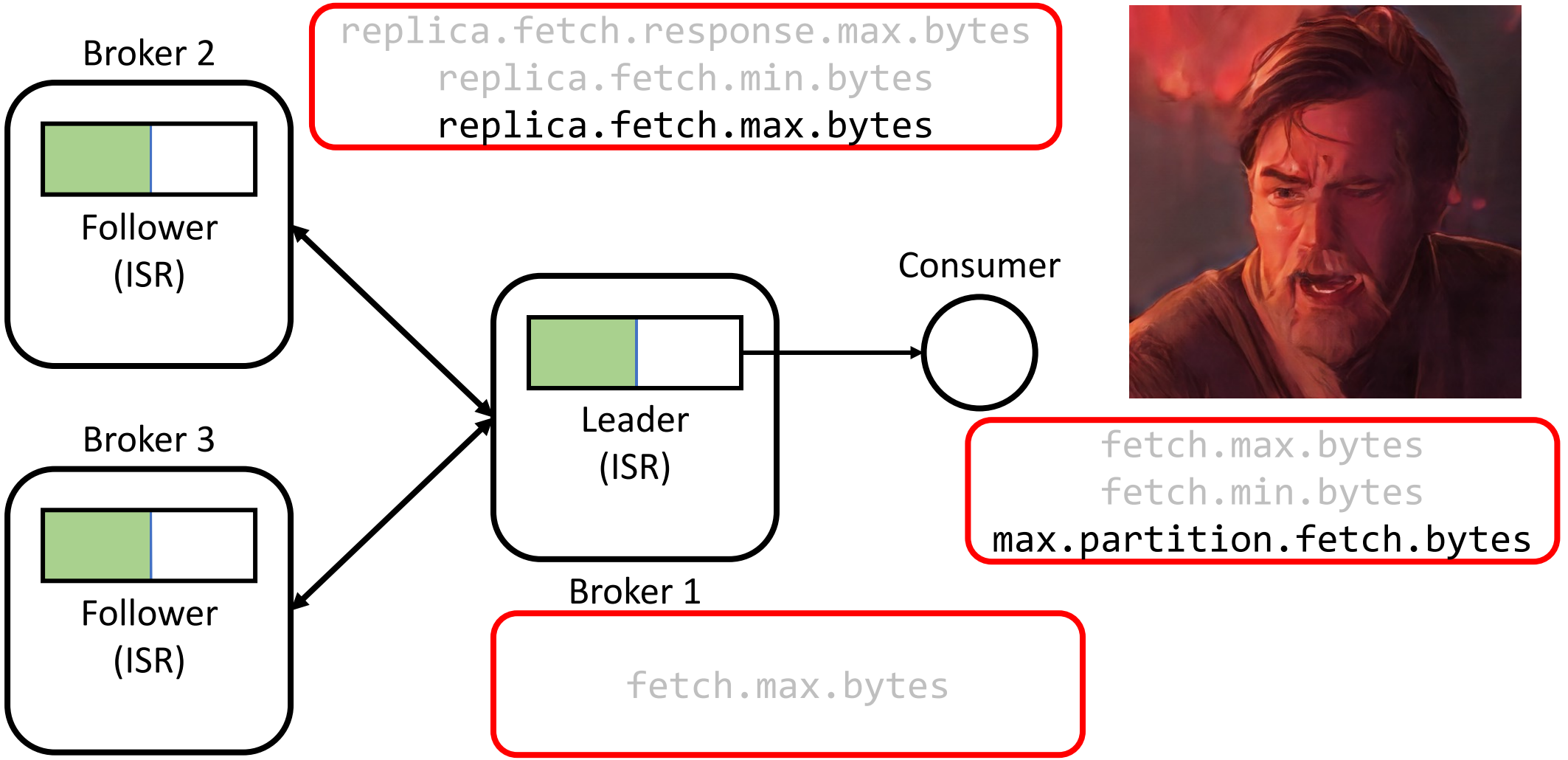
Репликация и чтение из Apache Kafka



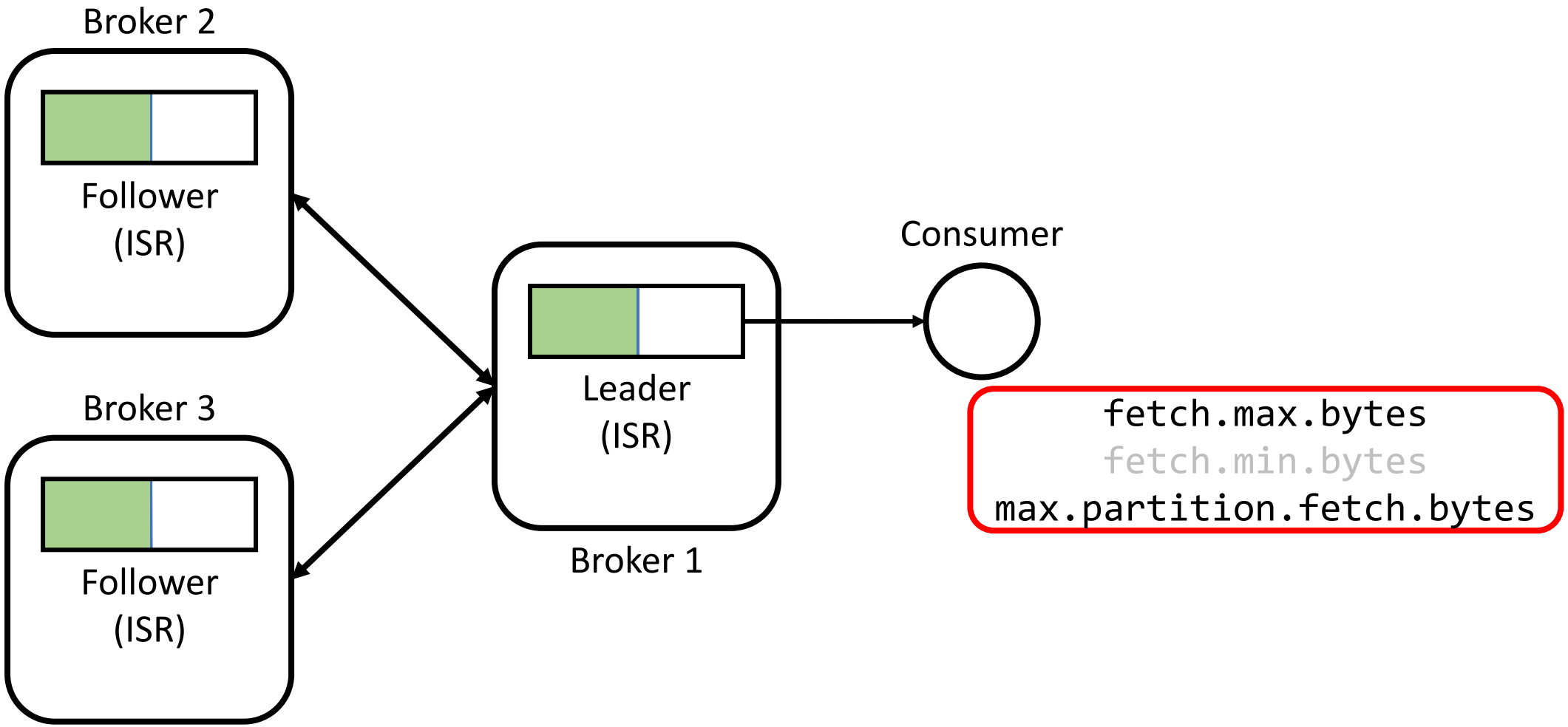
Репликация и чтение из Apache Kafka



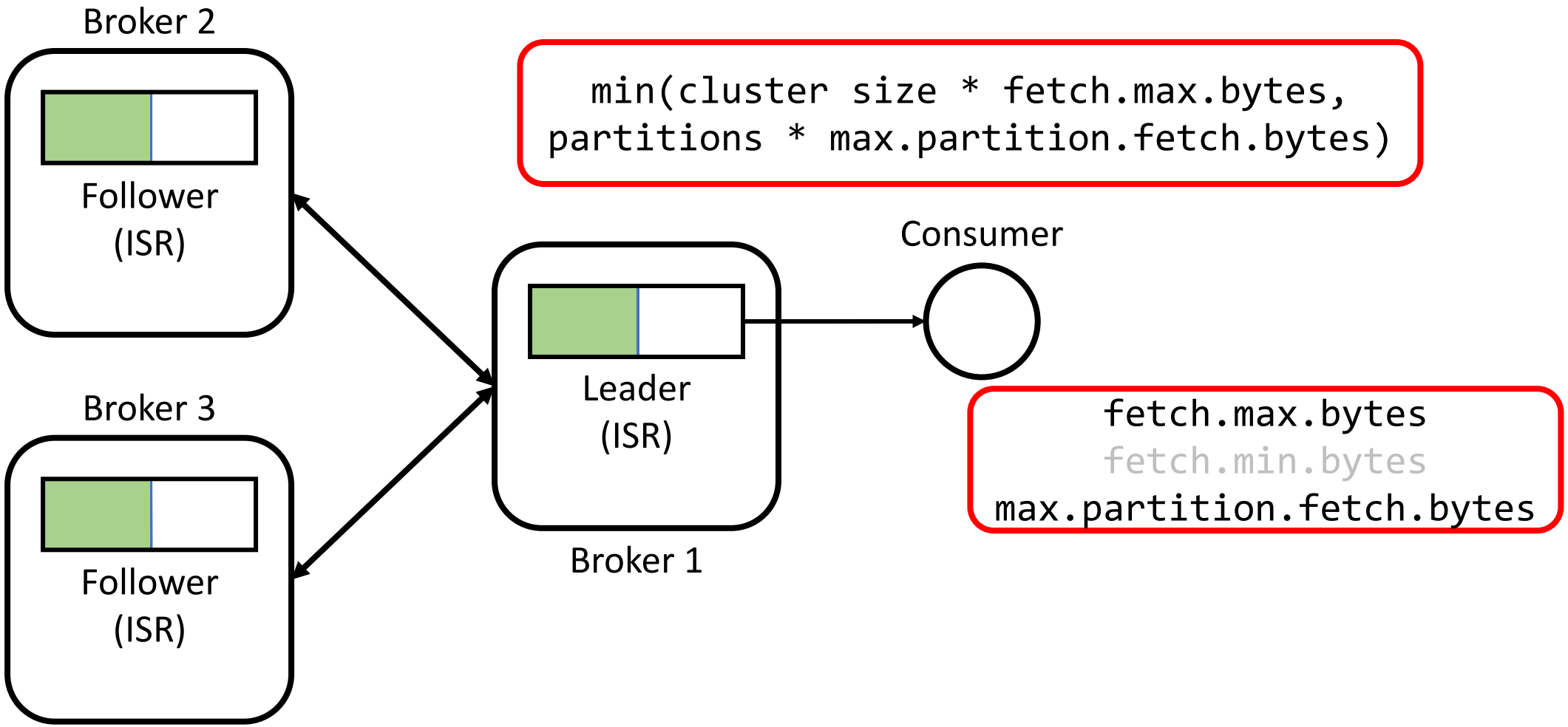
Репликация и чтение из Apache Kafka



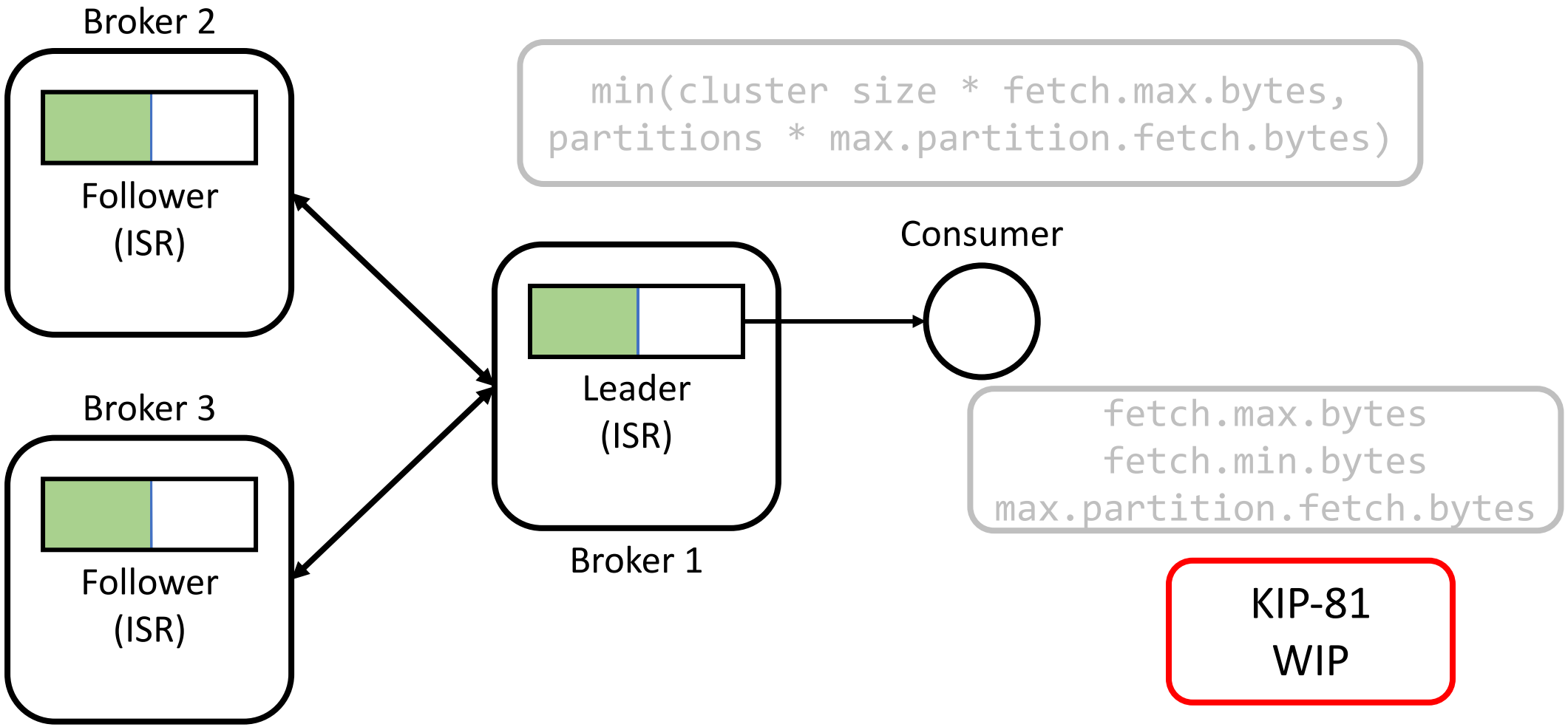
Репликация и чтение из Apache Kafka



Репликация и чтение из Apache Kafka

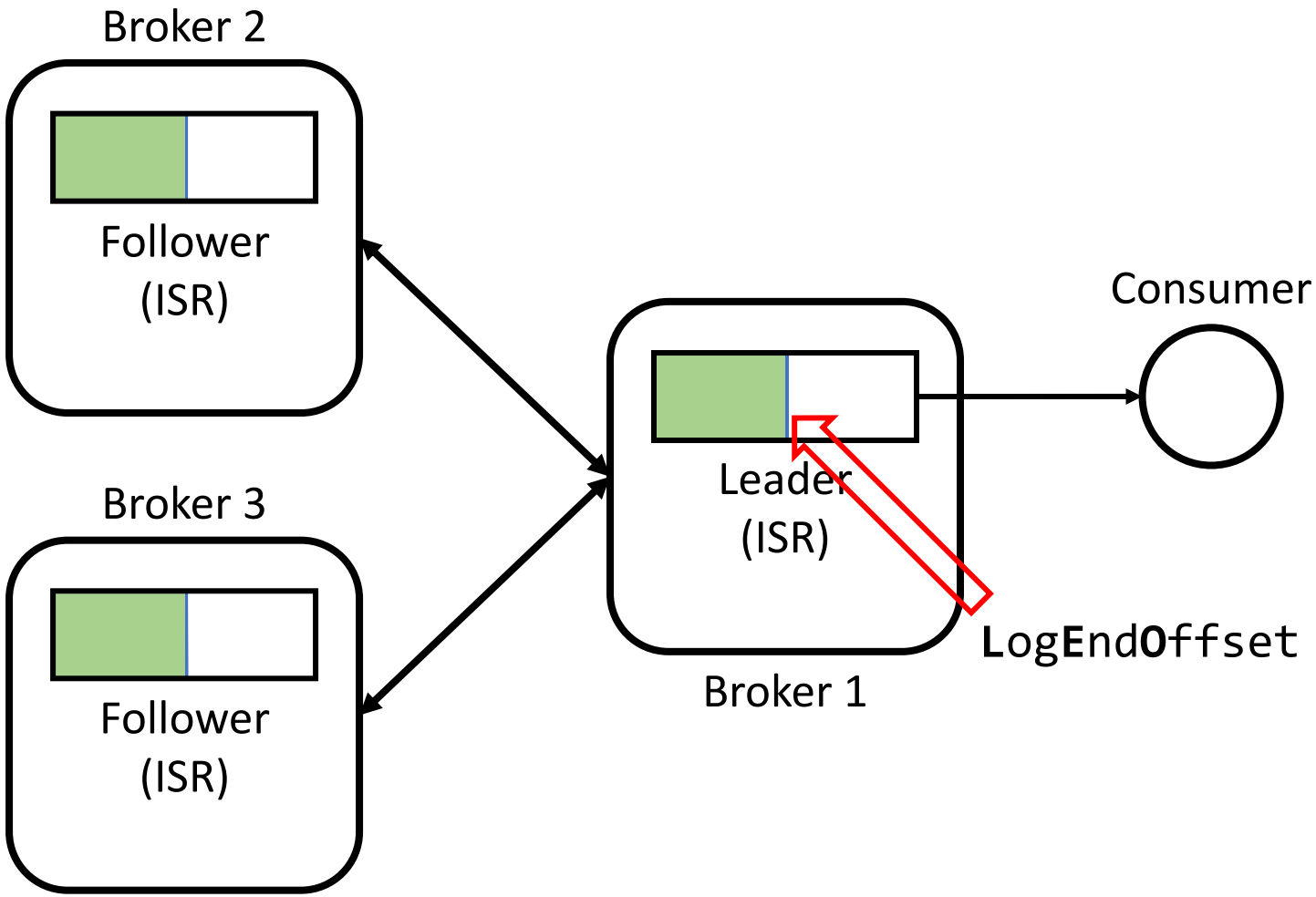


Репликация и чтение из Apache Kafka

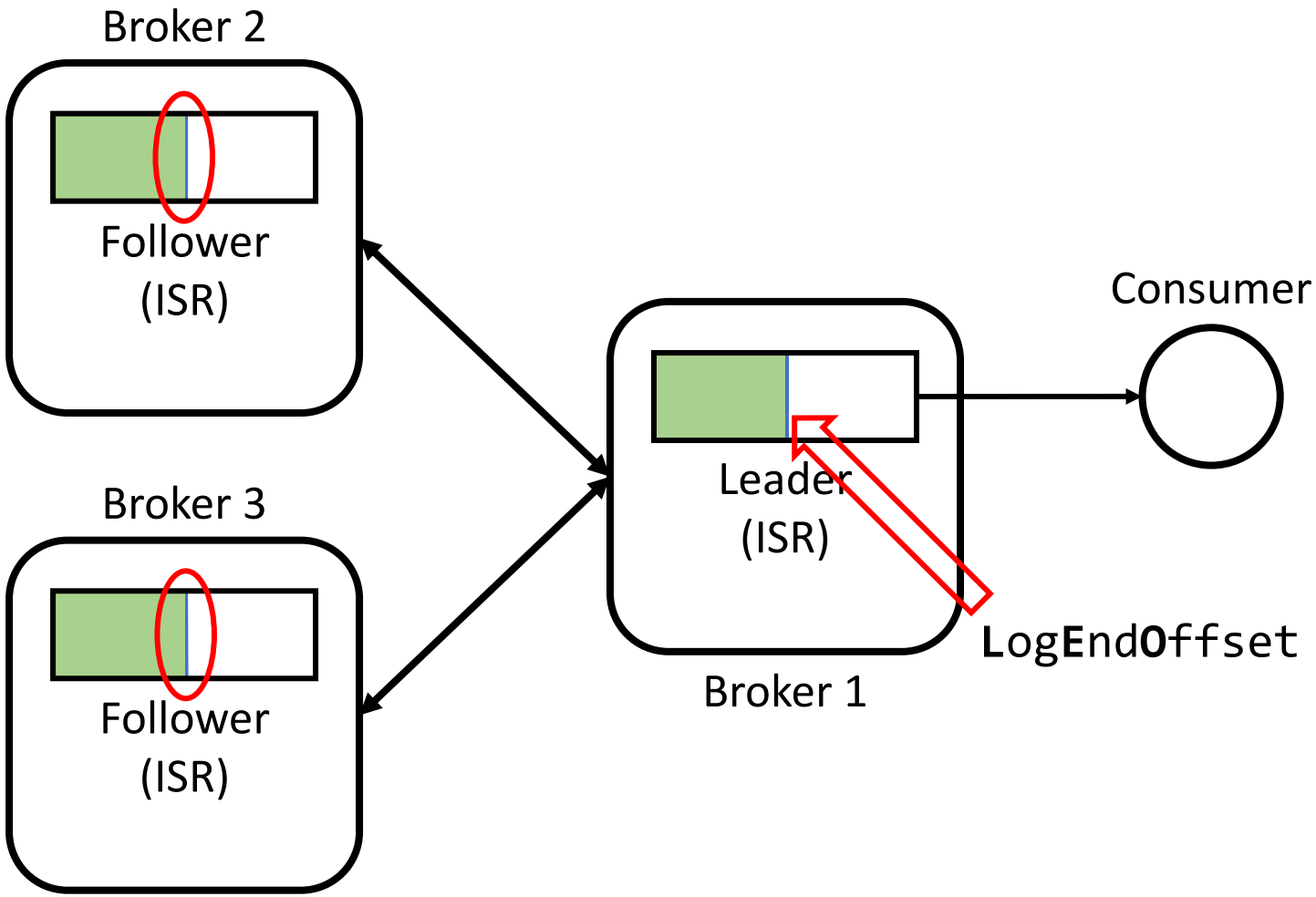


https://cwiki.apache.org/confluence/display/KAFKA/KIP-81%3A+Bound+Fetch+memory+usage+in+the+consumer

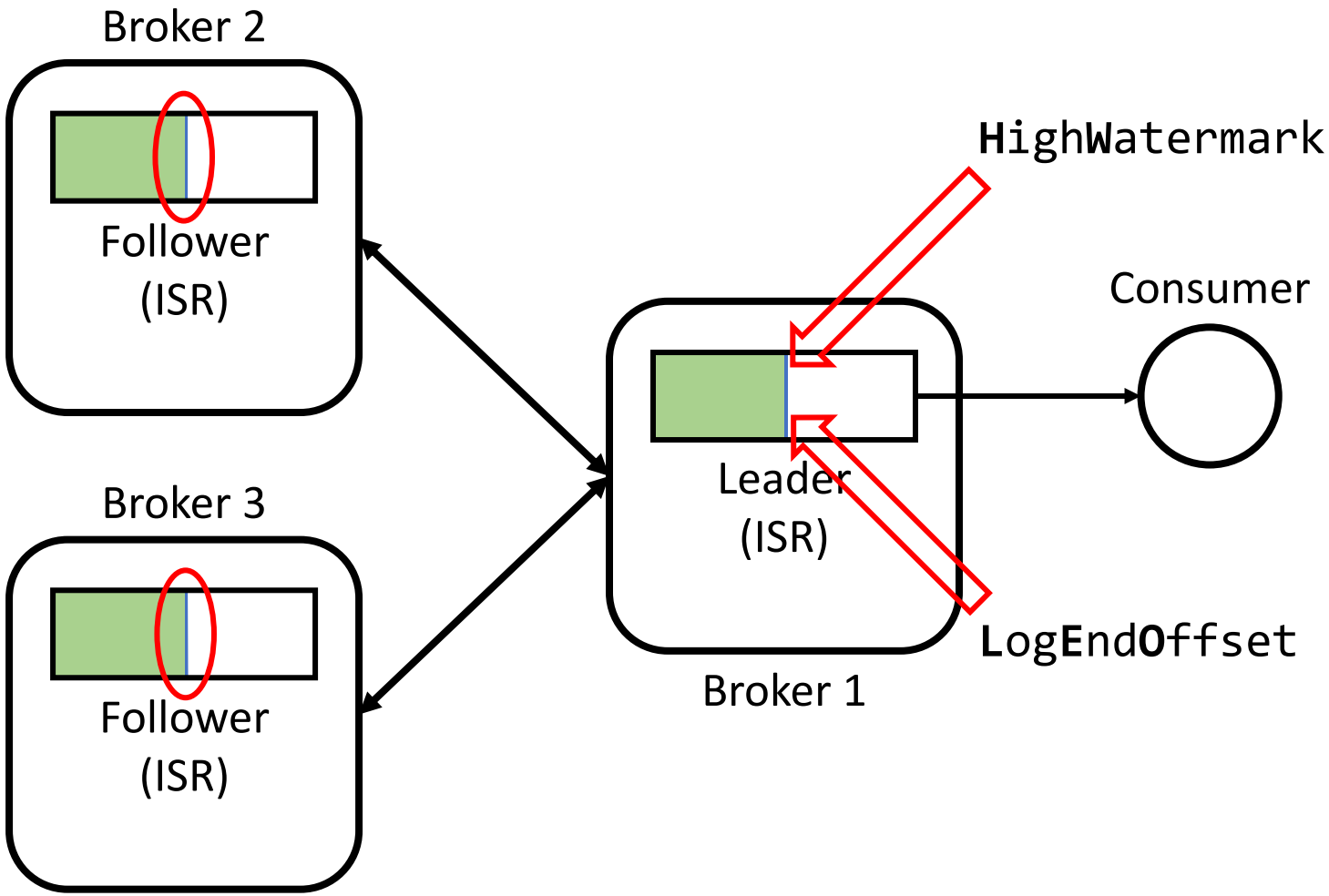
Репликация и чтение из Apache Kafka



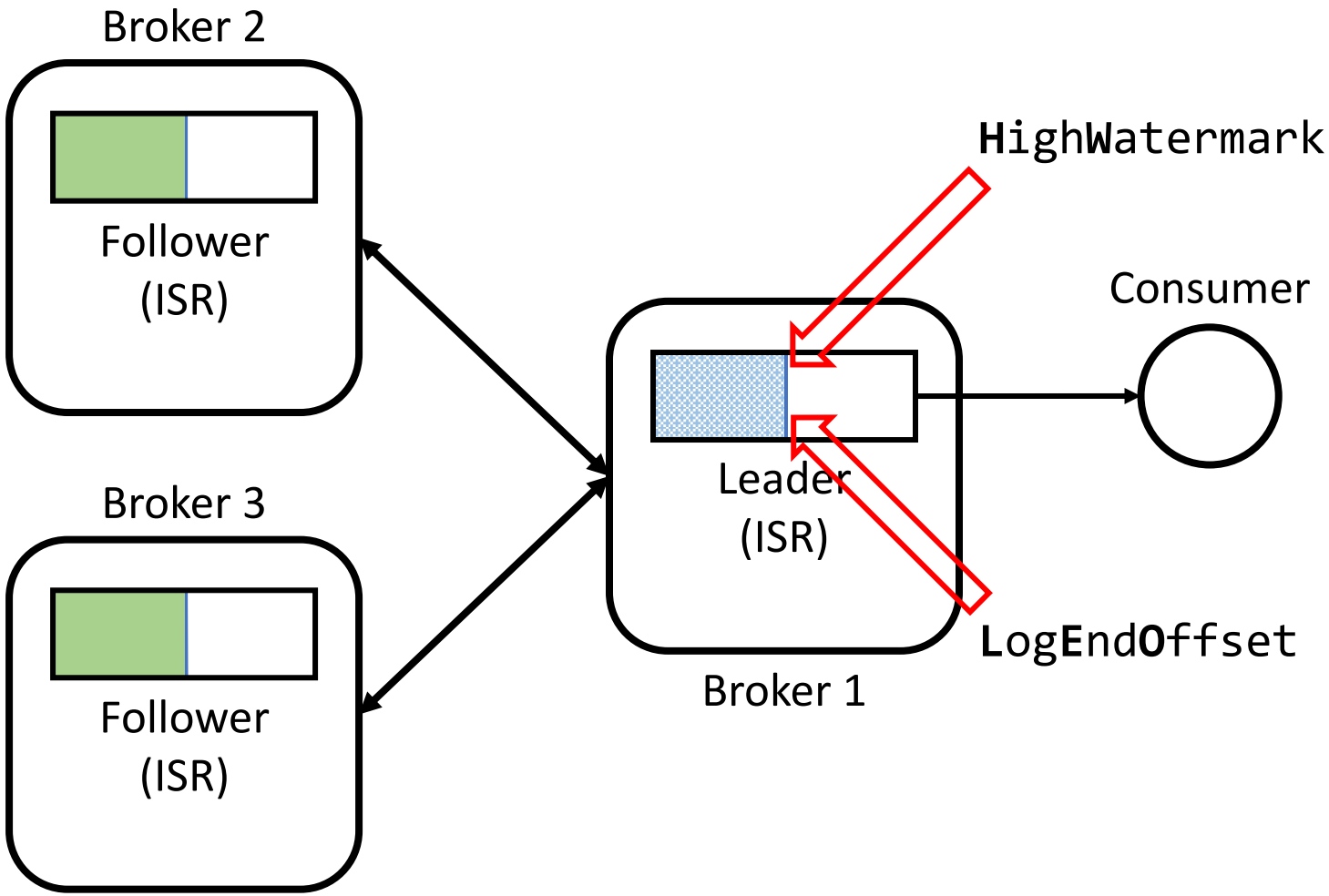
Репликация и чтение из Apache Kafka



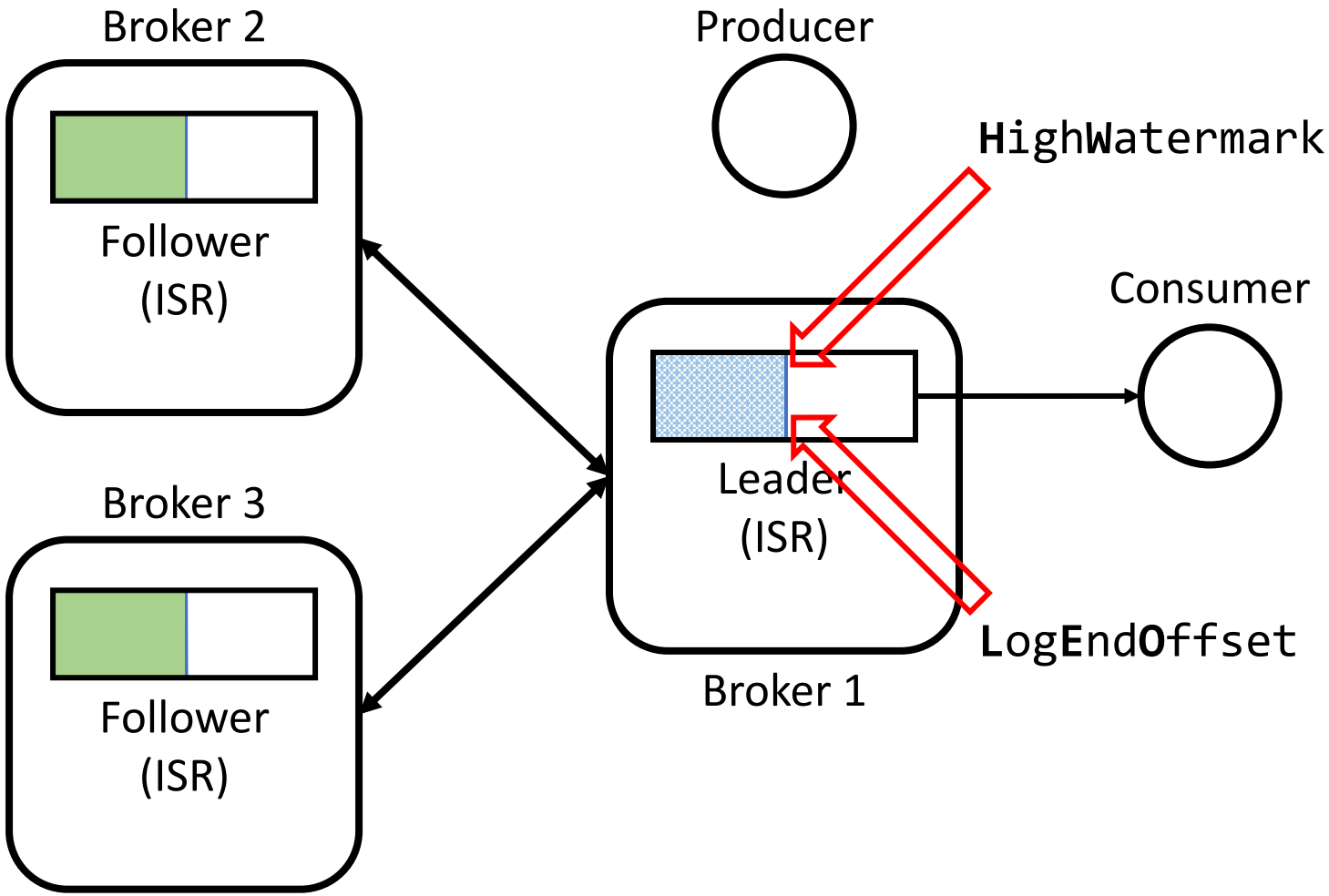
Репликация и чтение из Apache Kafka



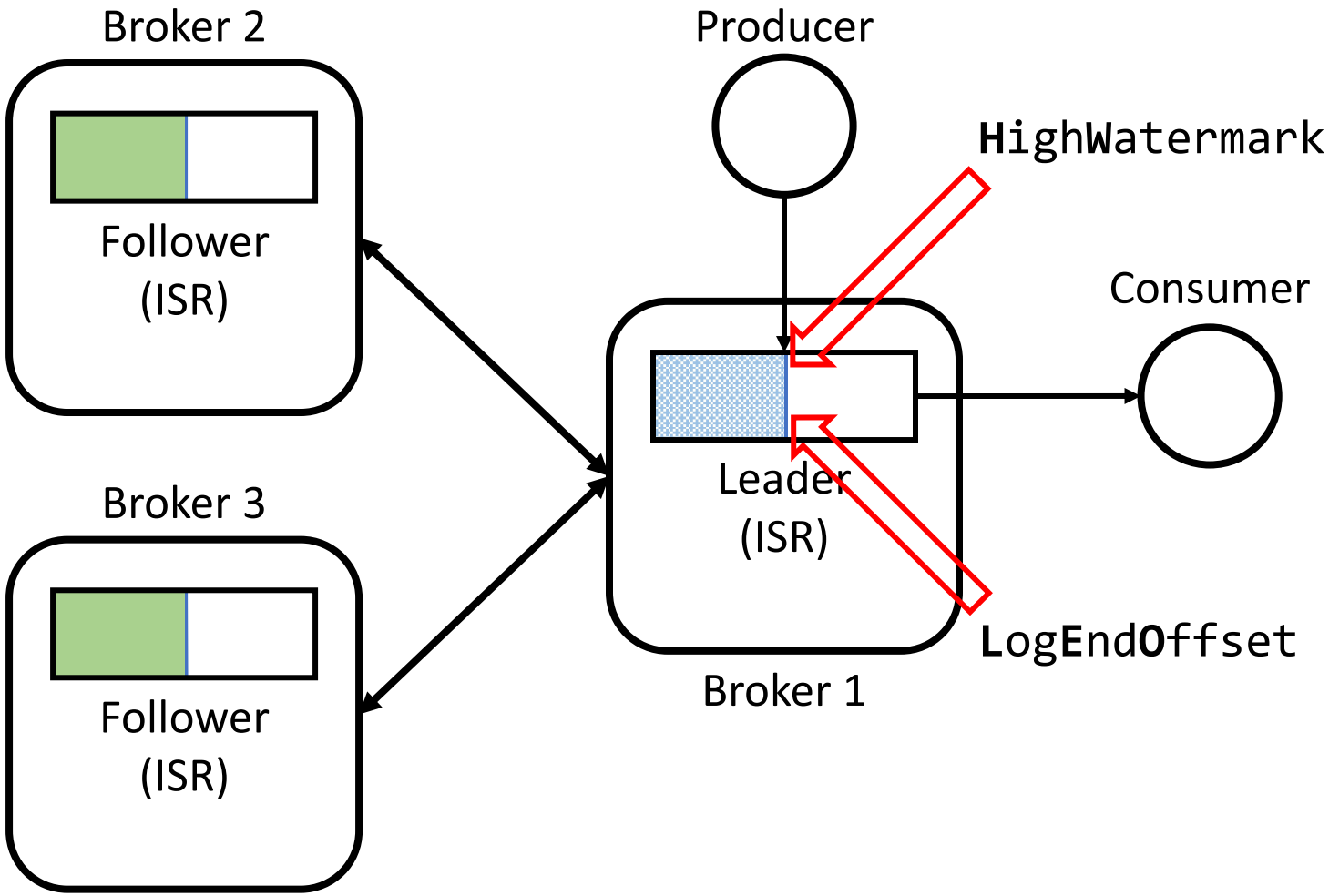
Репликация и чтение из Apache Kafka



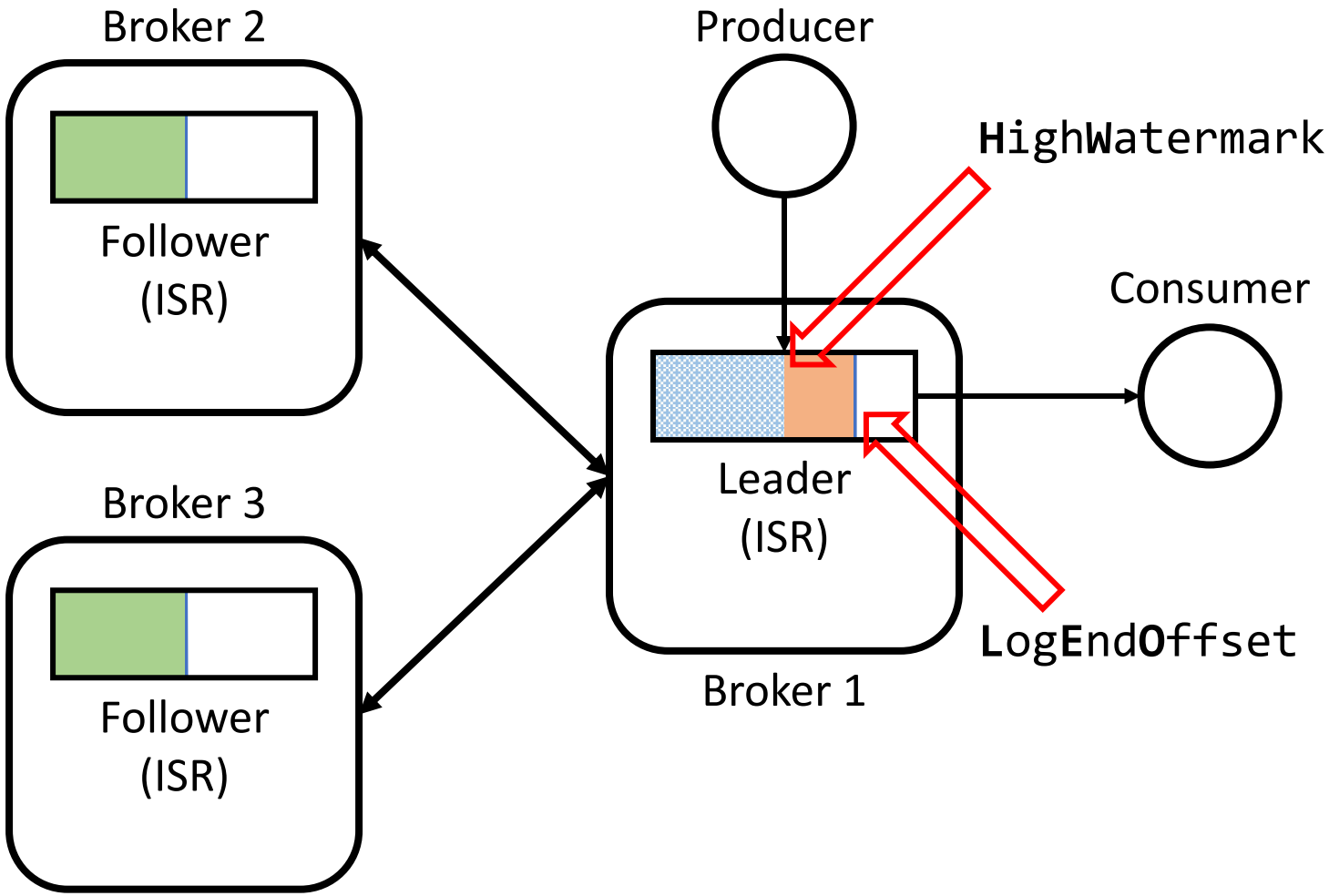
Репликация и чтение из Apache Kafka



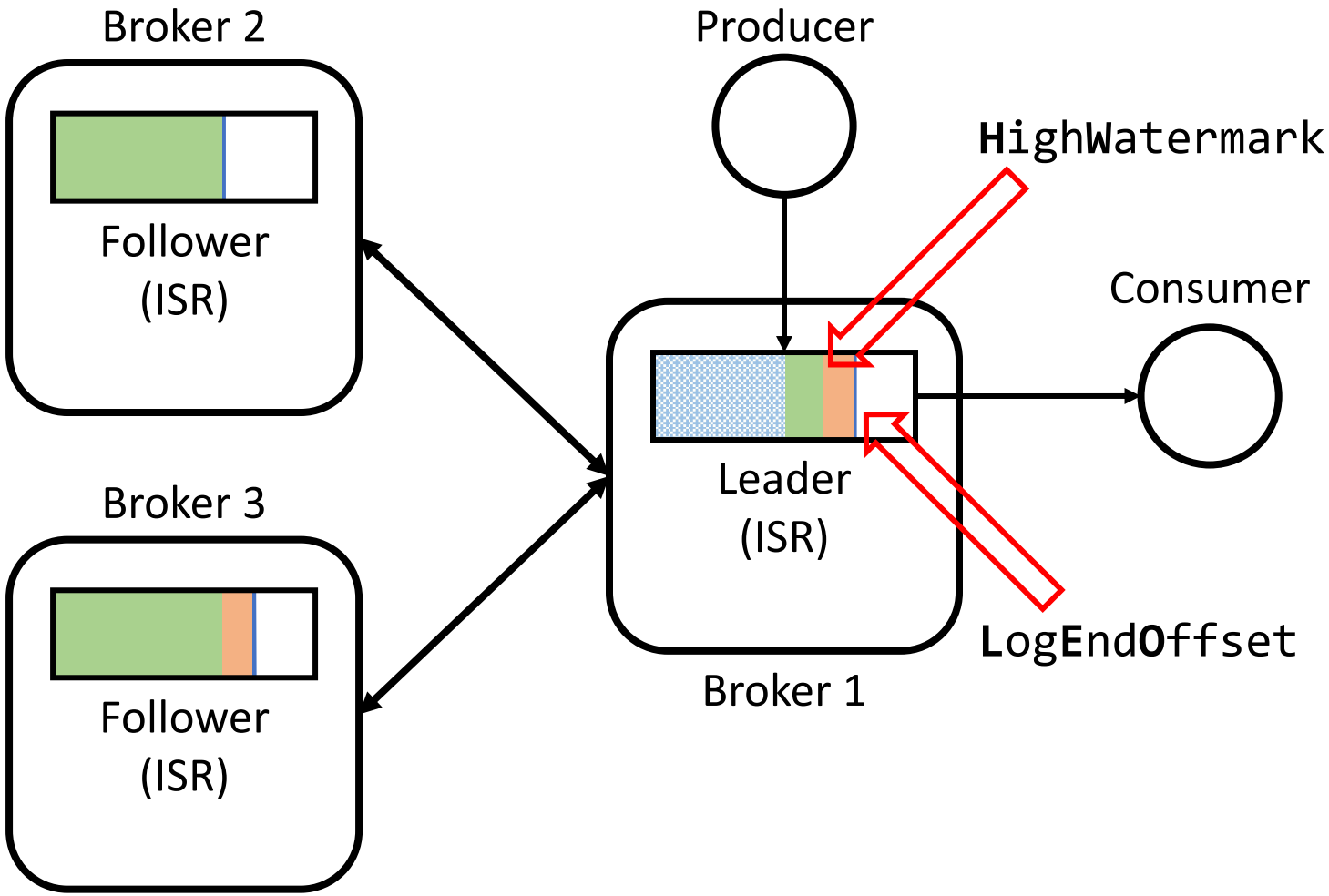
Репликация и чтение из Apache Kafka



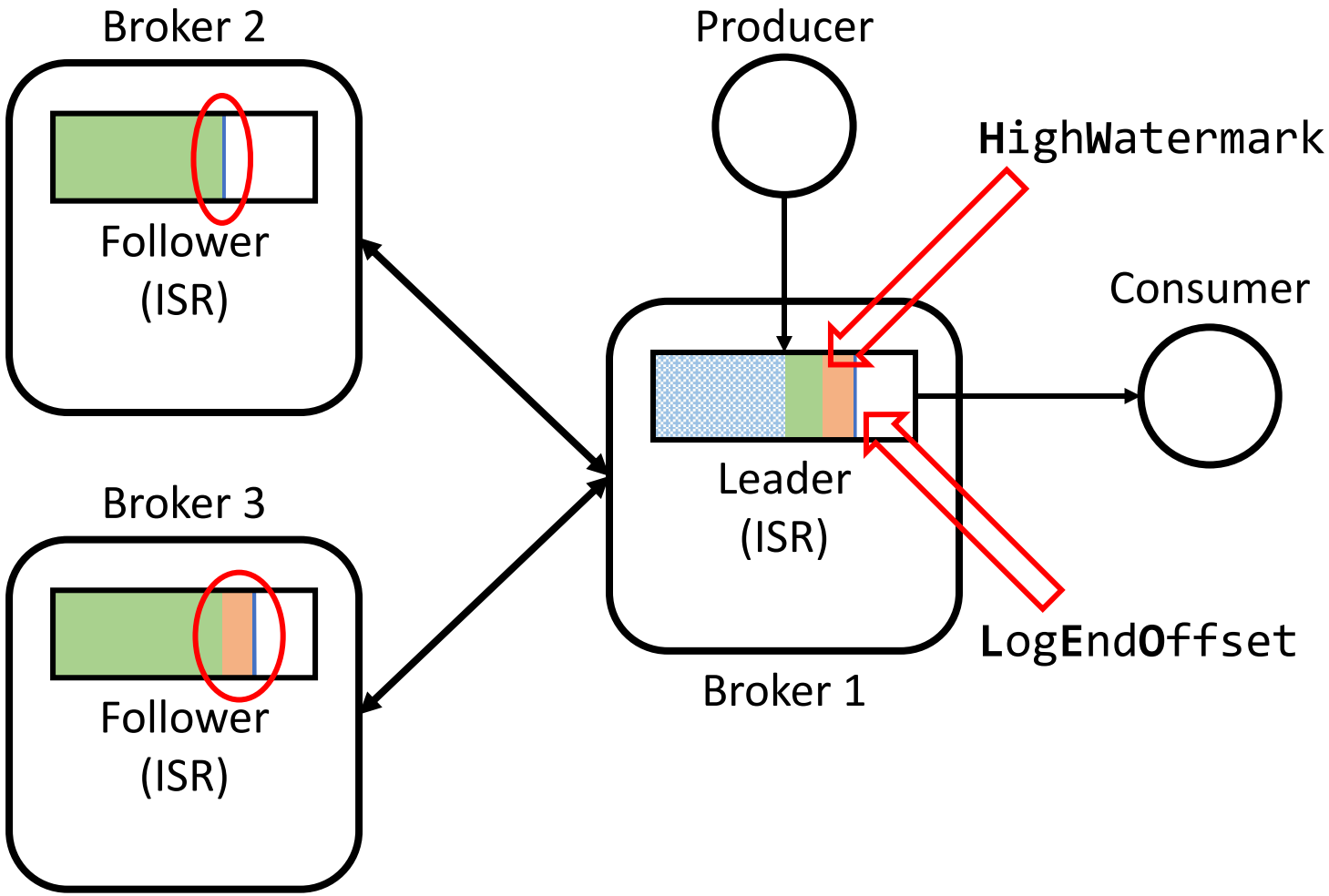
Репликация и чтение из Apache Kafka



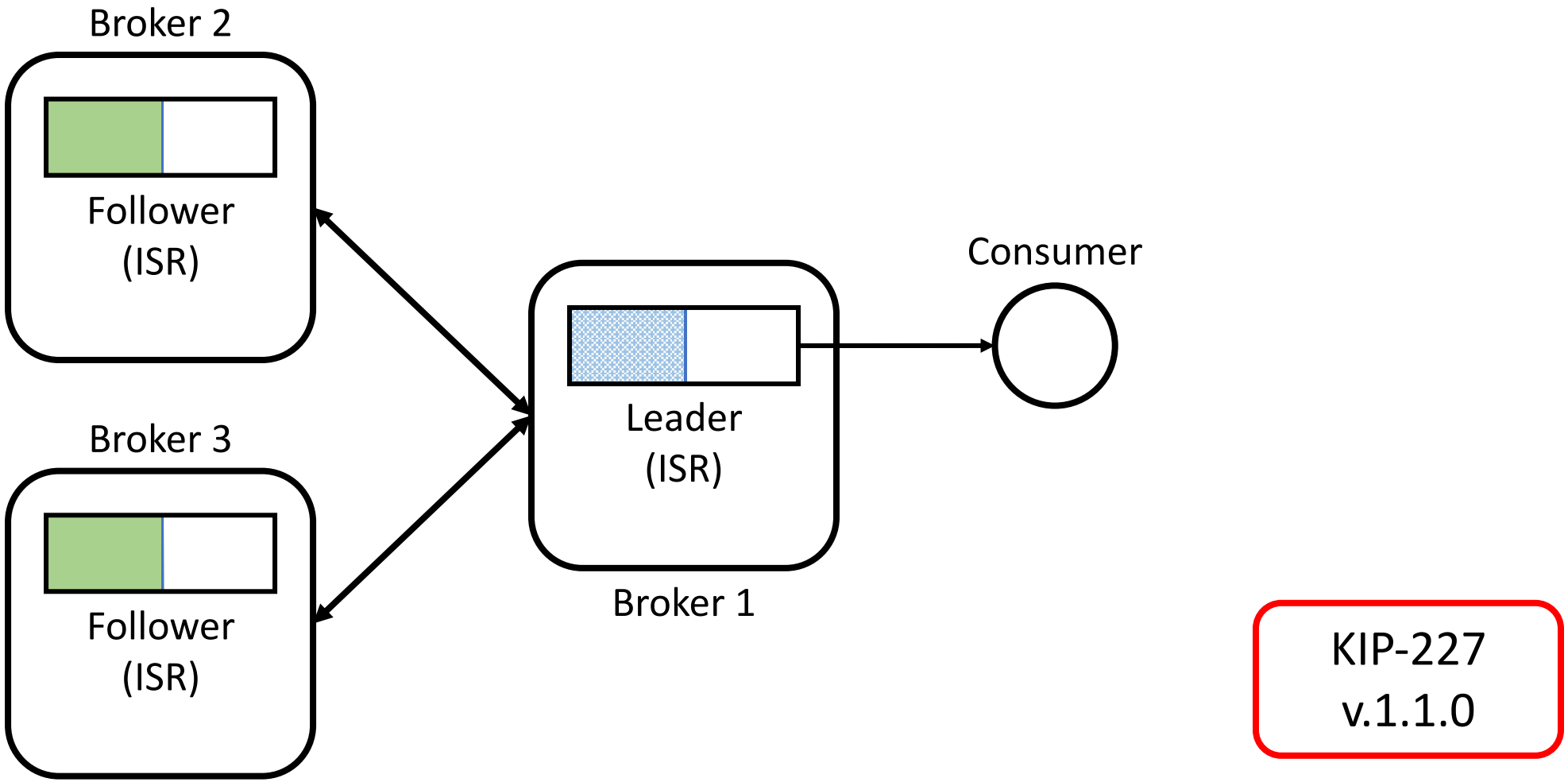
Репликация и чтение из Apache Kafka



Репликация и чтение из Apache Kafka



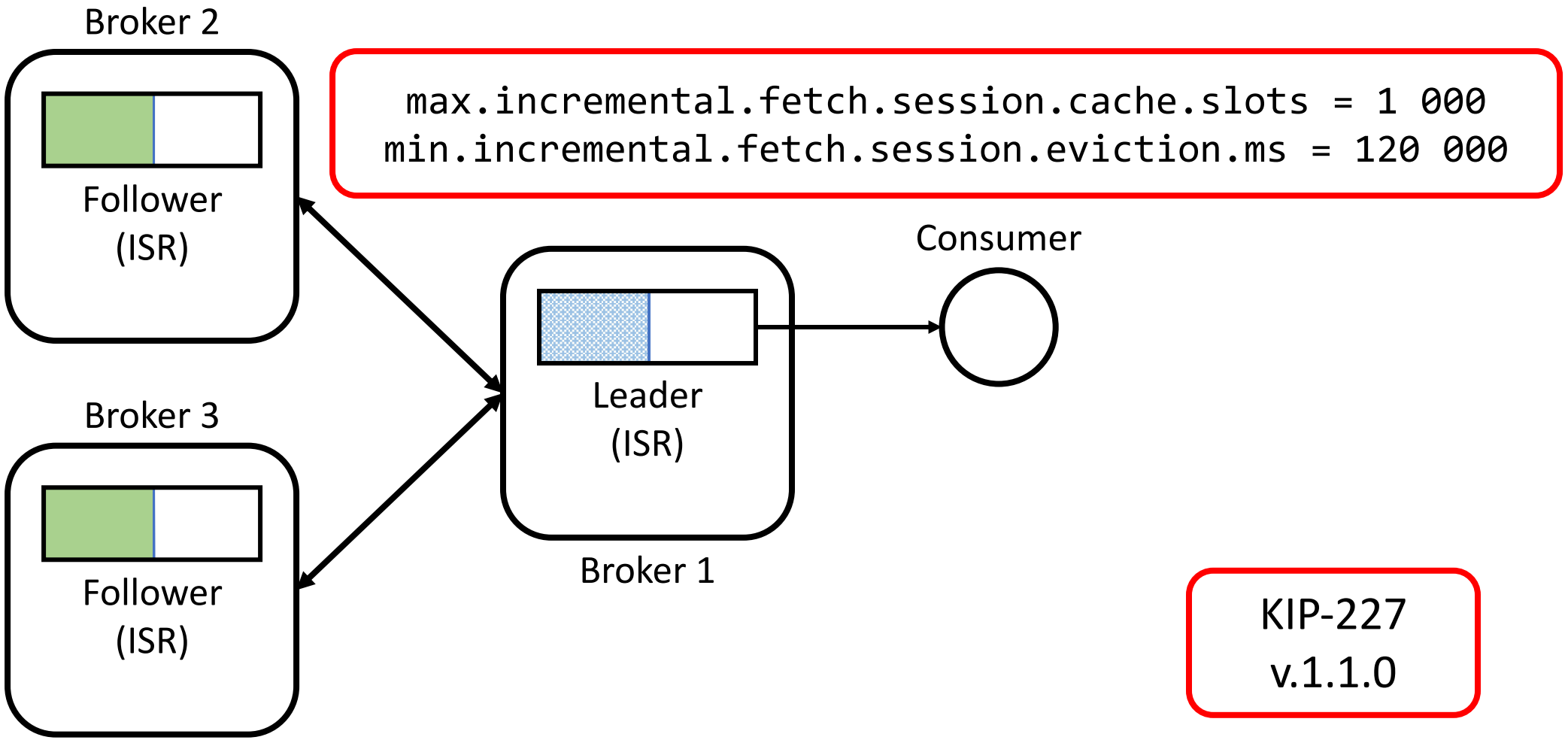
Репликация и чтение из Apache Kafka



KIP-227
v.1.1.0

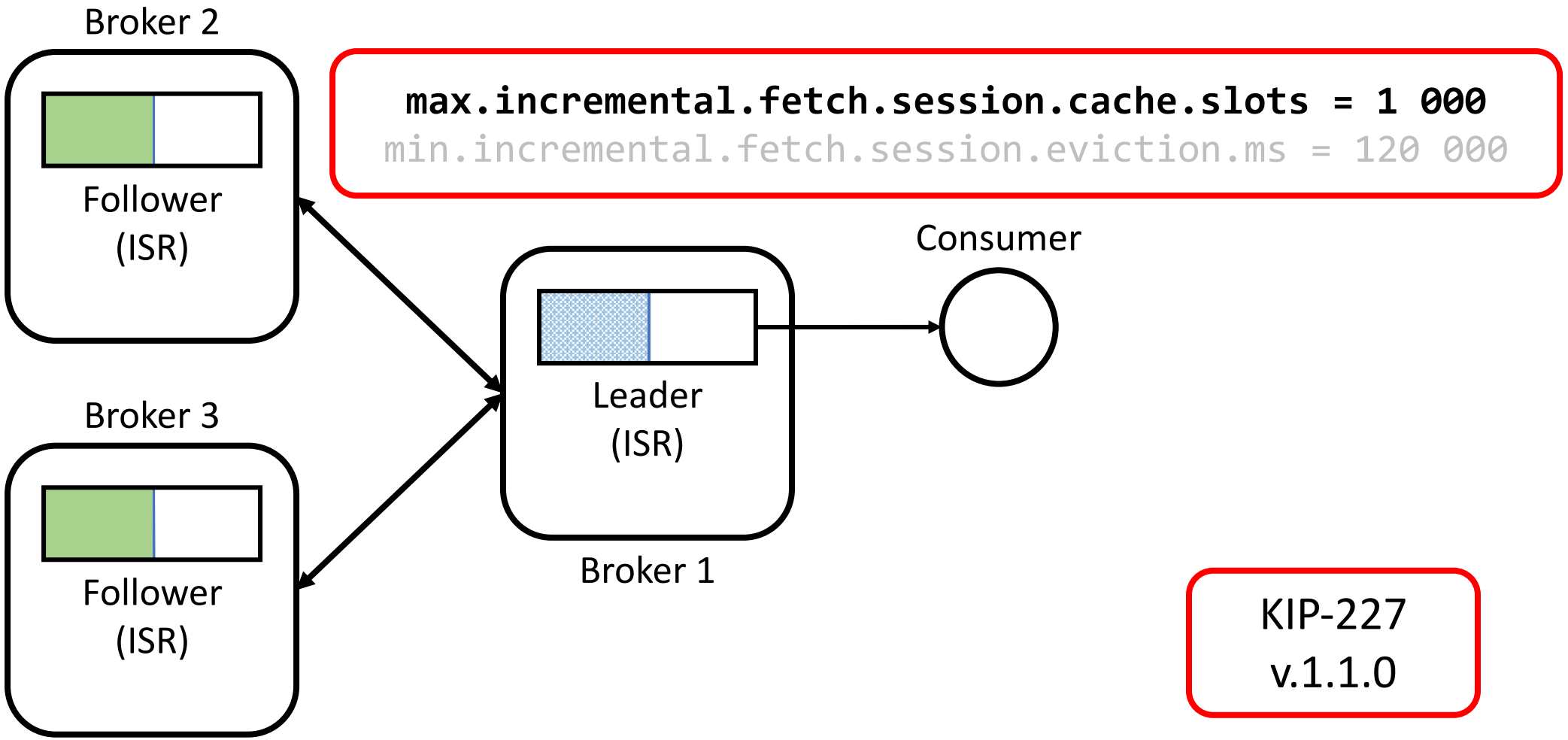
https://cwiki.apache.org/confluence/display/KAFKA/KIP-227%3A+Introduce+Incremental+FetchRequests+to+Increase+Partition+Scalability

Репликация и чтение из Apache Kafka



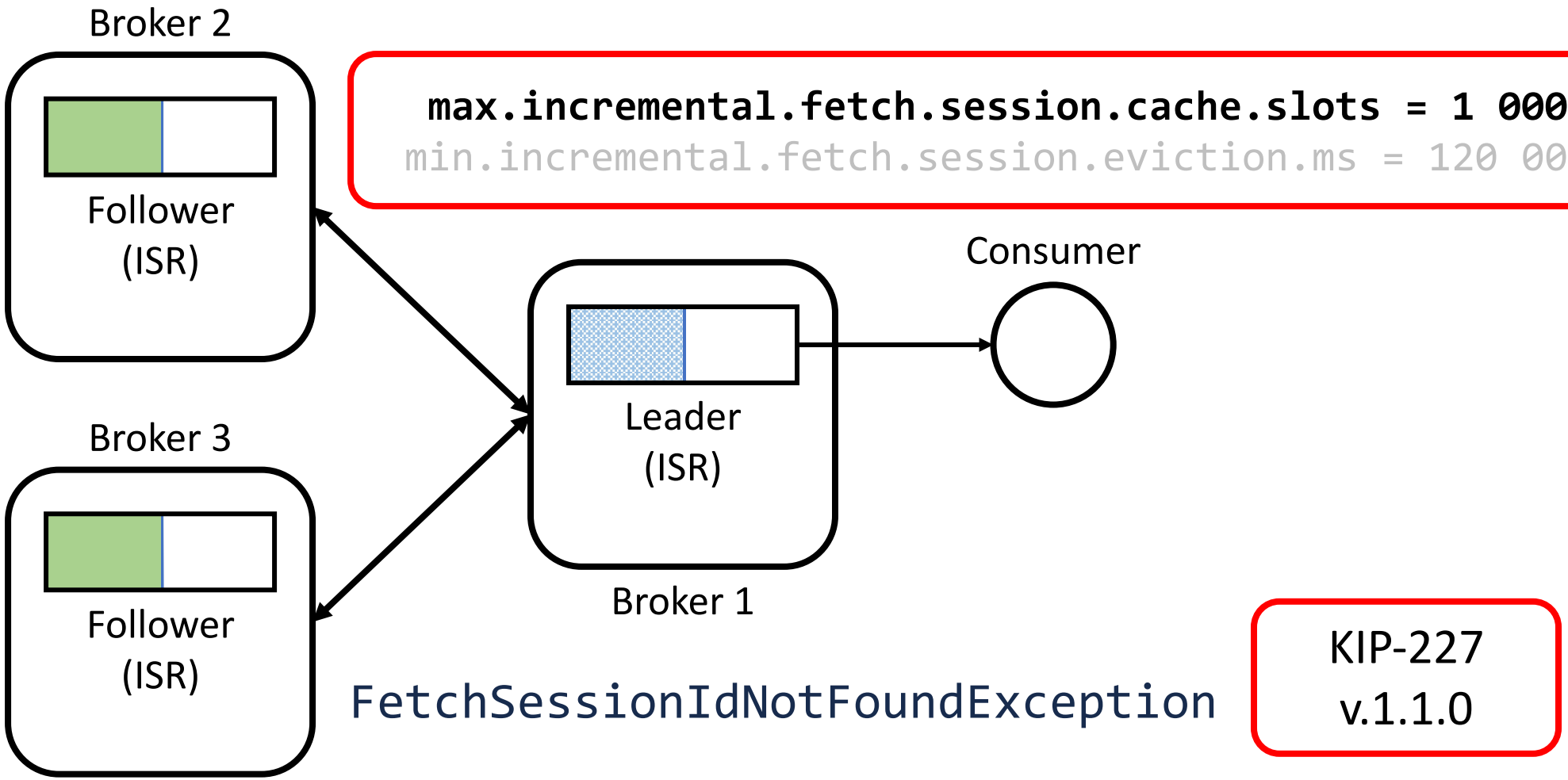
https://cwiki.apache.org/confluence/display/KAFKA/KIP-227%3A+Introduce+Incremental+FetchRequests+to+Increase+Partition+Scalability

Репликация и чтение из Apache Kafka



https://cwiki.apache.org/confluence/display/KAFKA/KIP-227%3A+Introduce+Incremental+FetchRequests+to+Increase+Partition+Scalability

Репликация и чтение из Apache Kafka



https://cwiki.apache.org/confluence/display/KAFKA/KIP-227%3A+Introduce+Incremental+FetchRequests+to+Increase+Partition+Scalability

Consumer под капотом

Consumer под капотом

— `KafkaConsumer` // consumer

Consumer под капотом

— `KafkaConsumer` // consumer

— `ConsumerMetadata` // metadata

Consumer под капотом

— `KafkaConsumer` // consumer
— `ConsumerMetadata` // metadata
— `SubscriptionState` // subscriptions

Consumer под капотом

— `KafkaConsumer` // consumer
— `ConsumerMetadata` // metadata
— `SubscriptionState` // subscriptions
— `ConsumerNetworkClient` // client

Consumer под капотом

- `KafkaConsumer` // consumer
- `ConsumerMetadata` // metadata
- `SubscriptionState` // subscriptions
- `ConsumerNetworkClient` // client
- `NetworkClient` // network client

Consumer под капотом

```
— KafkaConsumer                // consumer
— ConsumerMetadata             // metadata
— SubscriptionState            // subscriptions
— ConsumerNetworkClient        // client
— NetworkClient                // network client
— Fetcher                      // fetcher
```

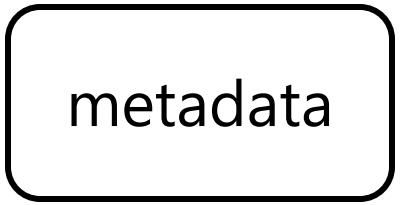
Consumer под капотом

```
— KafkaConsumer                // consumer
— ConsumerMetadata             // metadata
— SubscriptionState            // subscriptions
— ConsumerNetworkClient        // client
— NetworkClient                // network client
— Fetcher                      // fetcher
— ConsumerCoordinator          // coordinator
```

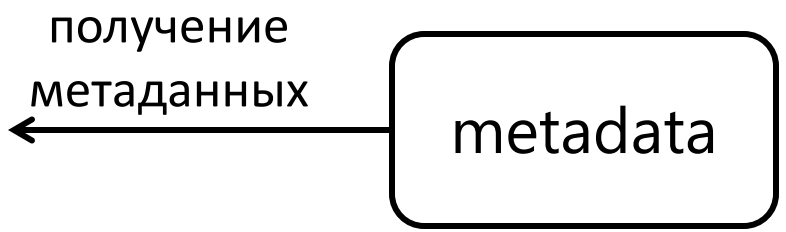
Consumer под капотом — `metadata`

```
— KafkaConsumer                // consumer
— ConsumerMetadata             // metadata
— SubscriptionState            // subscriptions
— ConsumerNetworkClient        // client
— NetworkClient                // network client
— Fetcher                      // fetcher
— ConsumerCoordinator          // coordinator
```

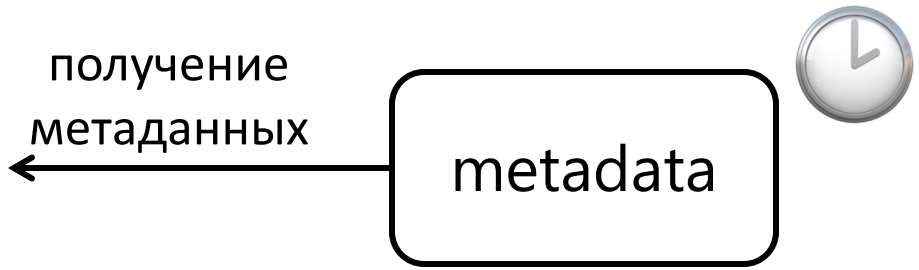
Consumer под капотом — `metadata`



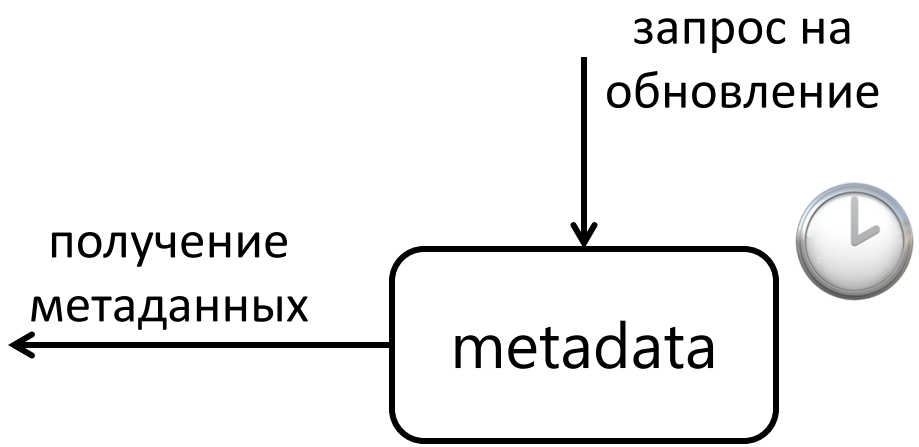
Consumer под капотом — `metadata`



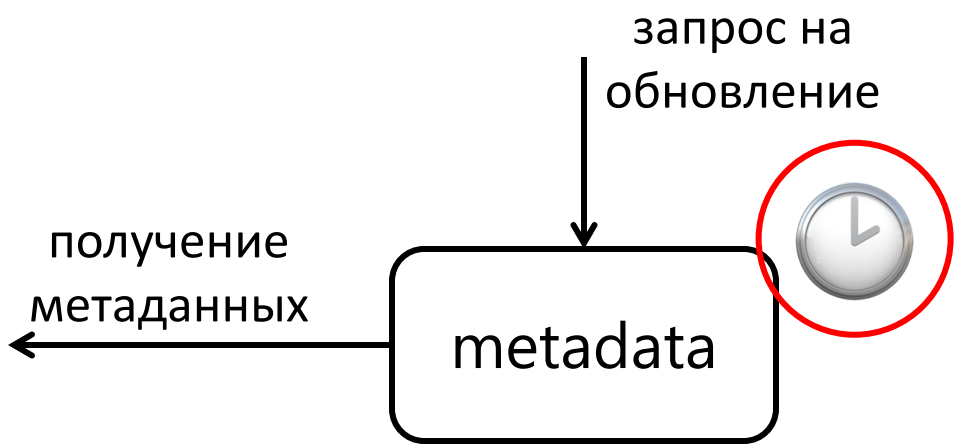
Consumer под капотом — metadata



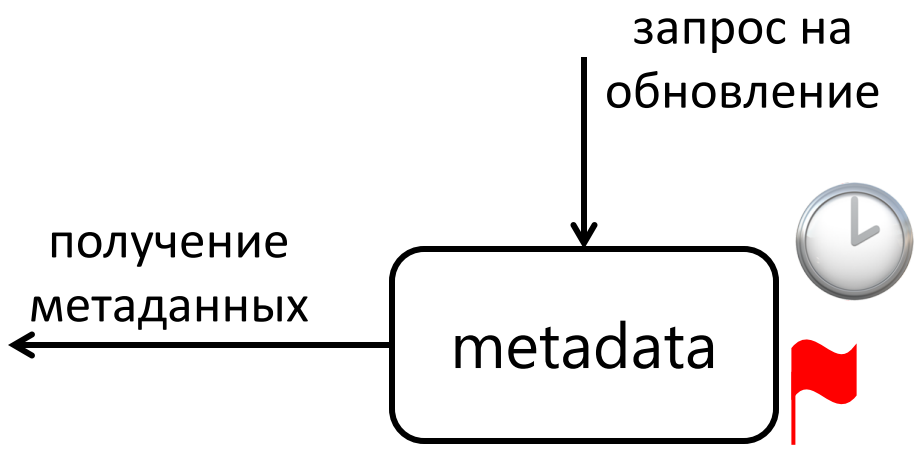
Consumer под капотом — metadata



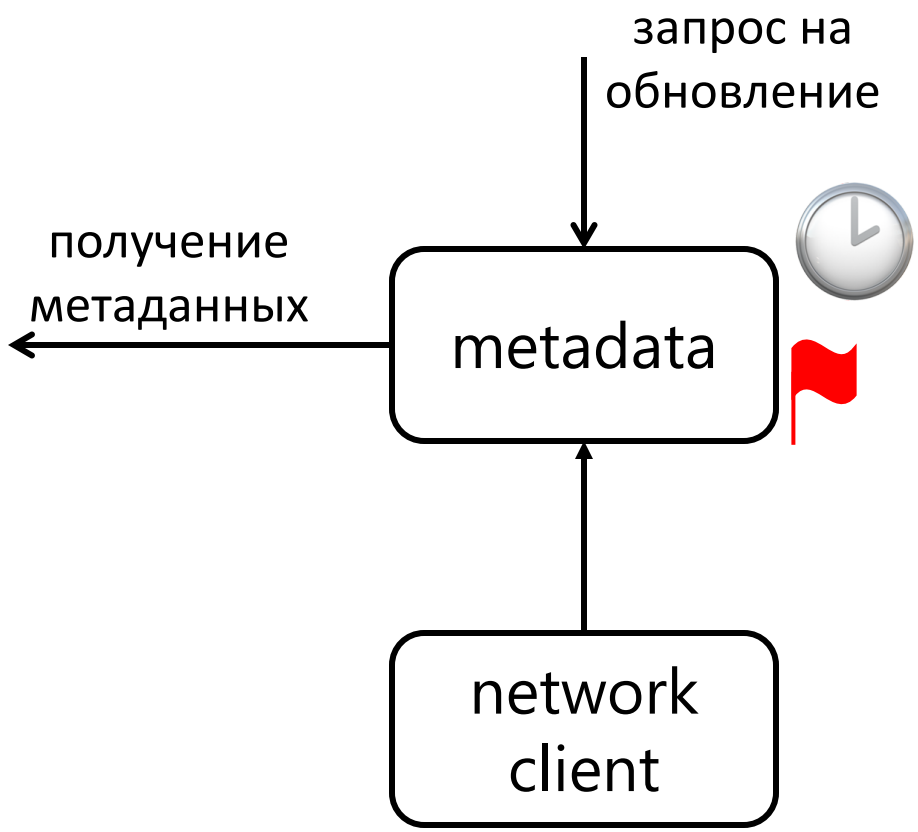
Consumer под капотом — metadata



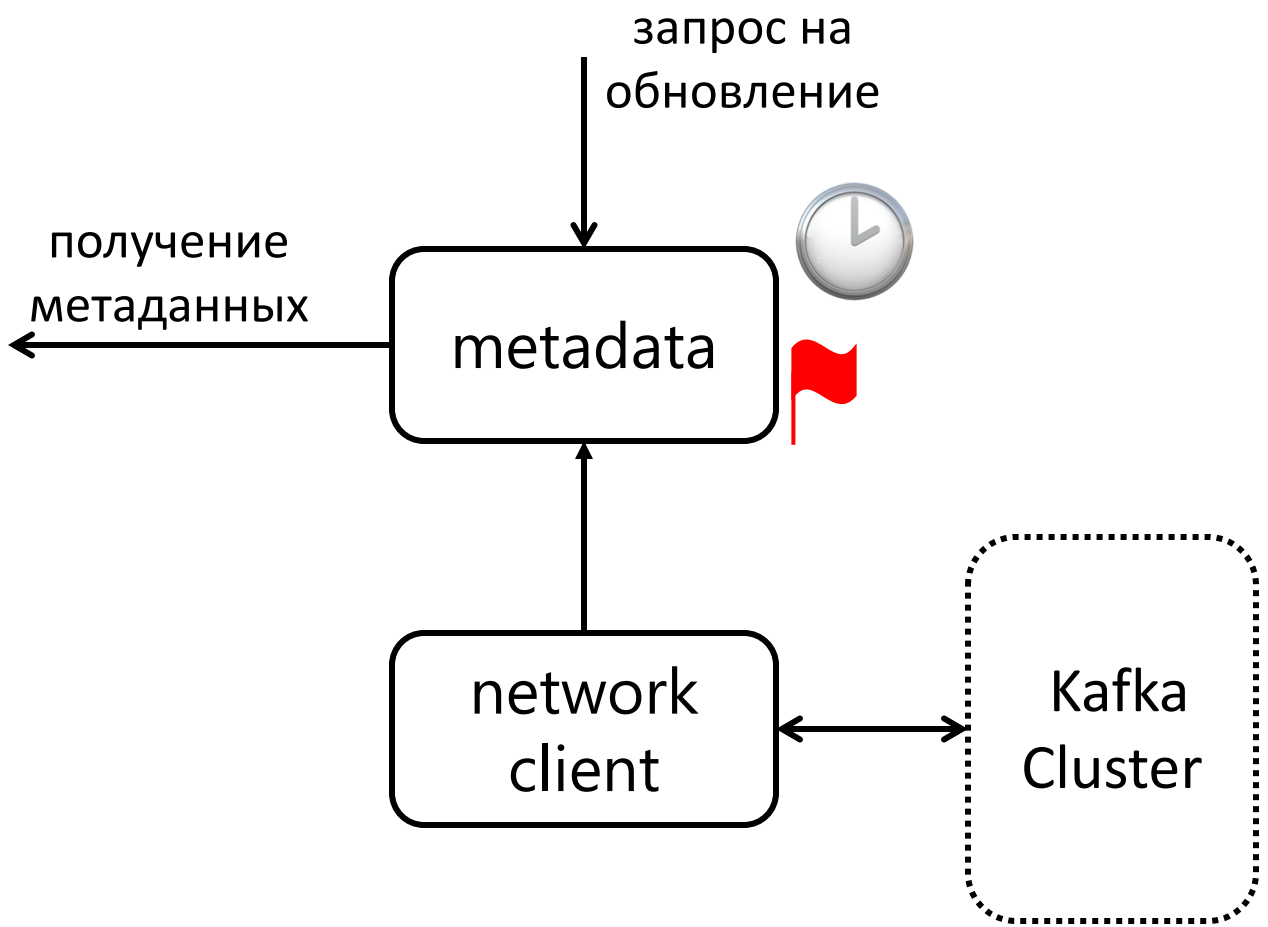
Consumer под капотом — `metadata`



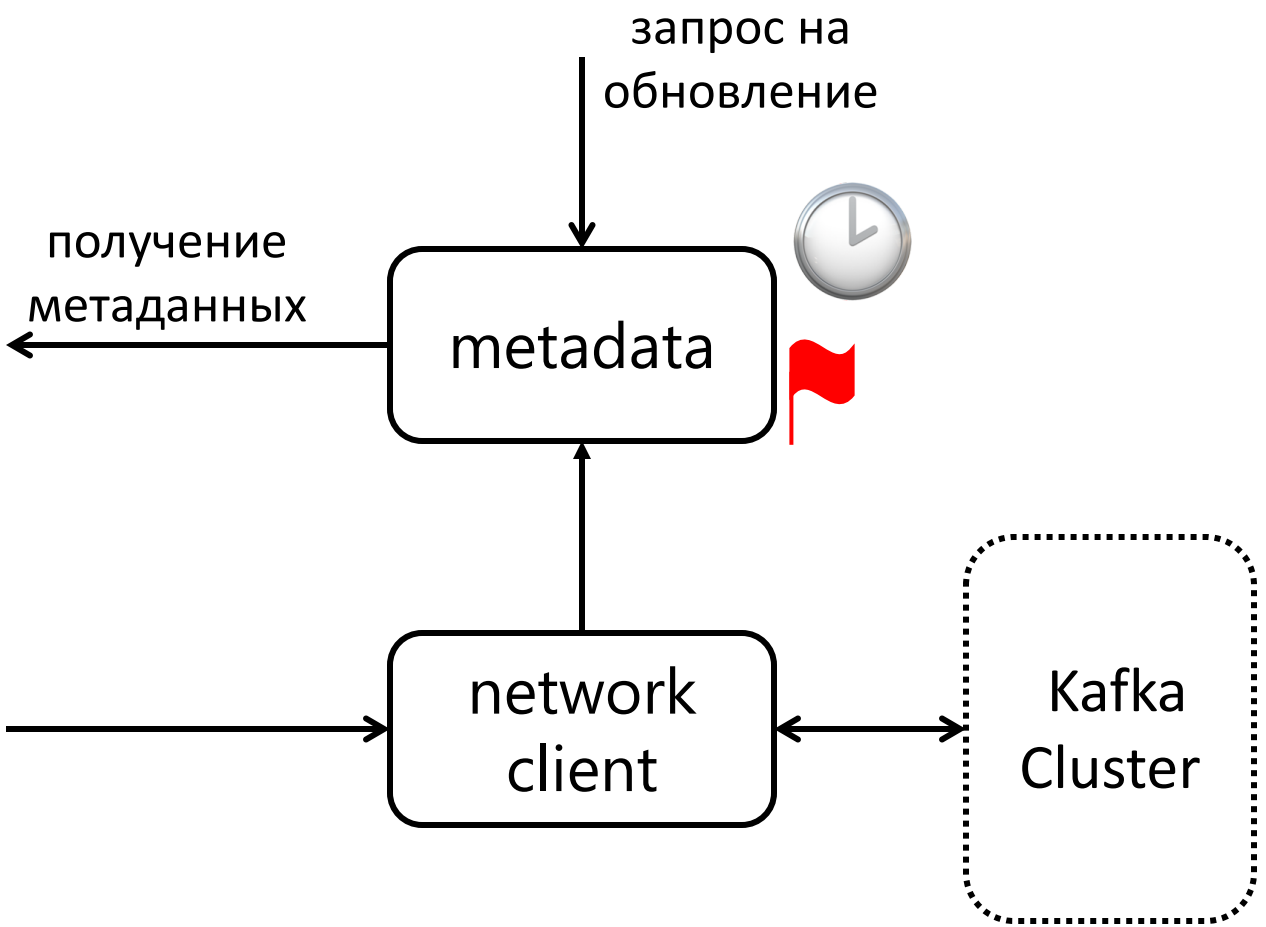
Consumer под капотом — `metadata`



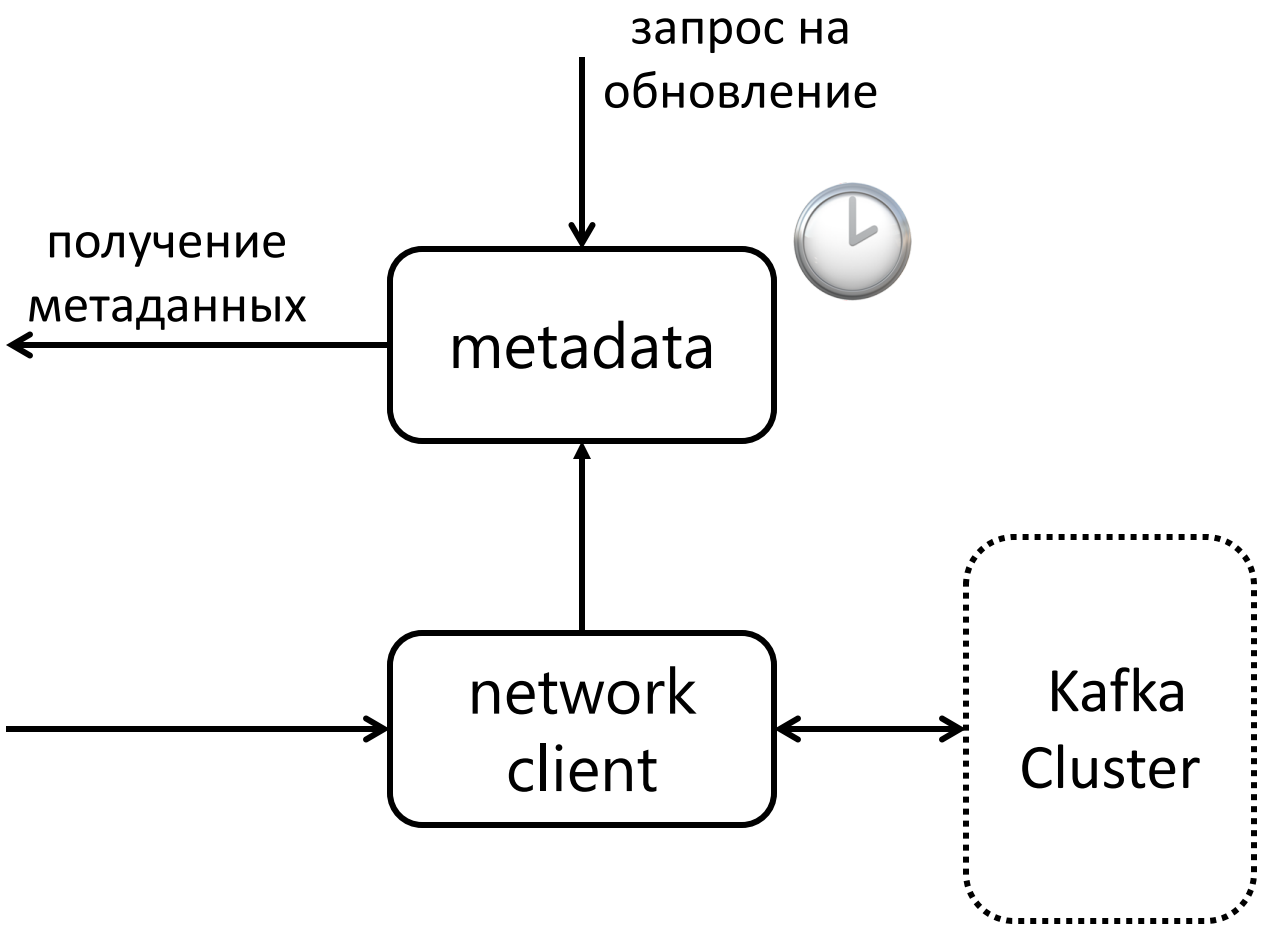
Consumer под капотом — `metadata`



Consumer под капотом — `metadata`



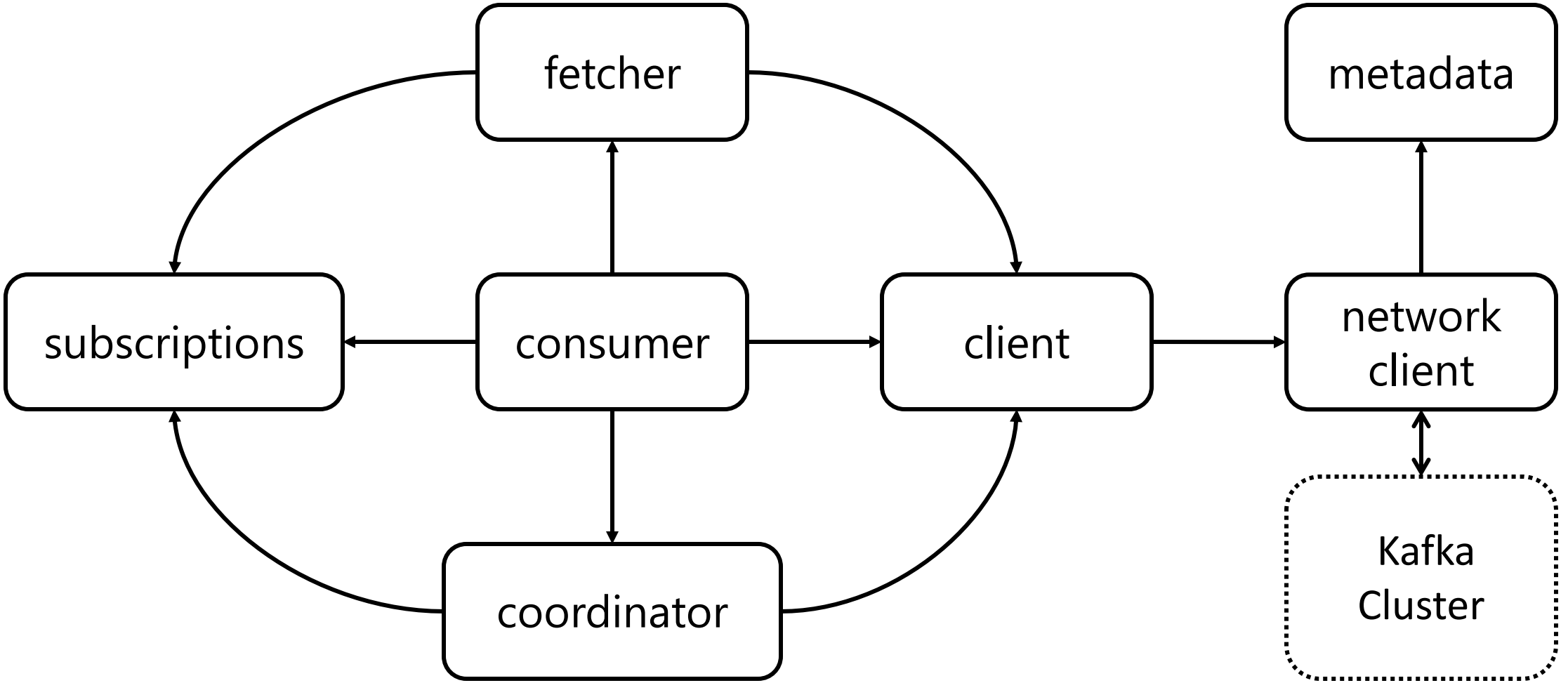
Consumer под капотом — `metadata`



Consumer под капотом

```
— KafkaConsumer                // consumer
— ConsumerMetadata             // metadata
— SubscriptionState            // subscriptions
— ConsumerNetworkClient        // client
— NetworkClient                // network client
— Fetcher                      // fetcher
— ConsumerCoordinator          // coordinator
```

Consumer под капотом



assign VS subscribe

assign VS subscribe

```
assign(Collection<TopicPartition> partitions)
```

`assign` VS `subscribe`

`assign(Collection<TopicPartition> partitions)`
— Конкретные партиции (не топики!)

`assign` VS `subscribe`

`assign(Collection<TopicPartition> partitions)`
— Конкретные партиции (не топики!)
— Изменение количества партиций необходимо отслеживать самостоятельно

`assign` VS `subscribe`

`assign(Collection<TopicPartition> partitions)`
— Конкретные партиции (не топики!)
— Изменение количества партиций необходимо отслеживать самостоятельно

Use case:
— внешняя координация чтения (оффсеты и партиции)
— чтение конкретных событий

assign VS subscribe

```
subscribe(Collection<String> topics)
```

`assign` VS `subscribe`

`subscribe(Collection<String> topics)`

— Автоматически подхватывает новые партиции

assign VS subscribe

`subscribe(Collection<String> topics)`
— Автоматически подхватывает новые партиции

`subscribe(Pattern pattern)`

KIP-191
v.1.0.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-191%3A+KafkaConsumer.subscribe%28%29+overload+that+takes+just+Pattern

`assign` VS `subscribe`

`subscribe(Collection<String> topics)`
— Автоматически подхватывает новые партиции

`subscribe(Pattern pattern)`
— Подписка на топики по шаблону

KIP-191
v.1.0.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-191%3A+KafkaConsumer.subscribe%28%29+overload+that+takes+just+Pattern

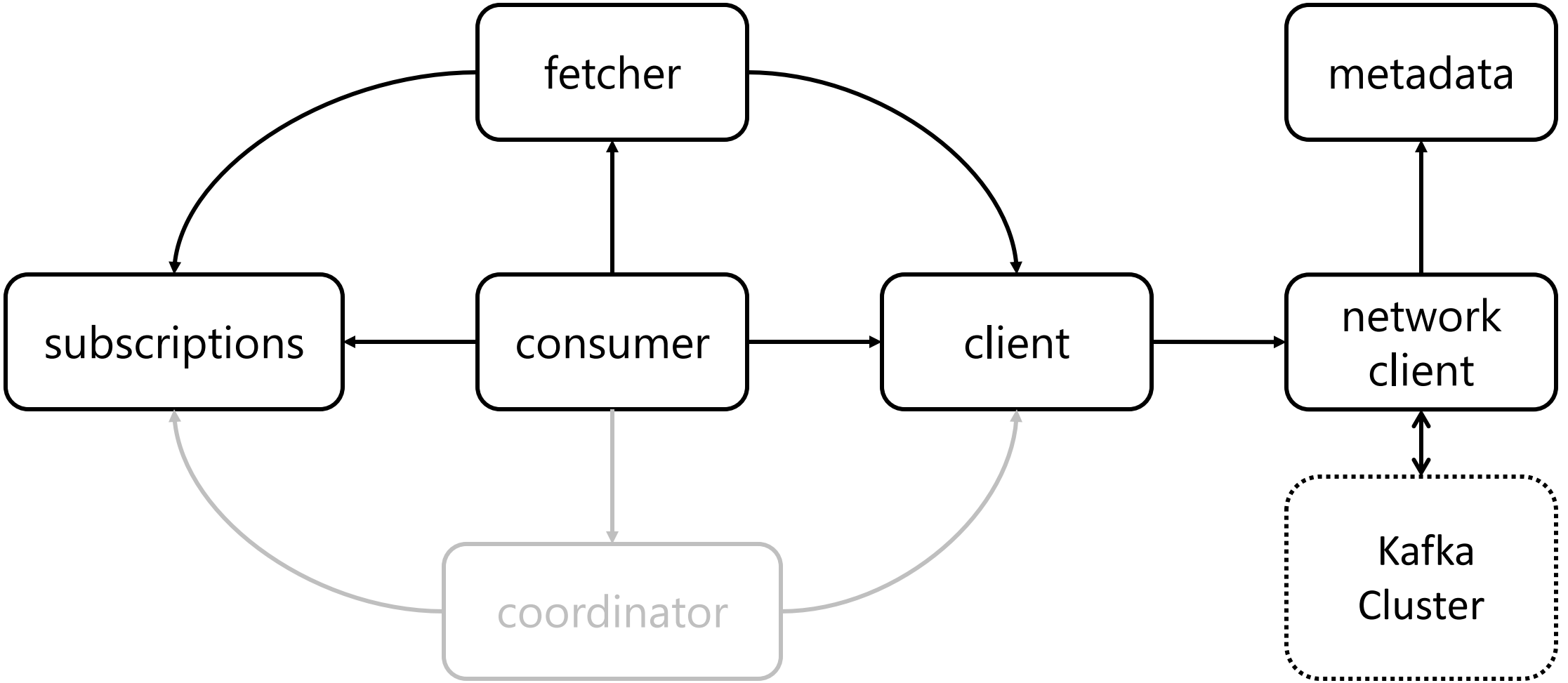
`assign` VS `subscribe`

`subscribe(Collection<String> topics)`
— Автоматически подхватывает новые партиции

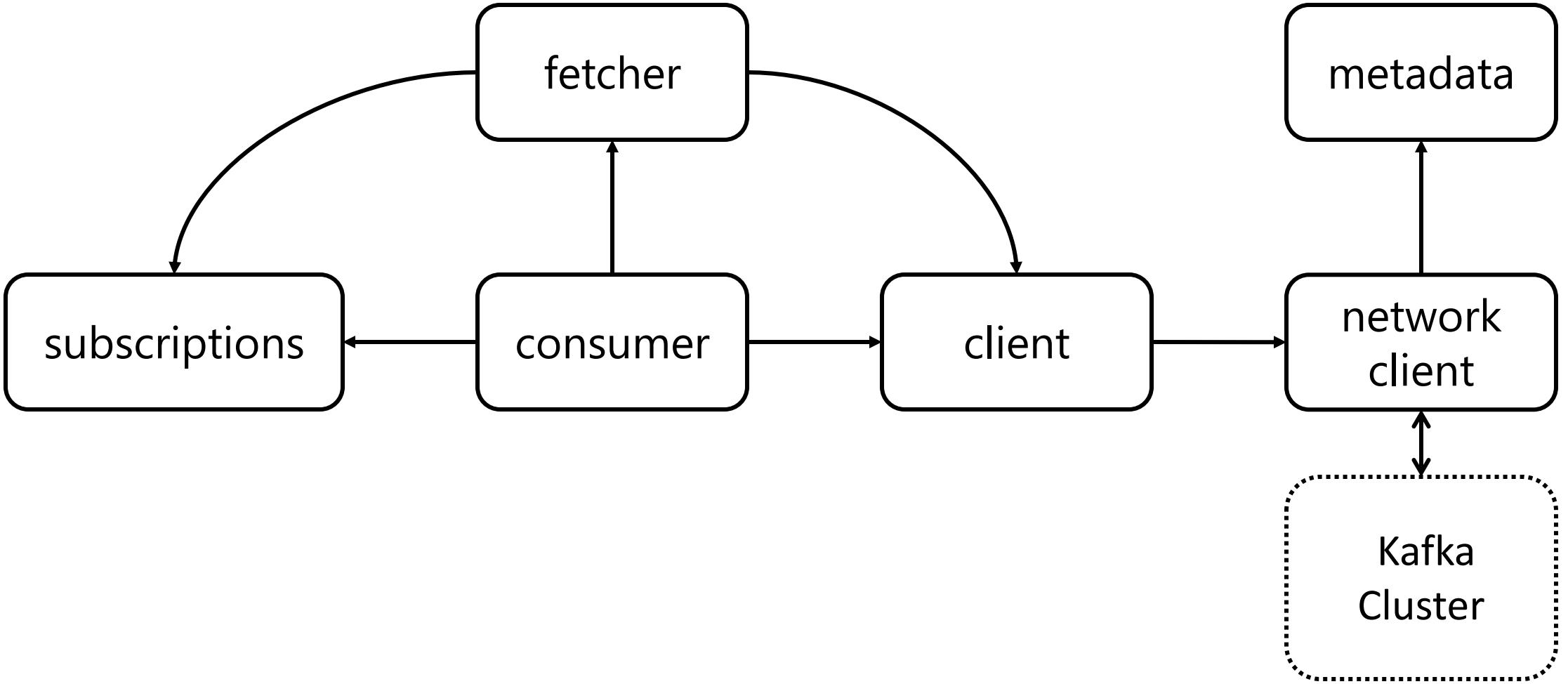
`subscribe(Pattern pattern)`
— Подписка на топики по шаблону
— Автоматически подхватывает новые топики

KIP-191
v.1.0.0

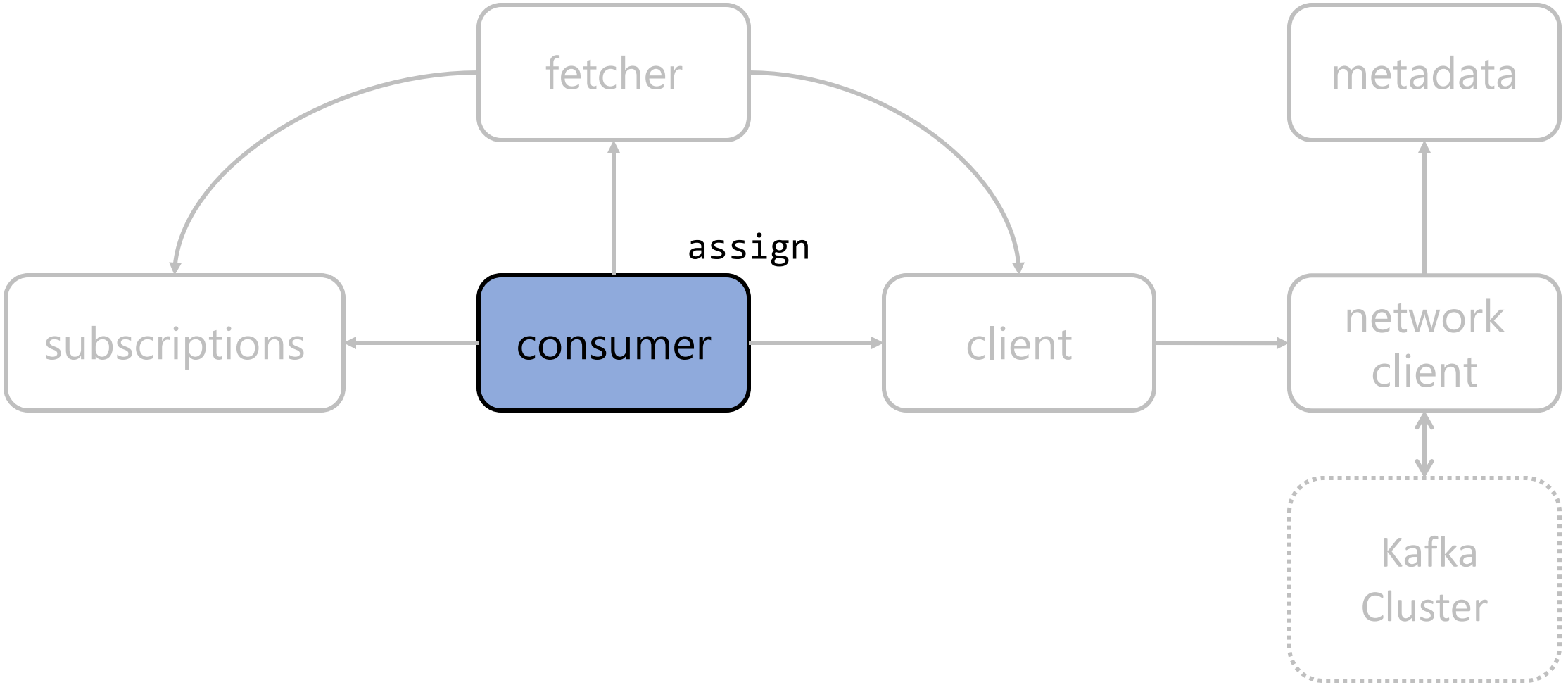
Consumer под капотом



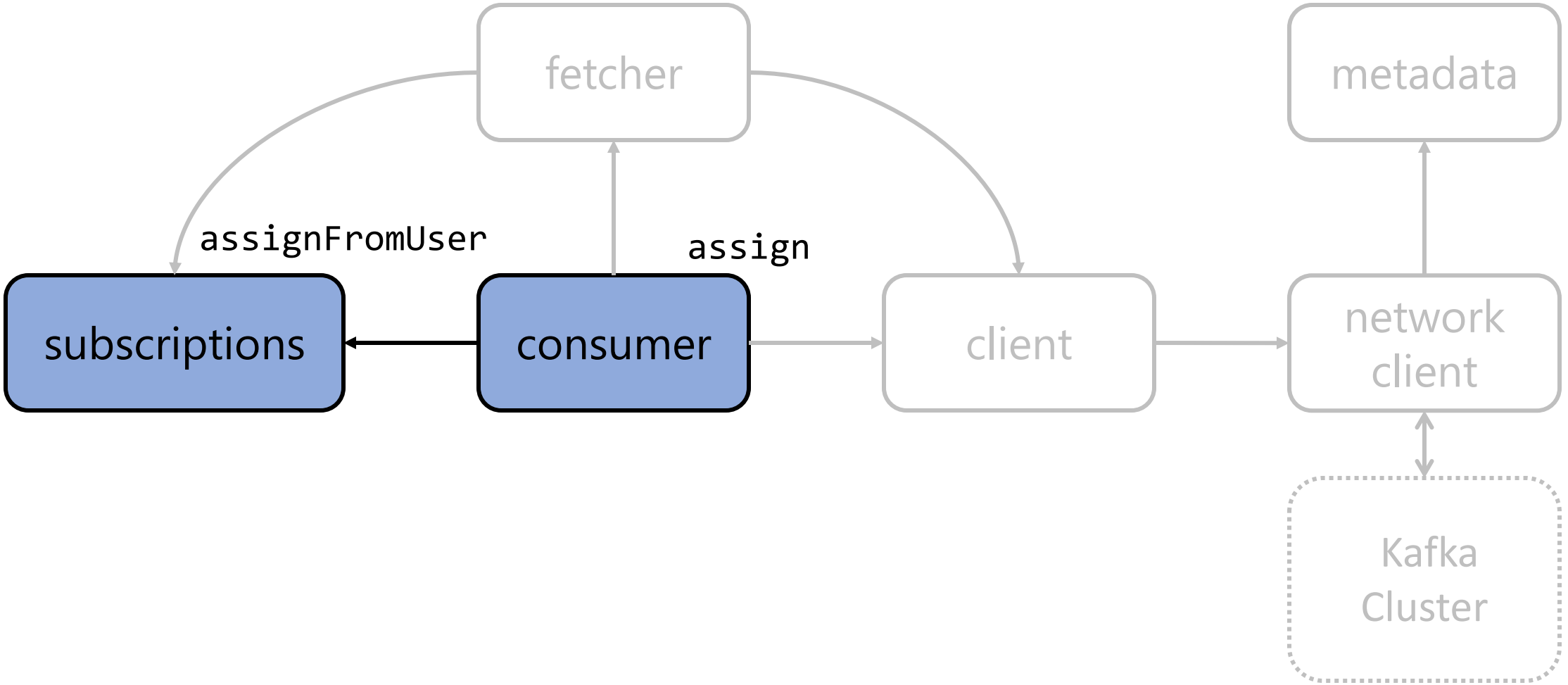
Consumer под капотом



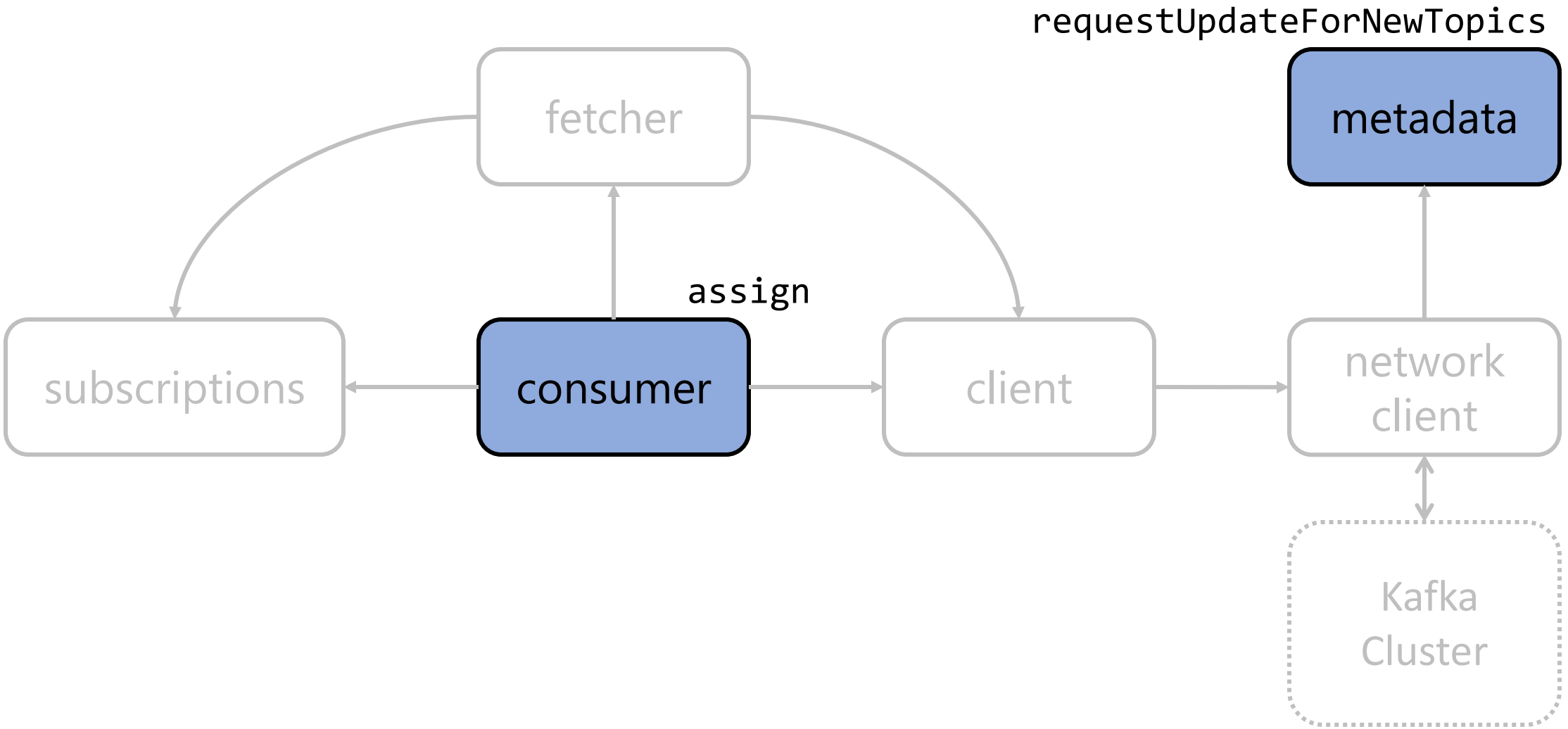
Consumer под капотом



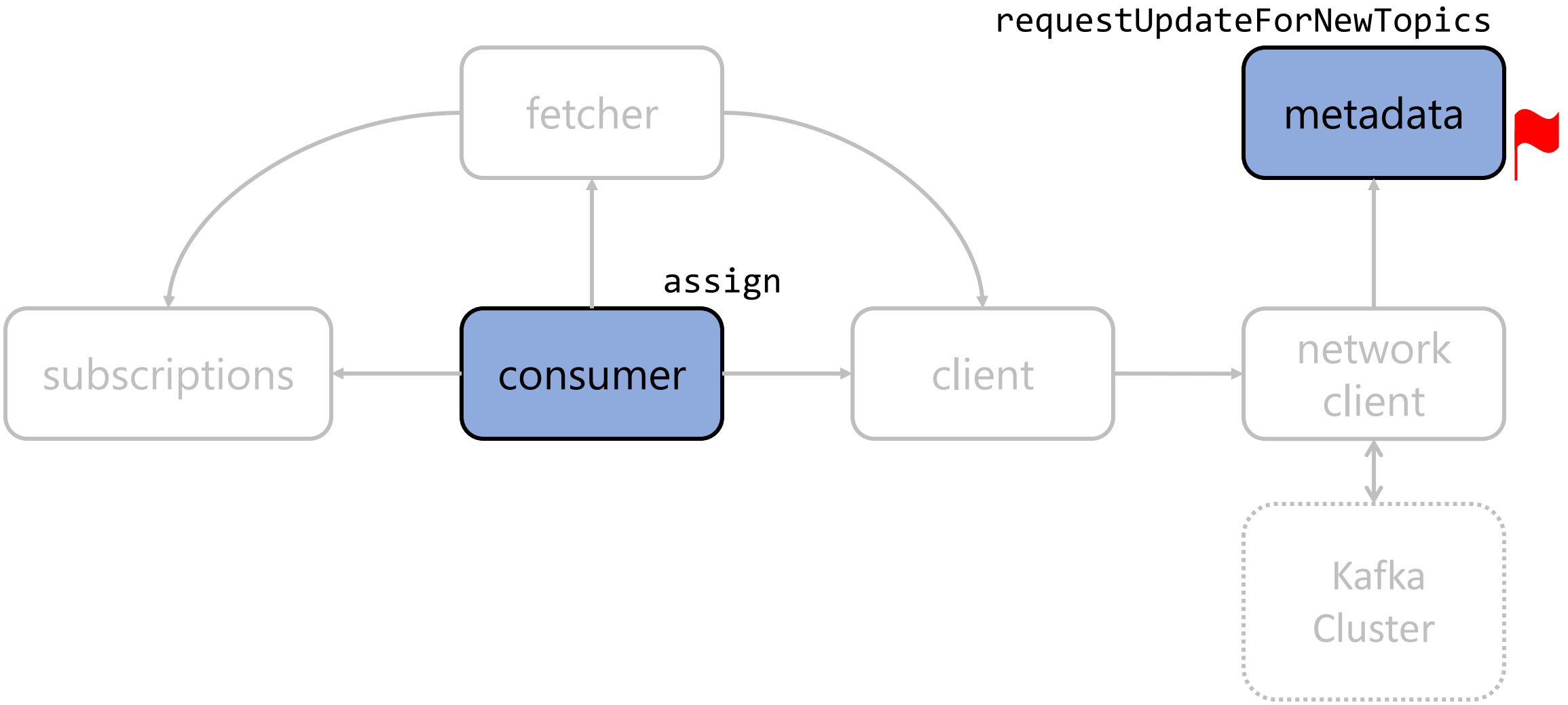
Consumer под капотом



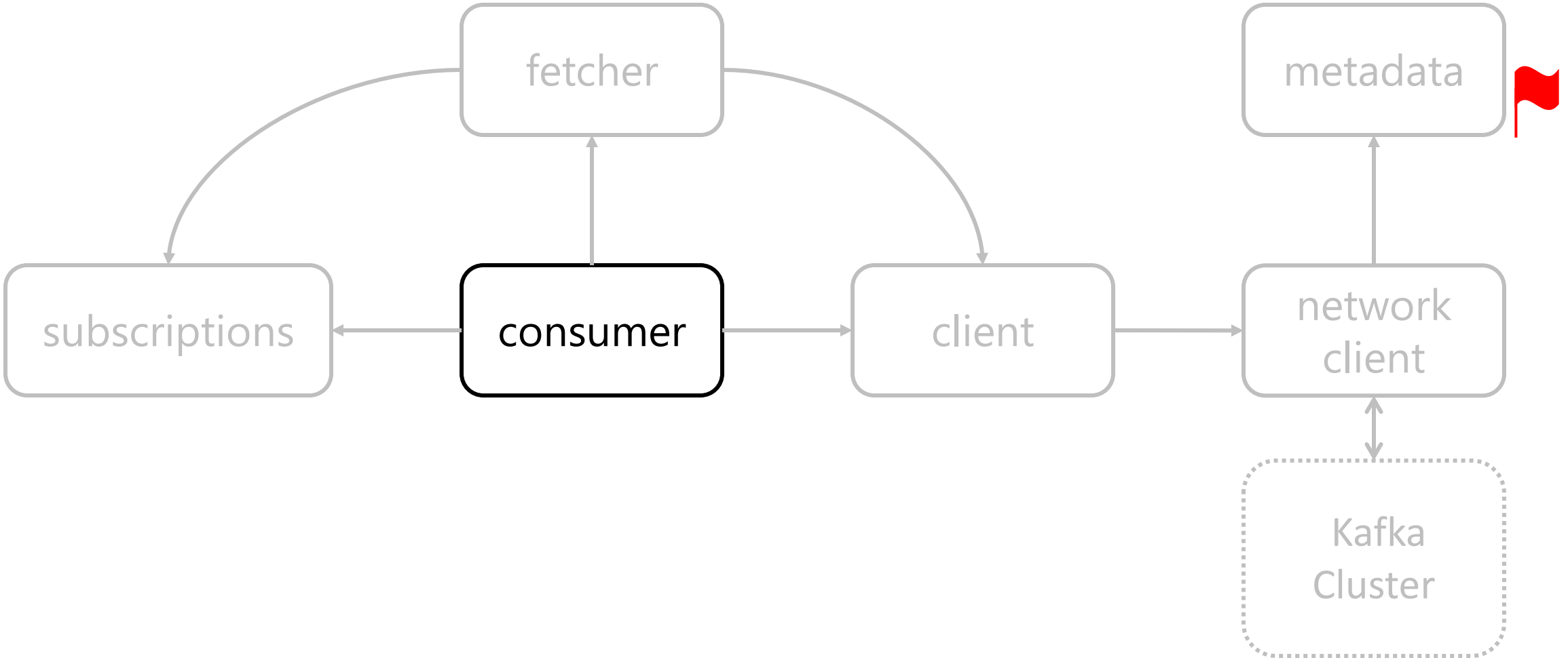
Consumer под капотом



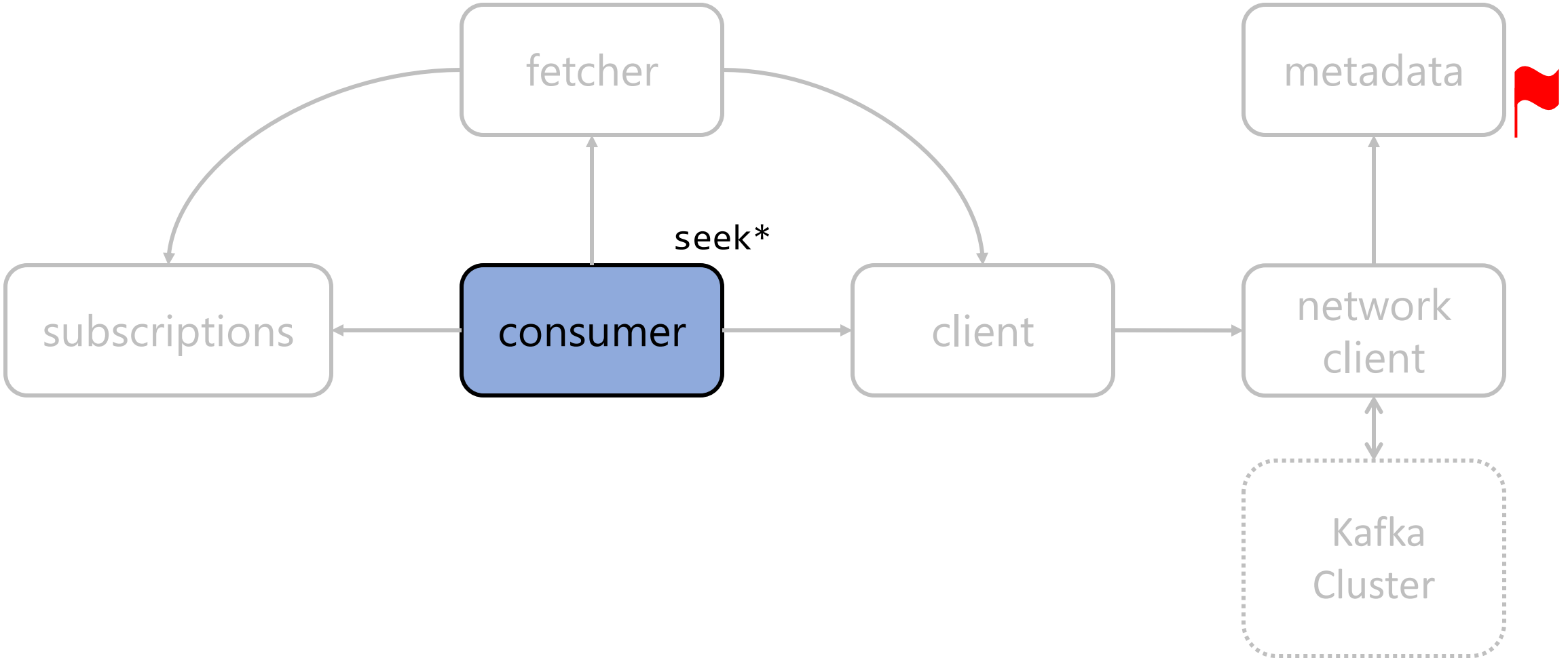
Consumer под капотом



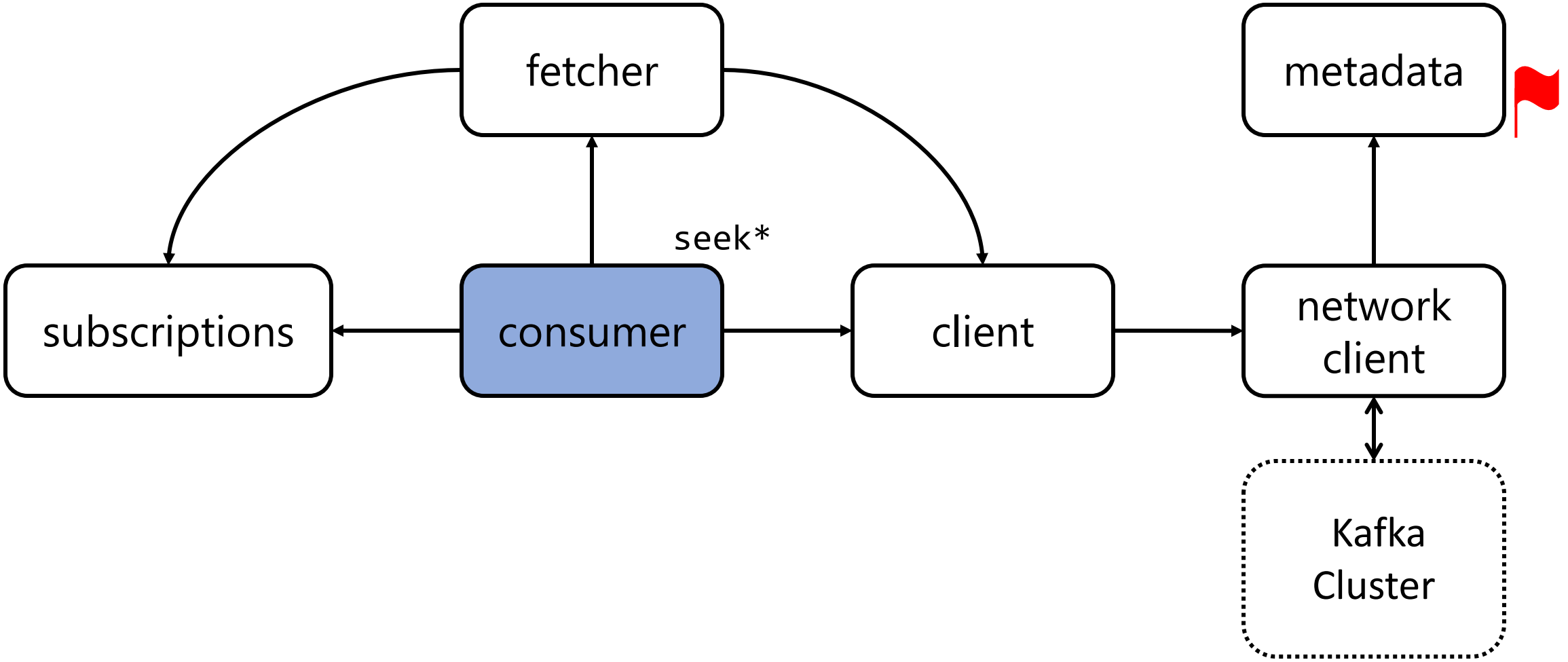
Consumer под капотом



Consumer под капотом



Consumer под капотом



Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Вызов метода изменит «координаты» консьюмера!

Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Вызов метода изменит «координаты» консьюмера!

— beginningOffsets

— endOffsets

— offsetsForTimes

Вызов метода НЕ изменит «координаты» консьюмера!

Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Вызов метода изменит «координаты» консьюмера!

— beginningOffsets

— endOffsets

— offsetsForTimes

Вызов метода НЕ изменит «координаты» консьюмера!

Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Вызов метода изменит «координаты» консьюмера!

— beginningOffsets

— endOffsets

— offsetsForTimes

Вызов метода НЕ изменит «координаты» консьюмера!

Consumer под капотом

seek*

— seekToBeginning

— seekToEnd

— seek

Вызов метода изменит «координаты» консьюмера!

— beginningOffsets

— endOffsets

— offsetsForTimes

Вызов метода НЕ изменит «координаты» консьюмера!

Consumer под капотом

— beginningOffsets
— endOffsets
— offsetsForTimes

Consumer под капотом

— beginningOffsets

— endOffsets

— offsetsForTimes

```
request.timeout.ms
      = 30 000
```

Consumer под капотом

— beginningOffsets

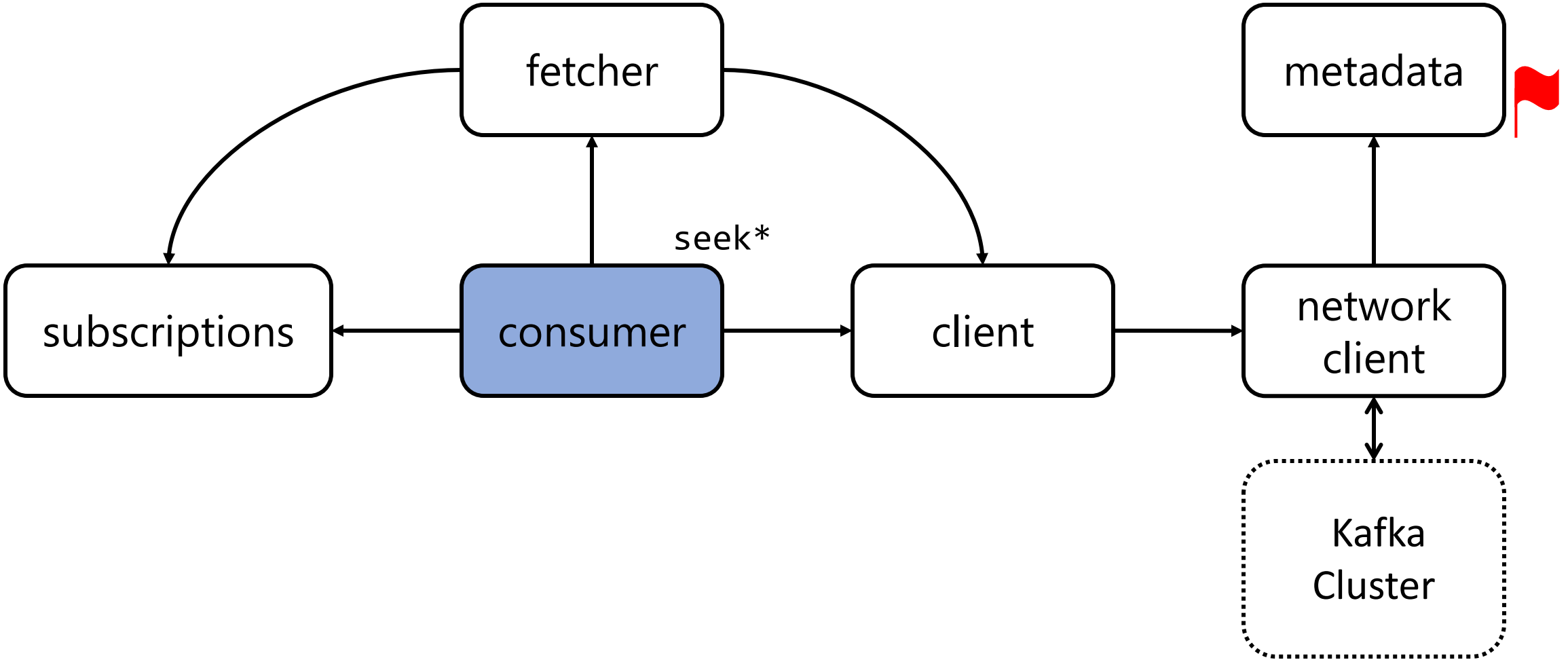
— endOffsets

— offsetsForTimes

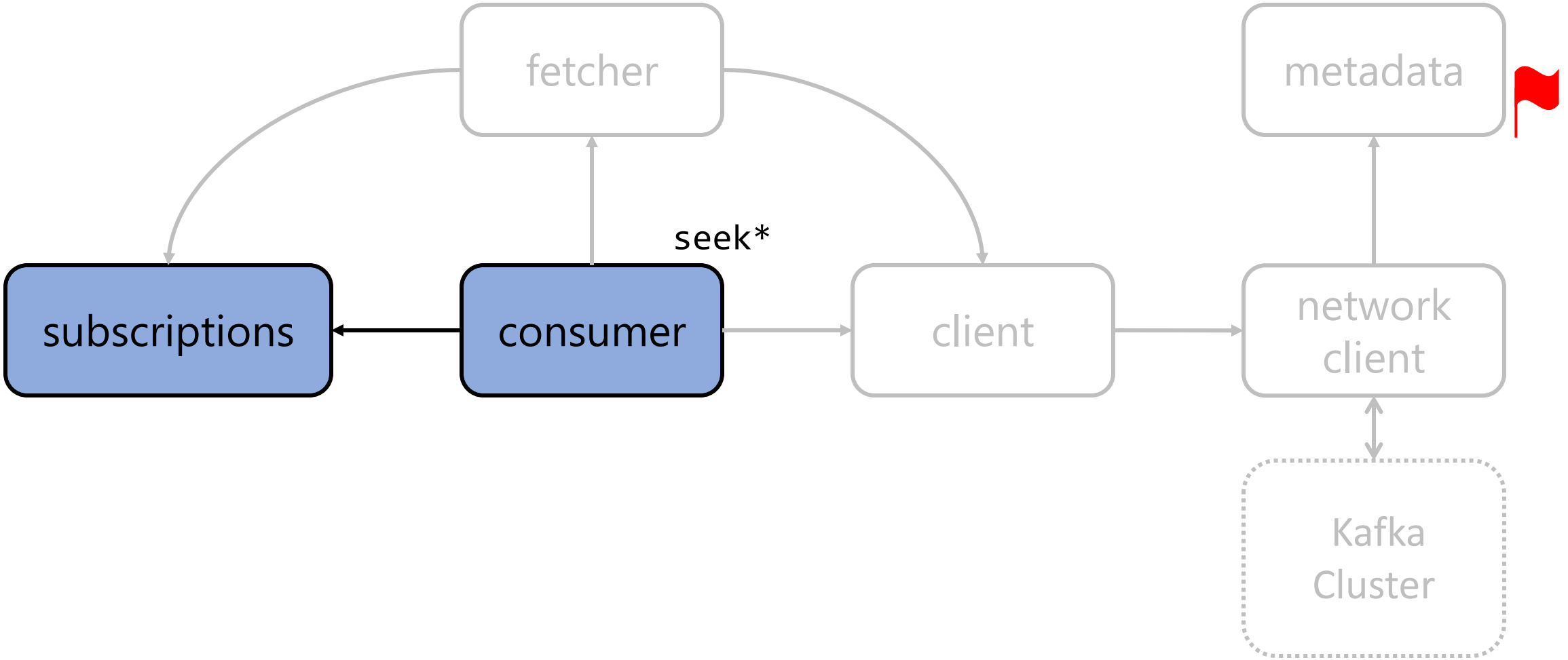
```
default.api.timeout.ms
        = 60 000
```

KIP-266
v.2.0.0

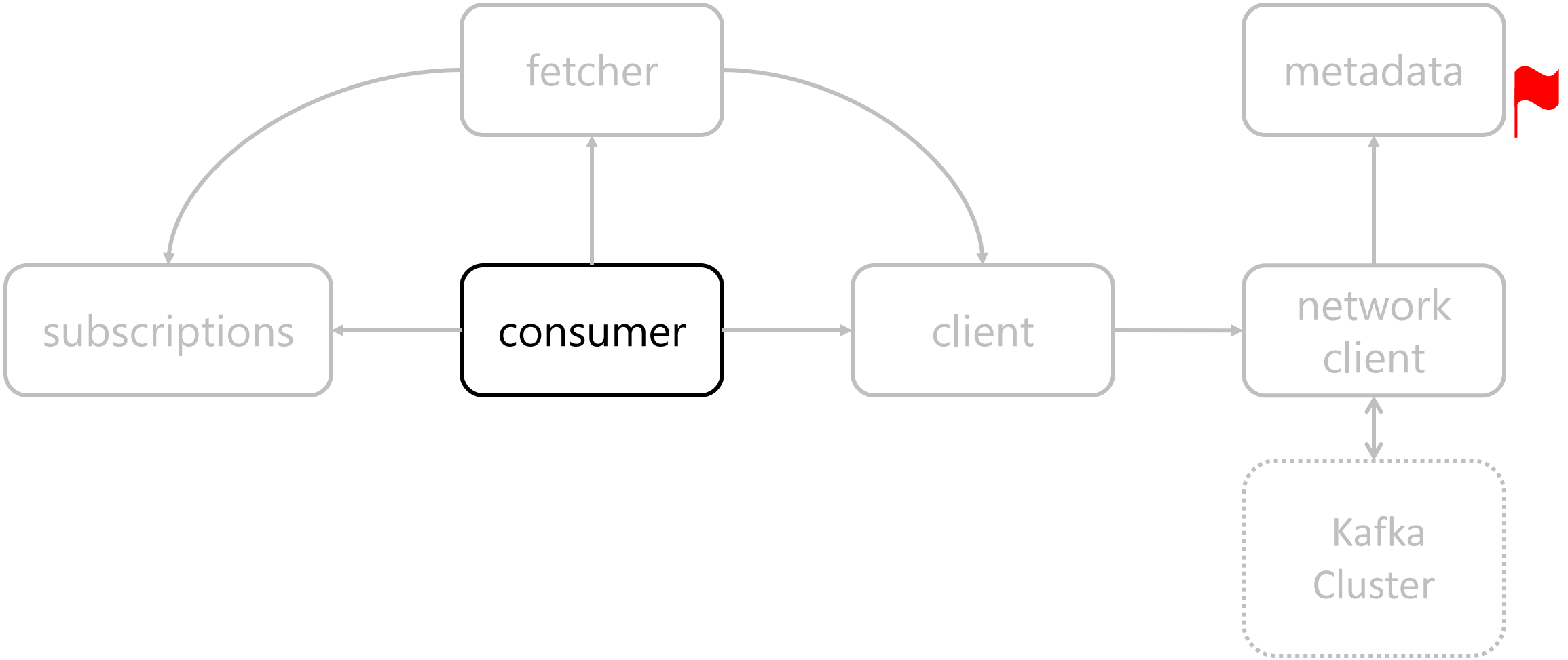
Consumer под капотом



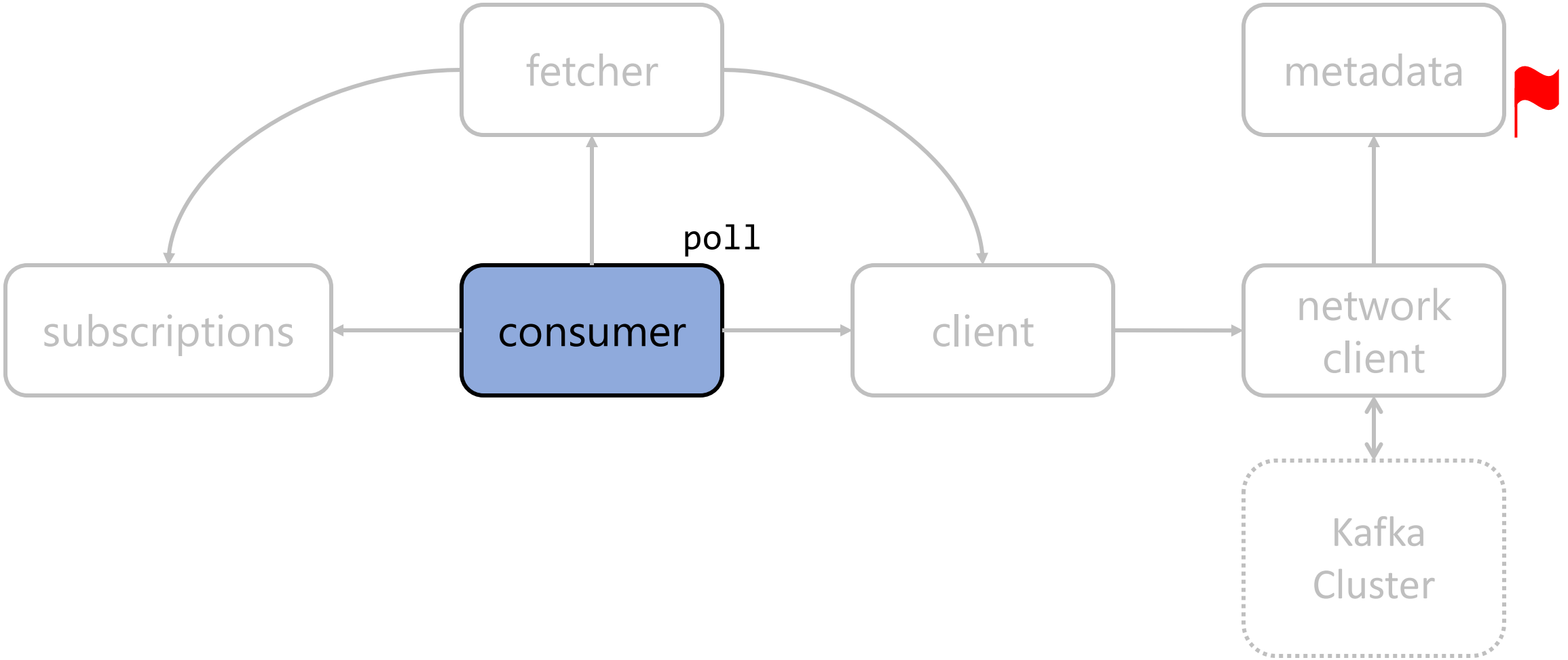
Consumer под капотом



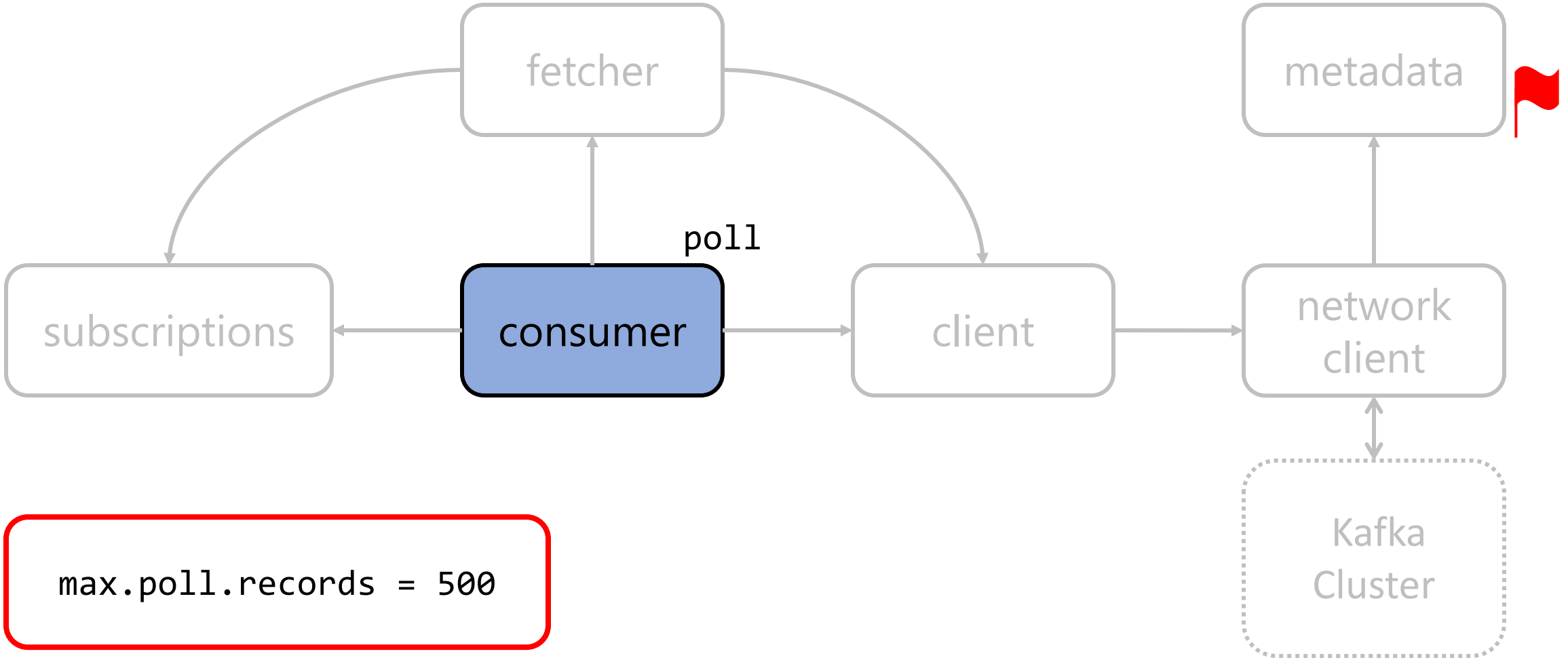
Consumer под капотом



Consumer под капотом

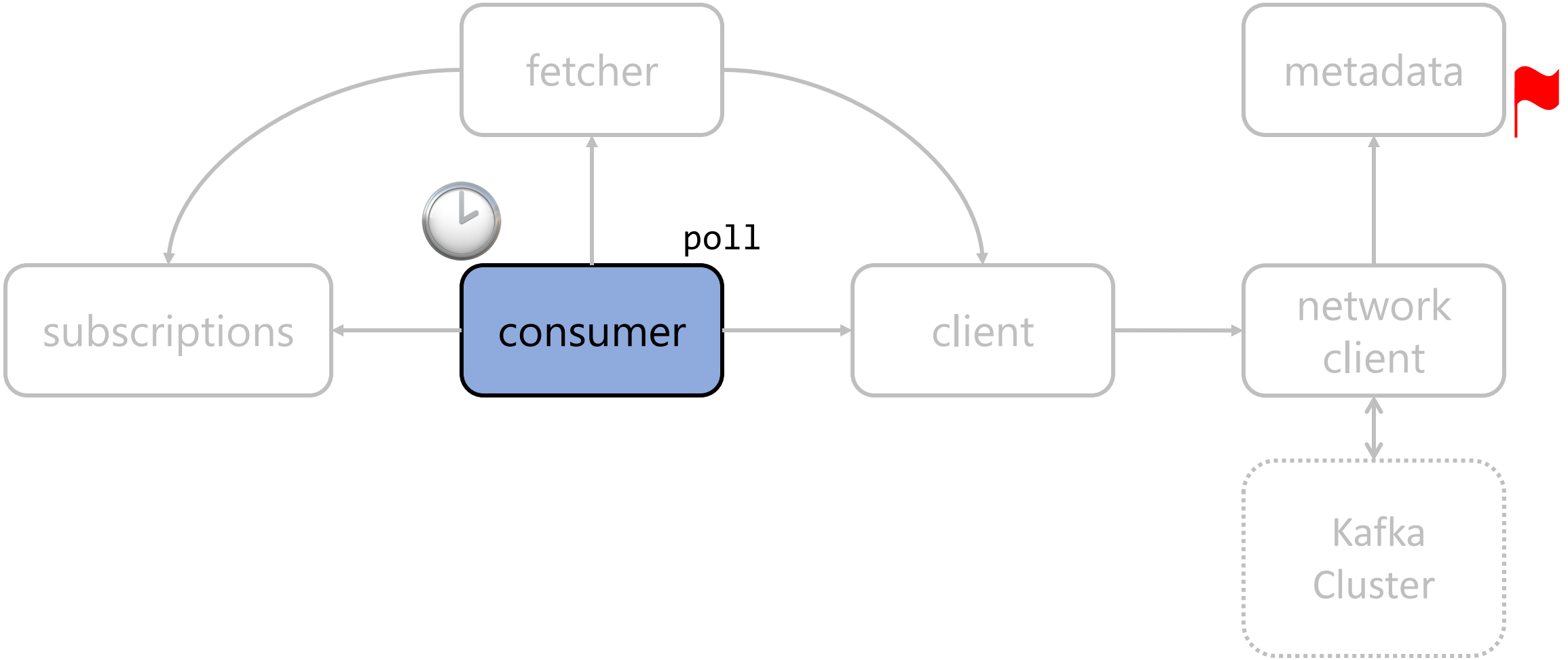


Consumer под капотом

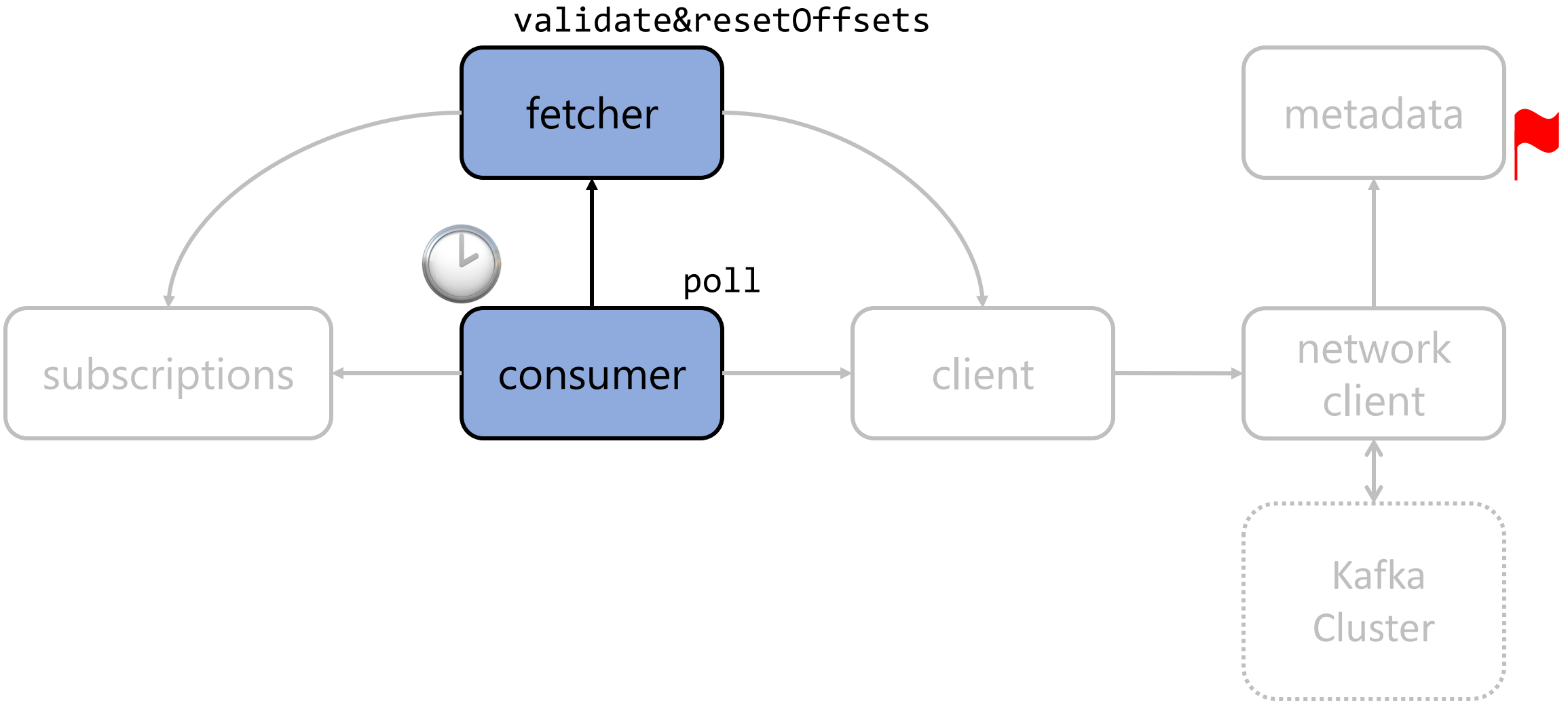



```
max.poll.records = 500
```

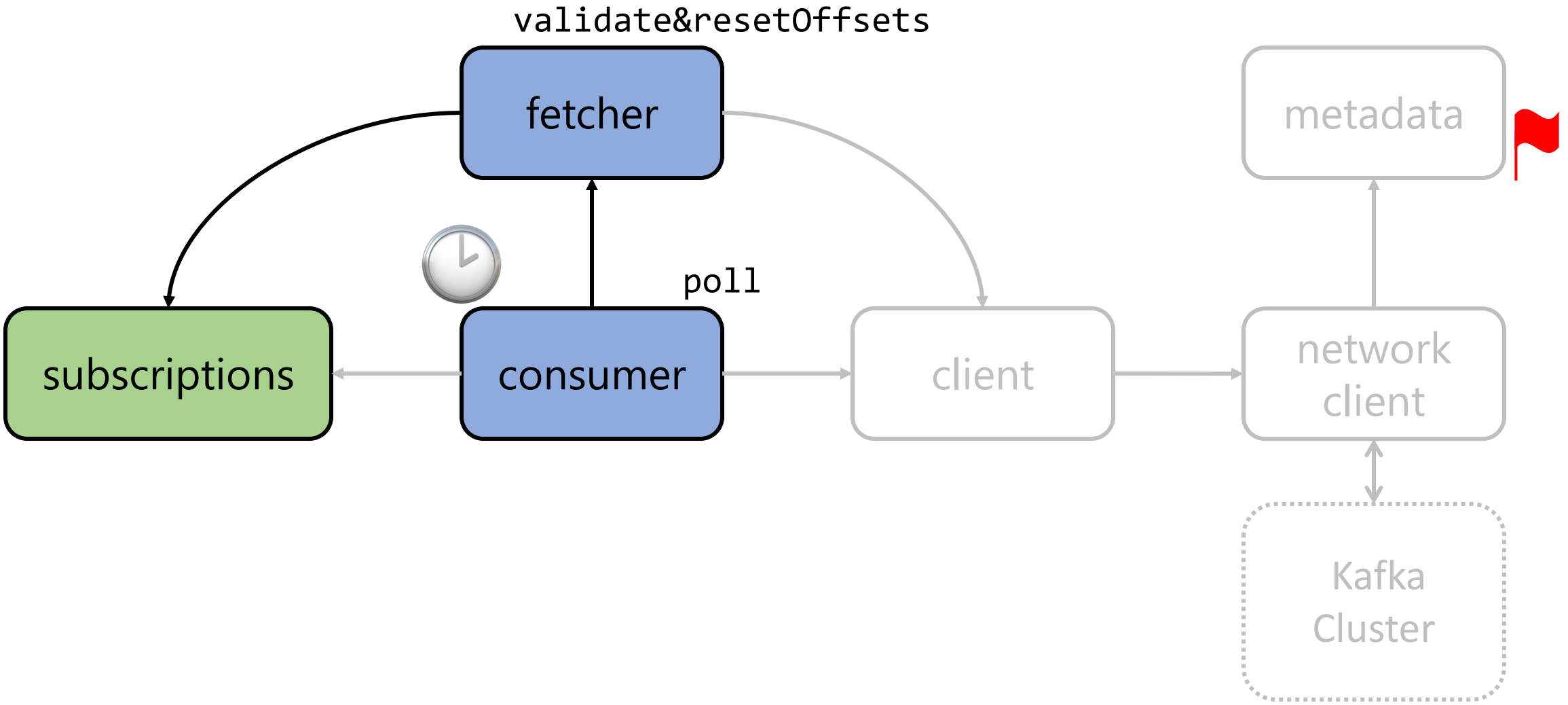
Consumer под капотом



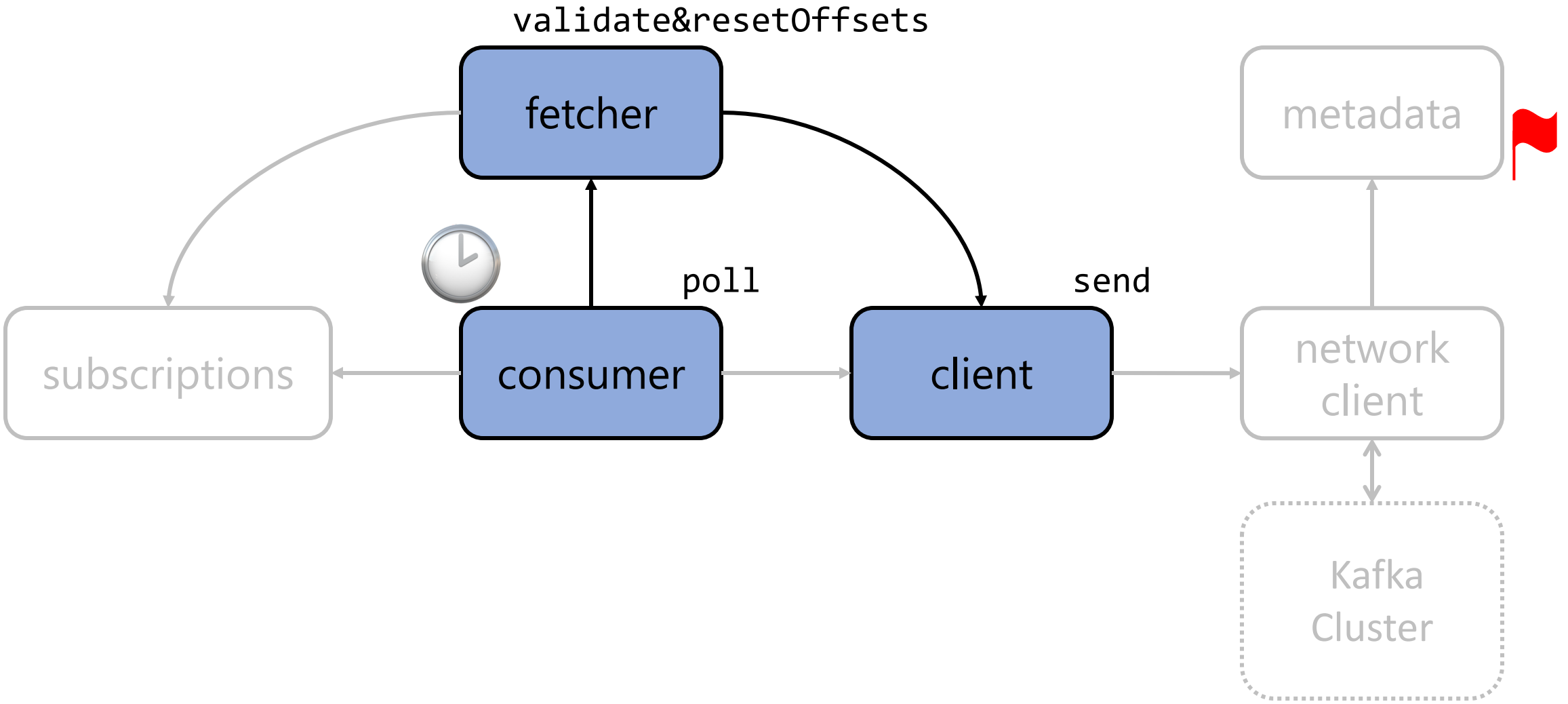
Consumer под капотом



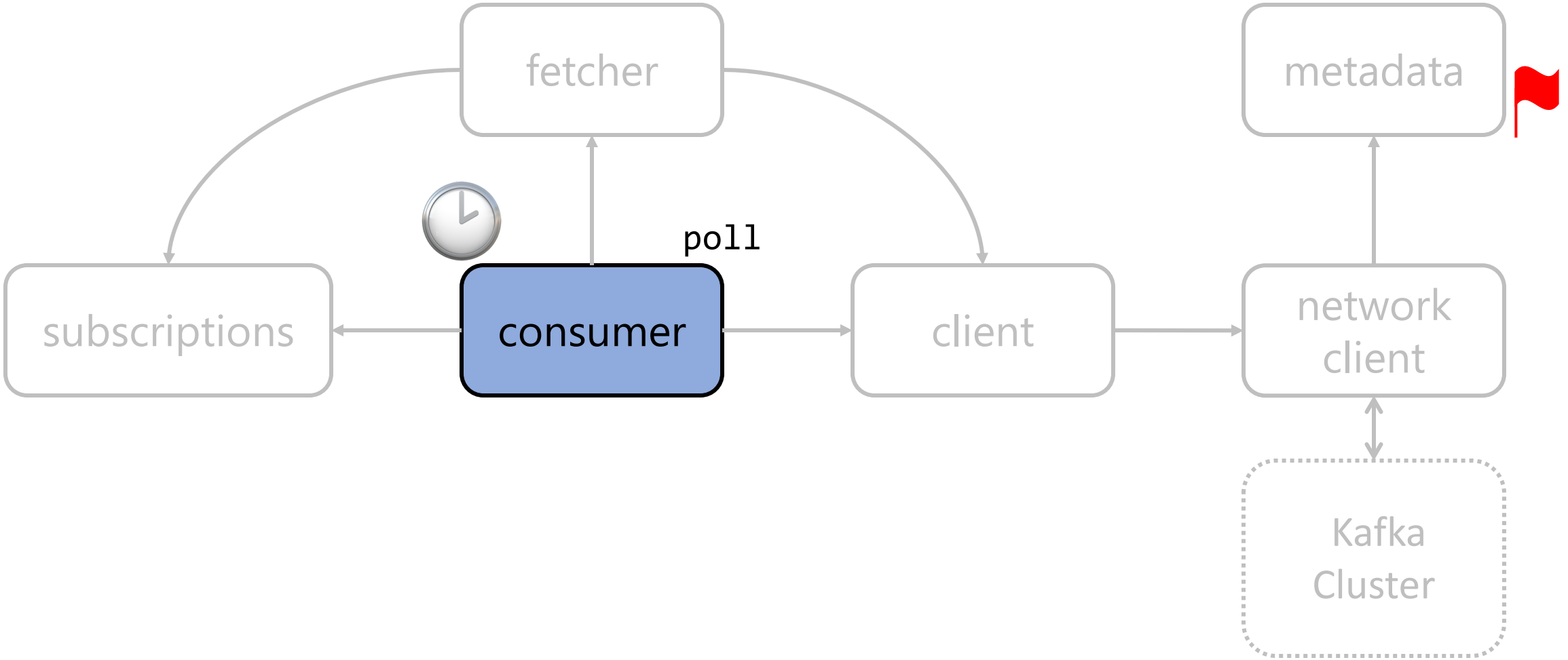
Consumer под капотом



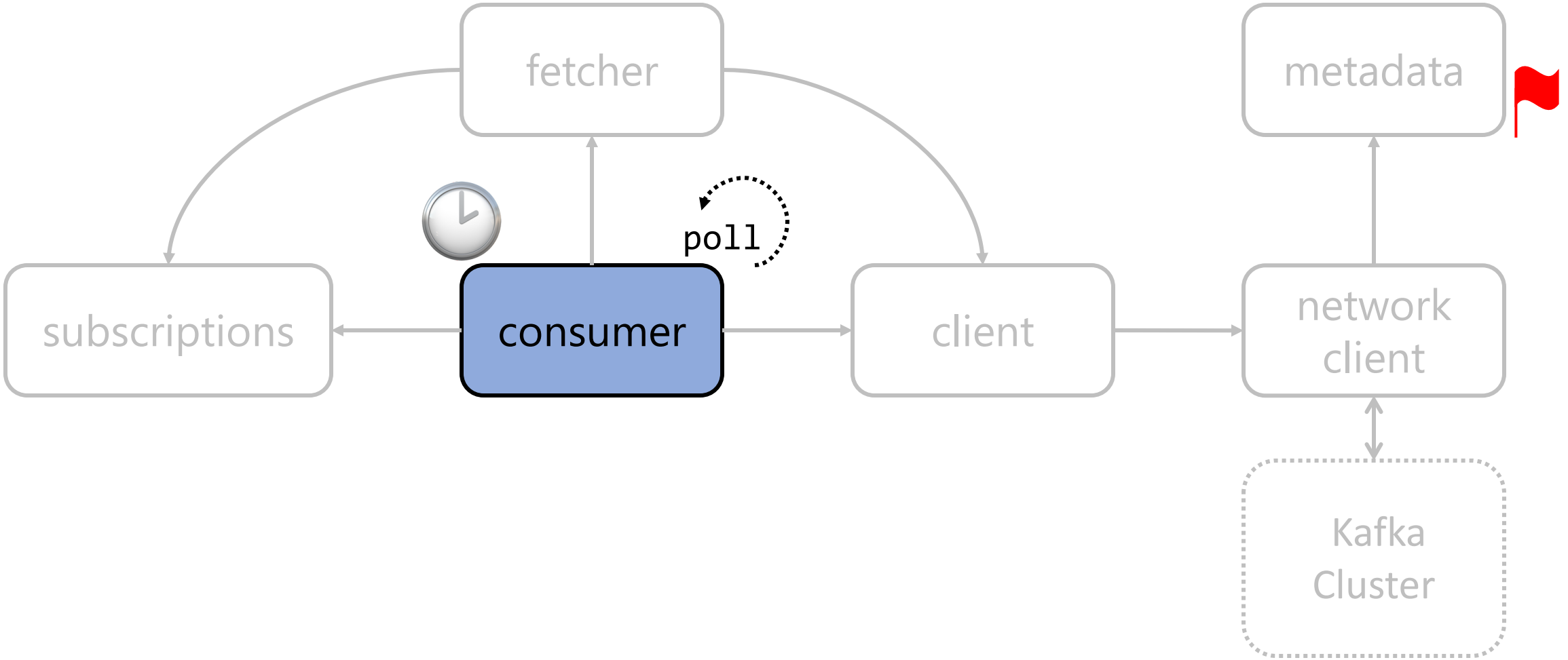
Consumer под капотом



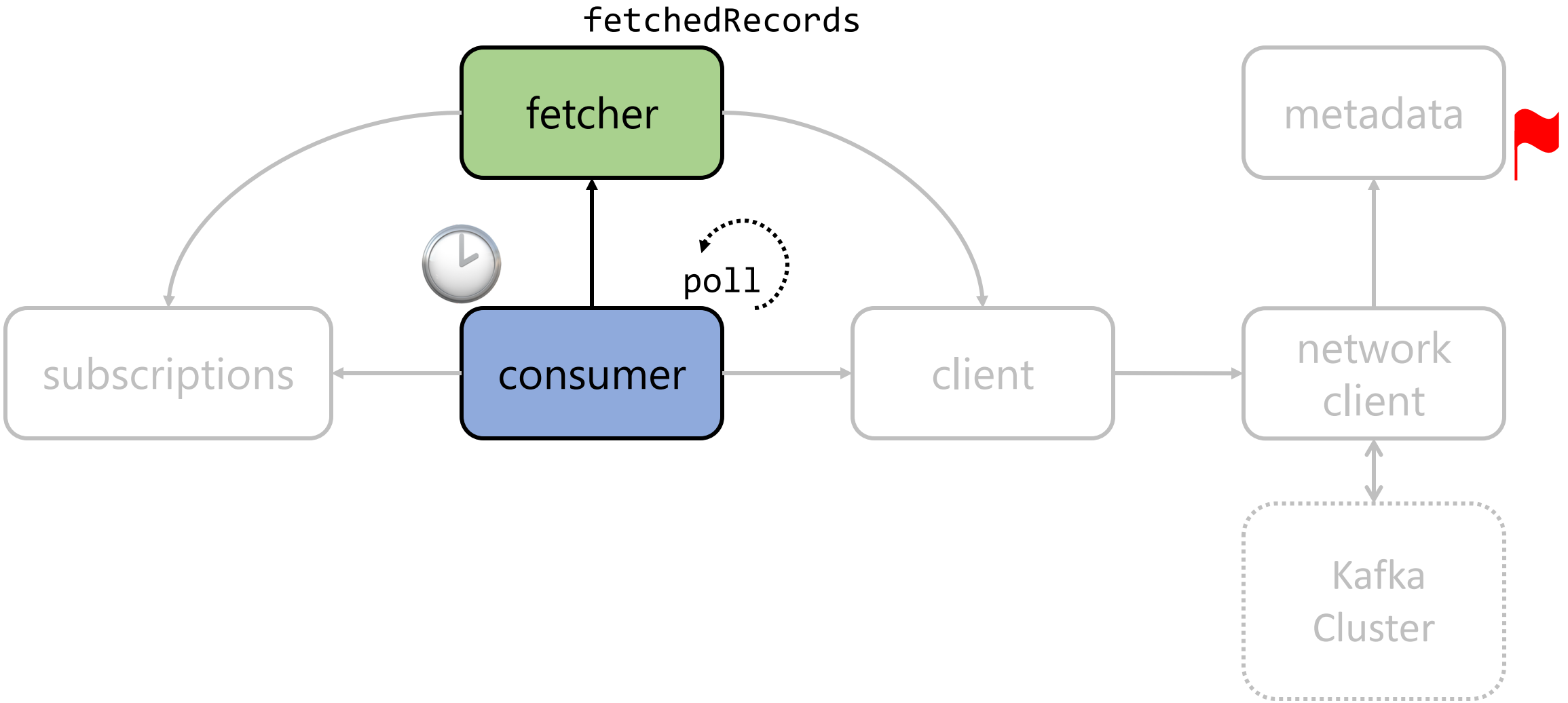
Consumer под капотом



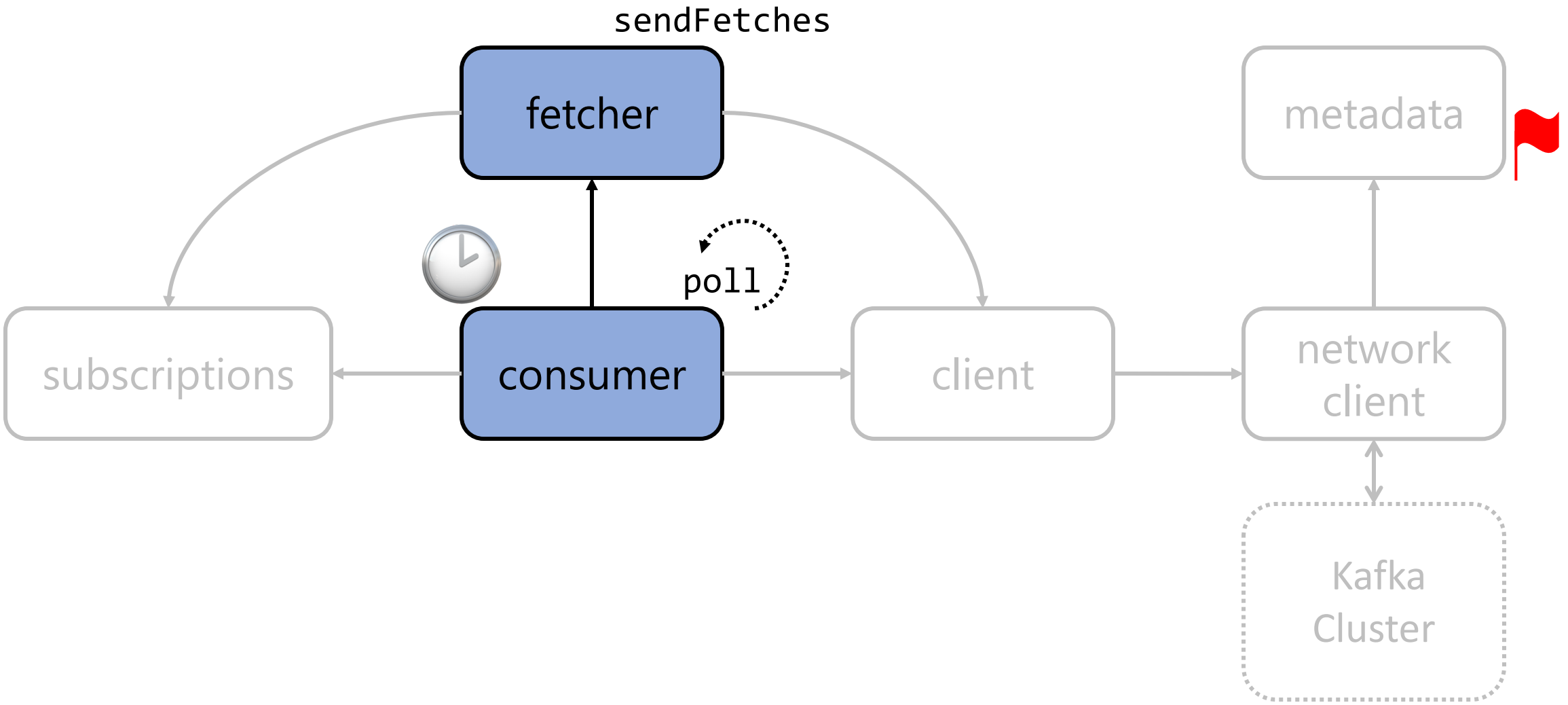
Consumer под капотом



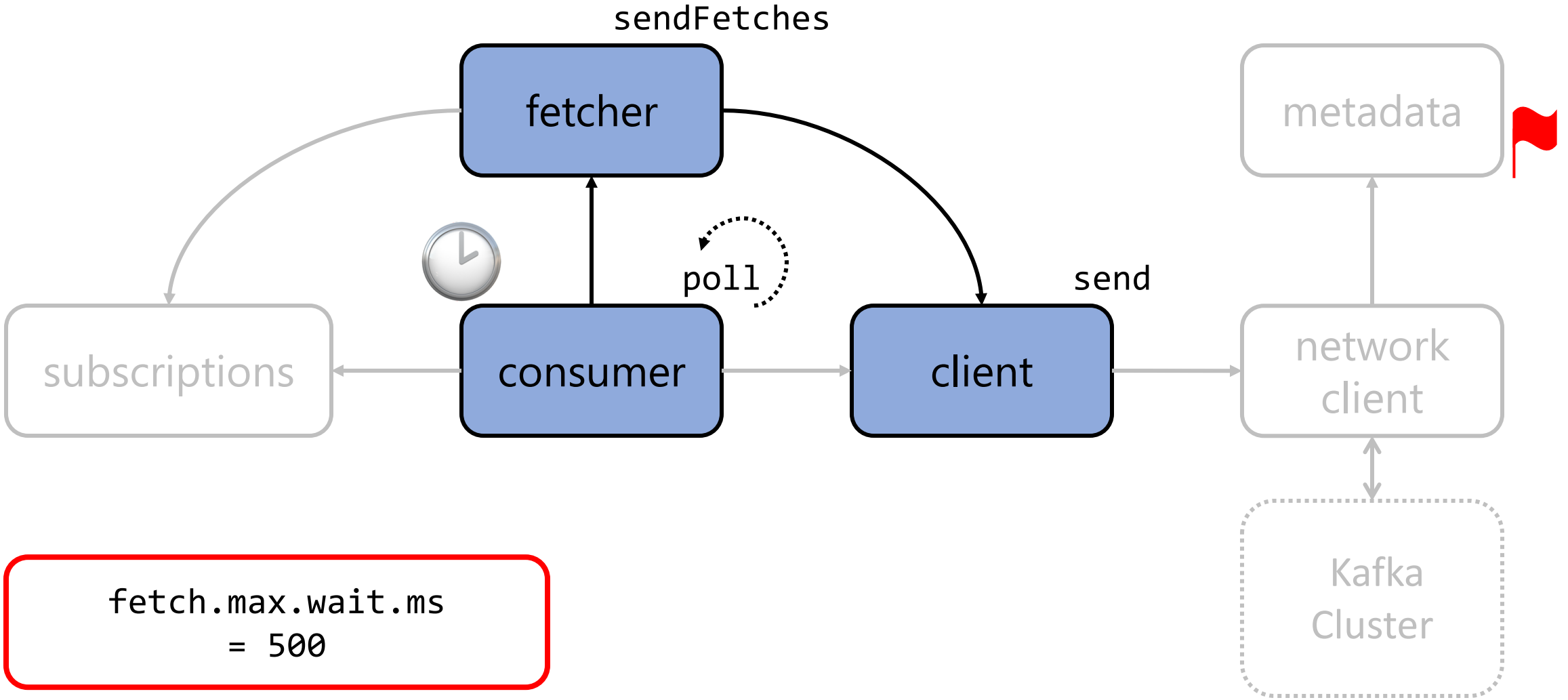
Consumer под капотом



Consumer под капотом

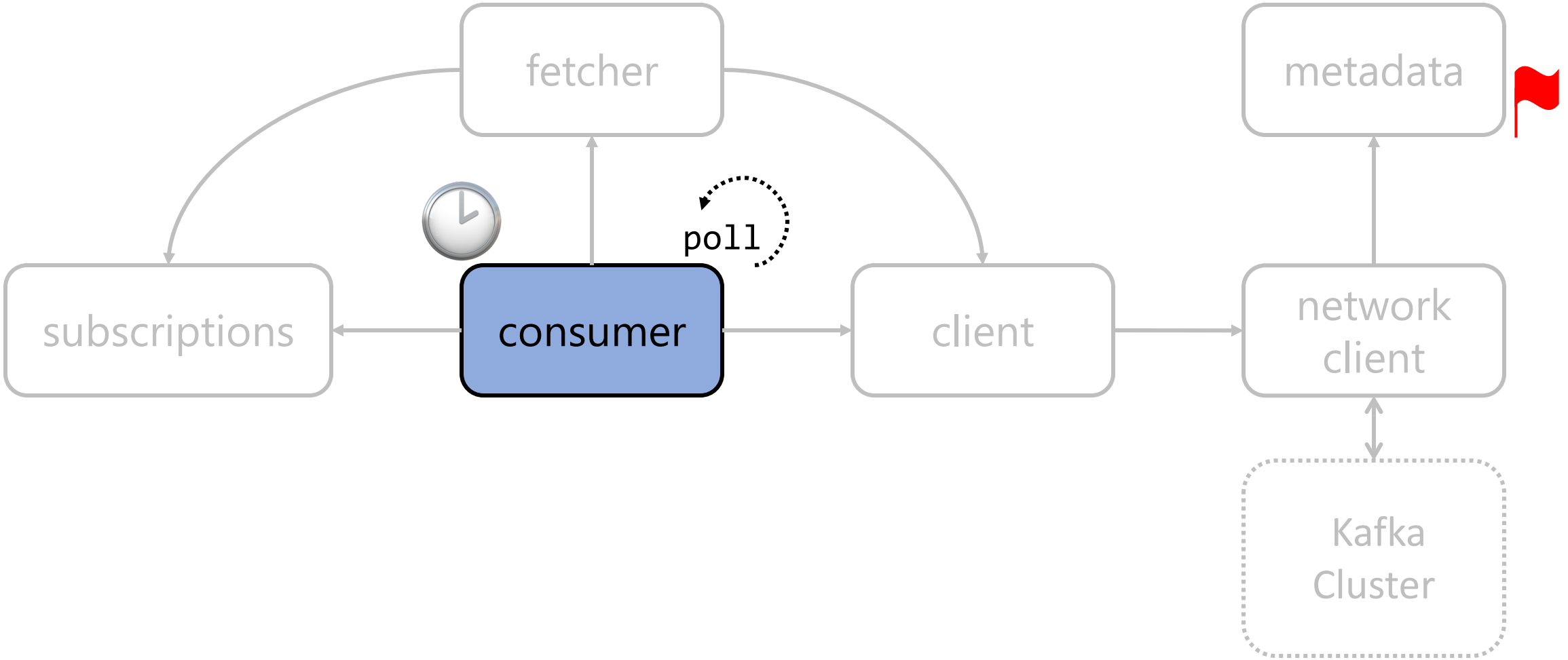


Consumer под капотом

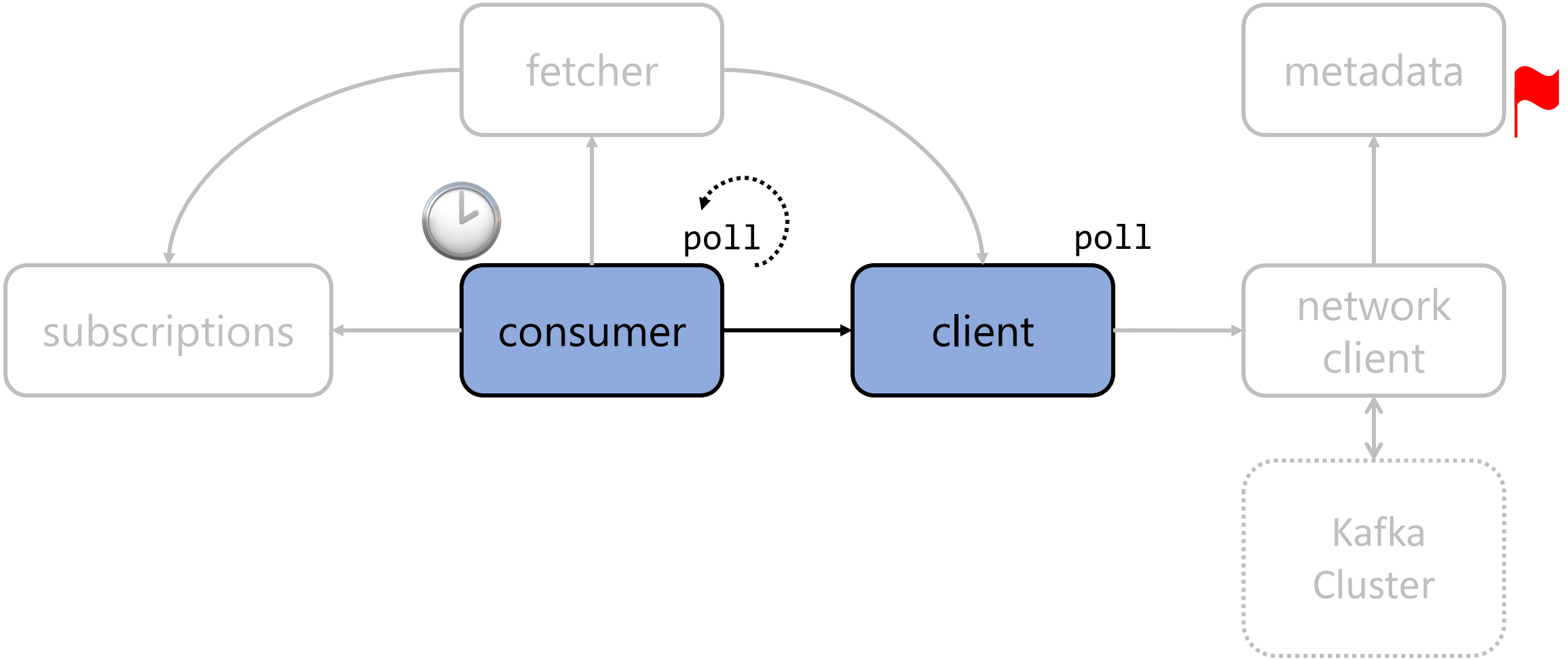



```
fetch.max.wait.ms
        = 500
```

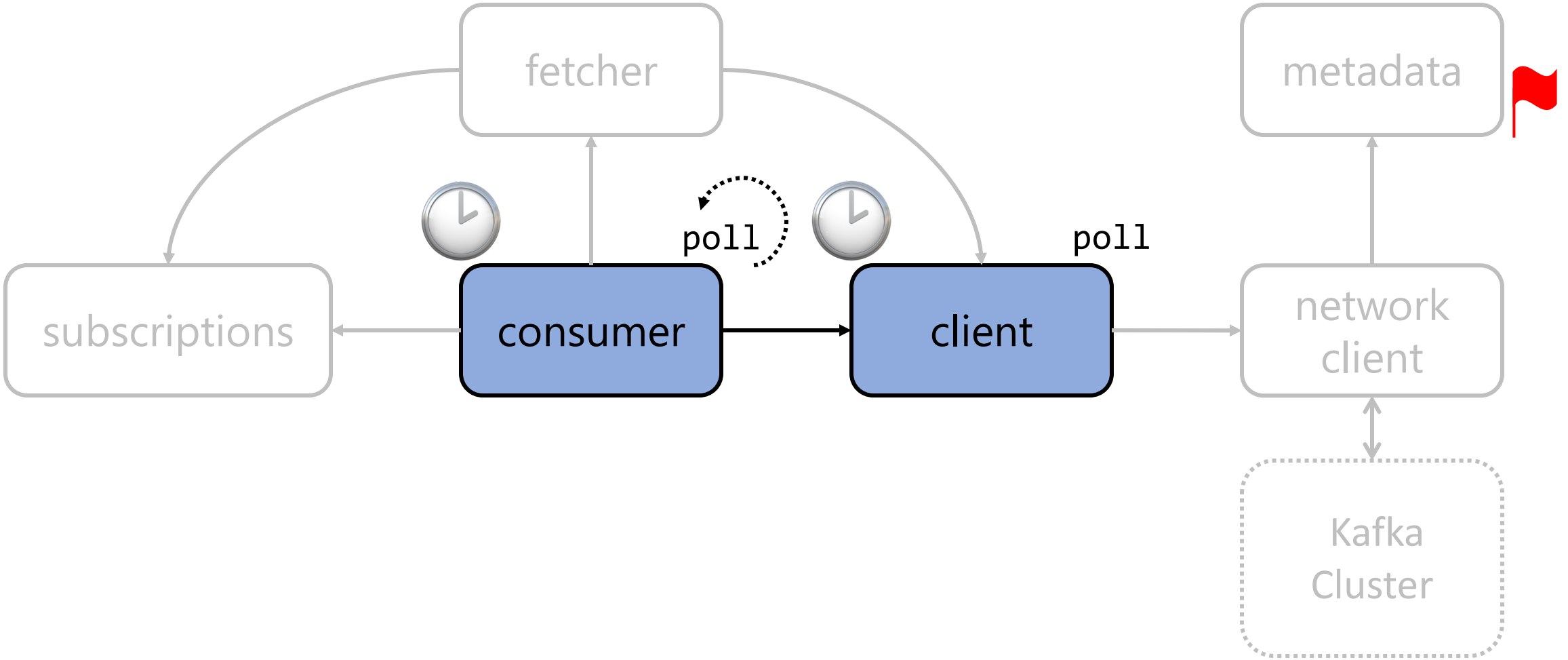
Consumer под капотом



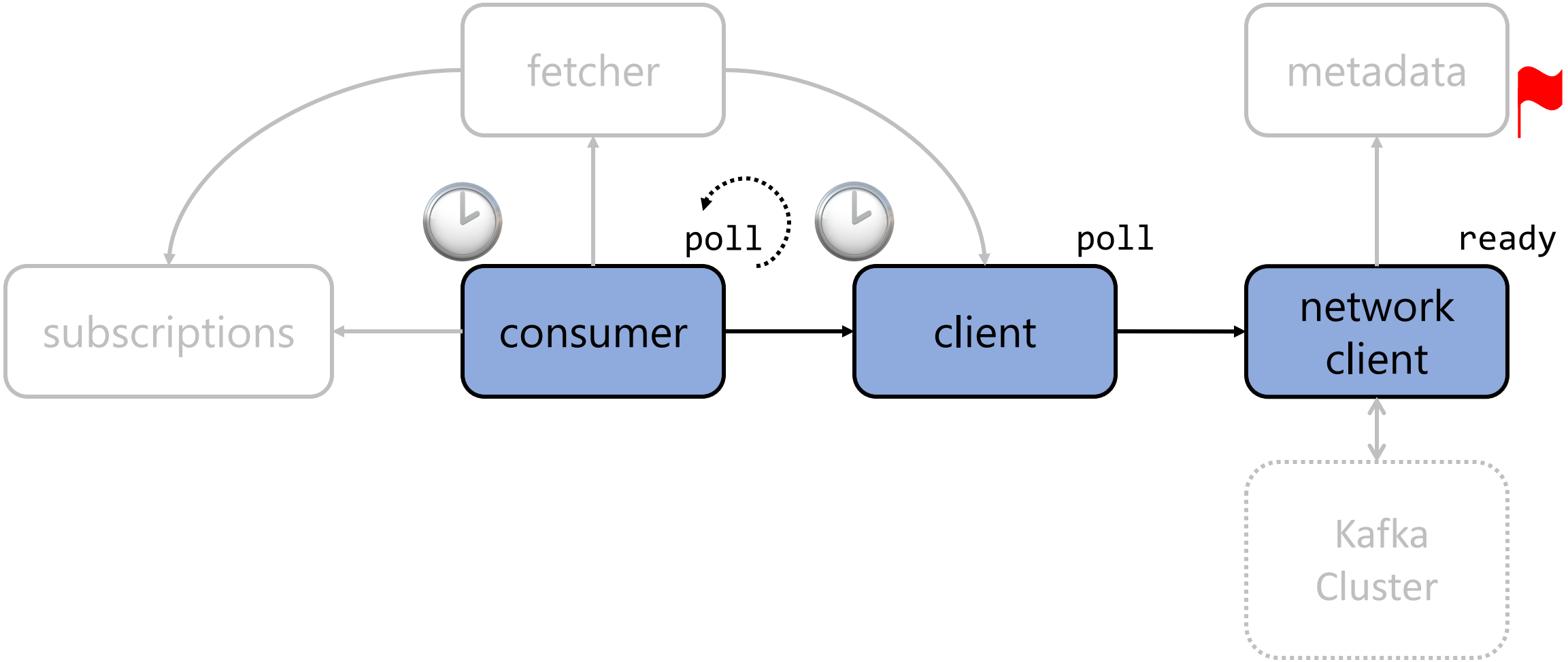
Consumer под капотом



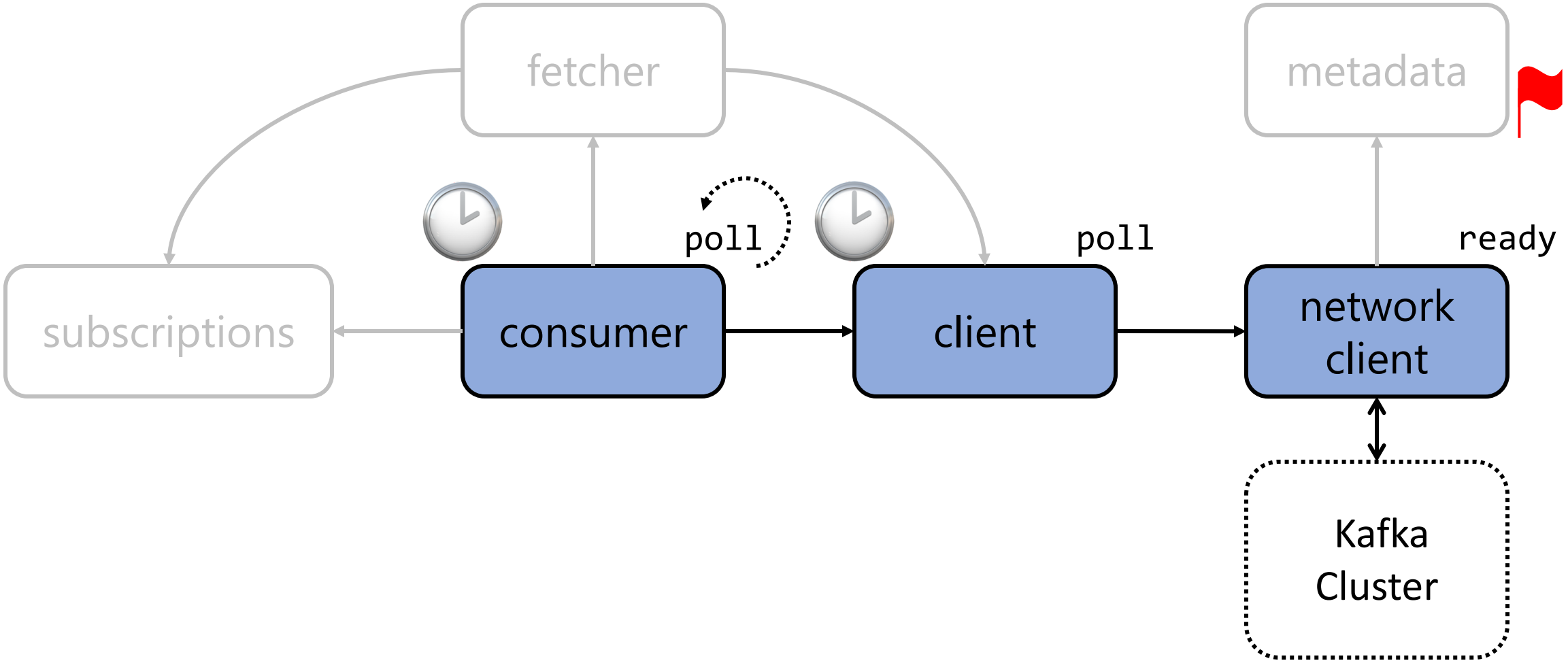
Consumer под капотом



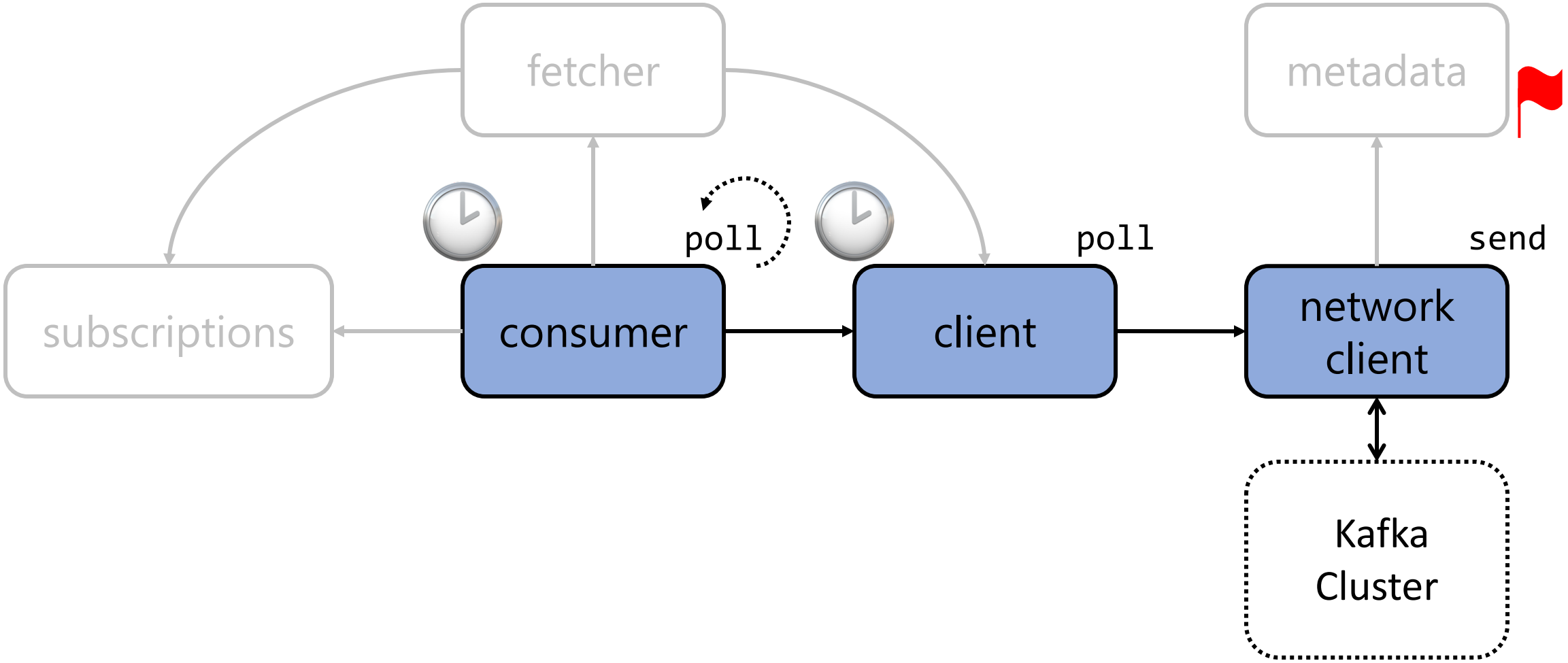
Consumer под капотом



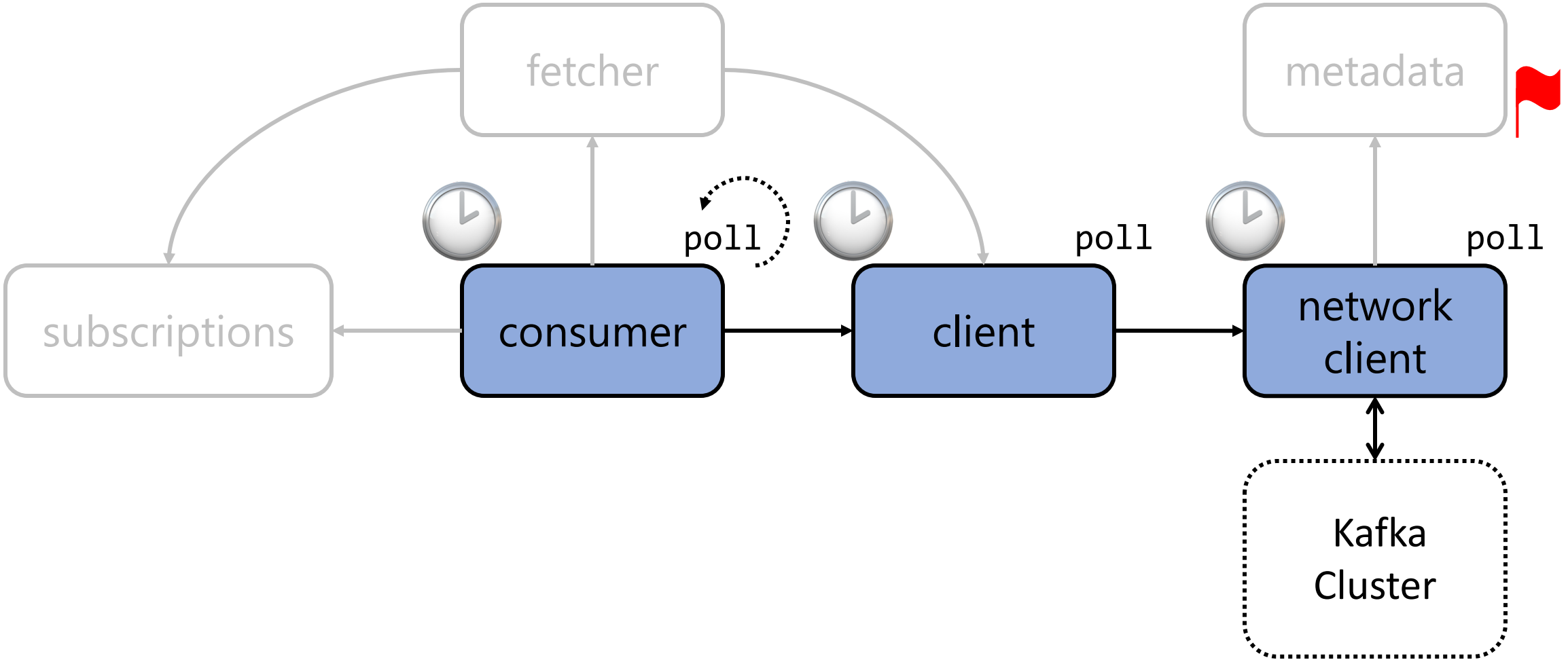
Consumer под капотом



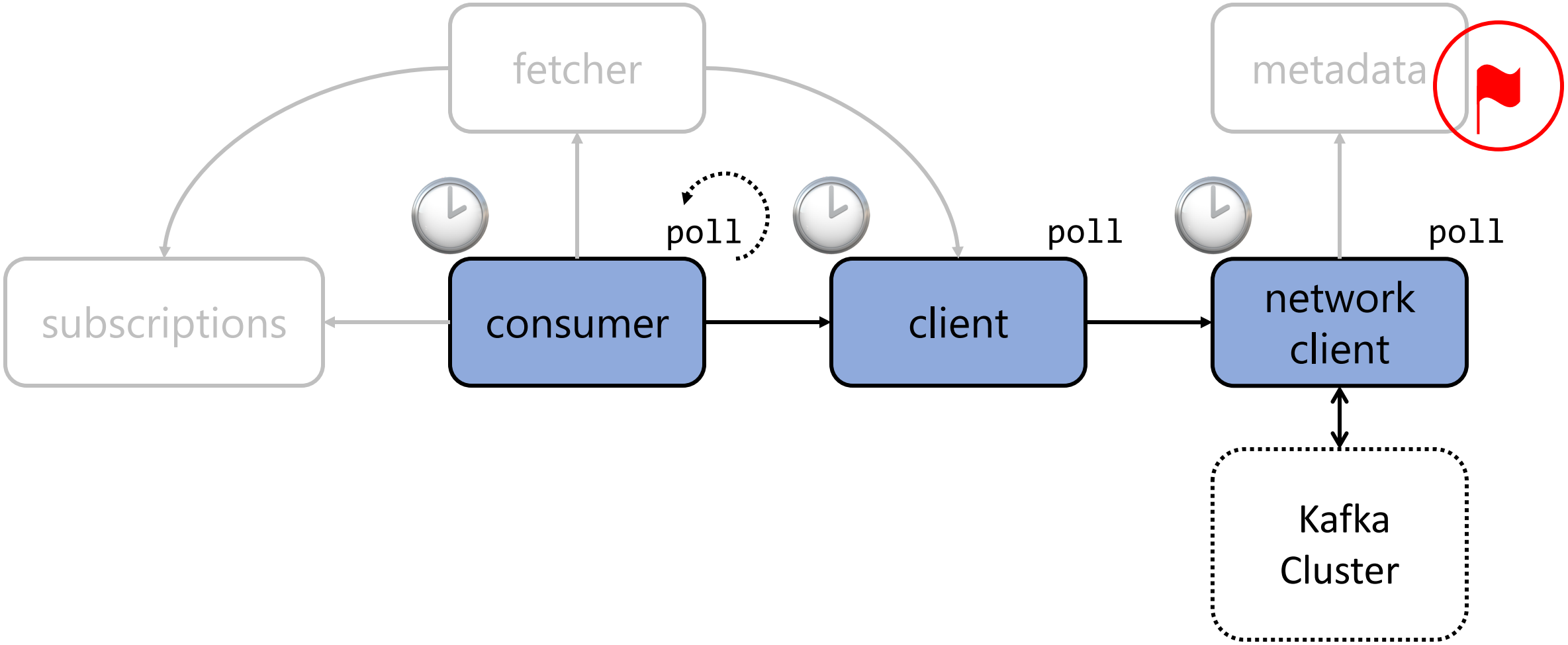
Consumer под капотом



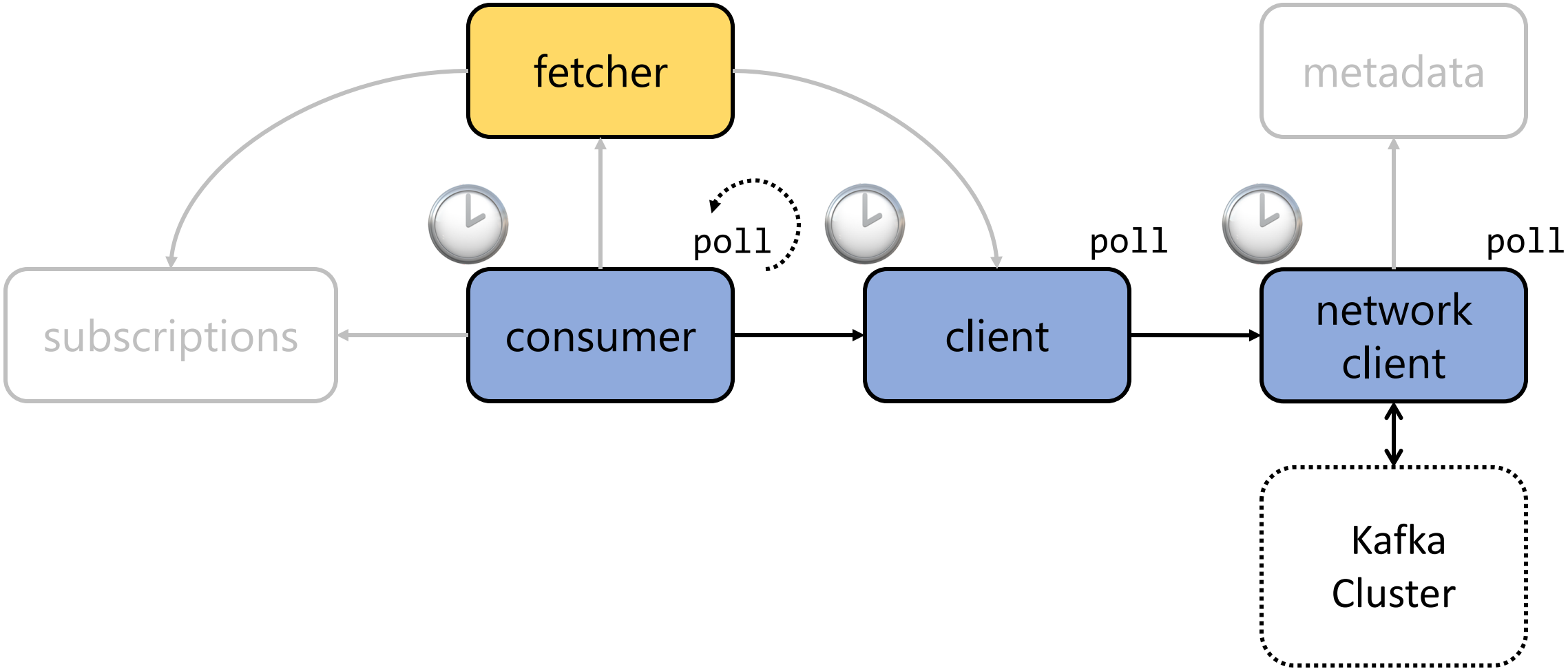
Consumer под капотом



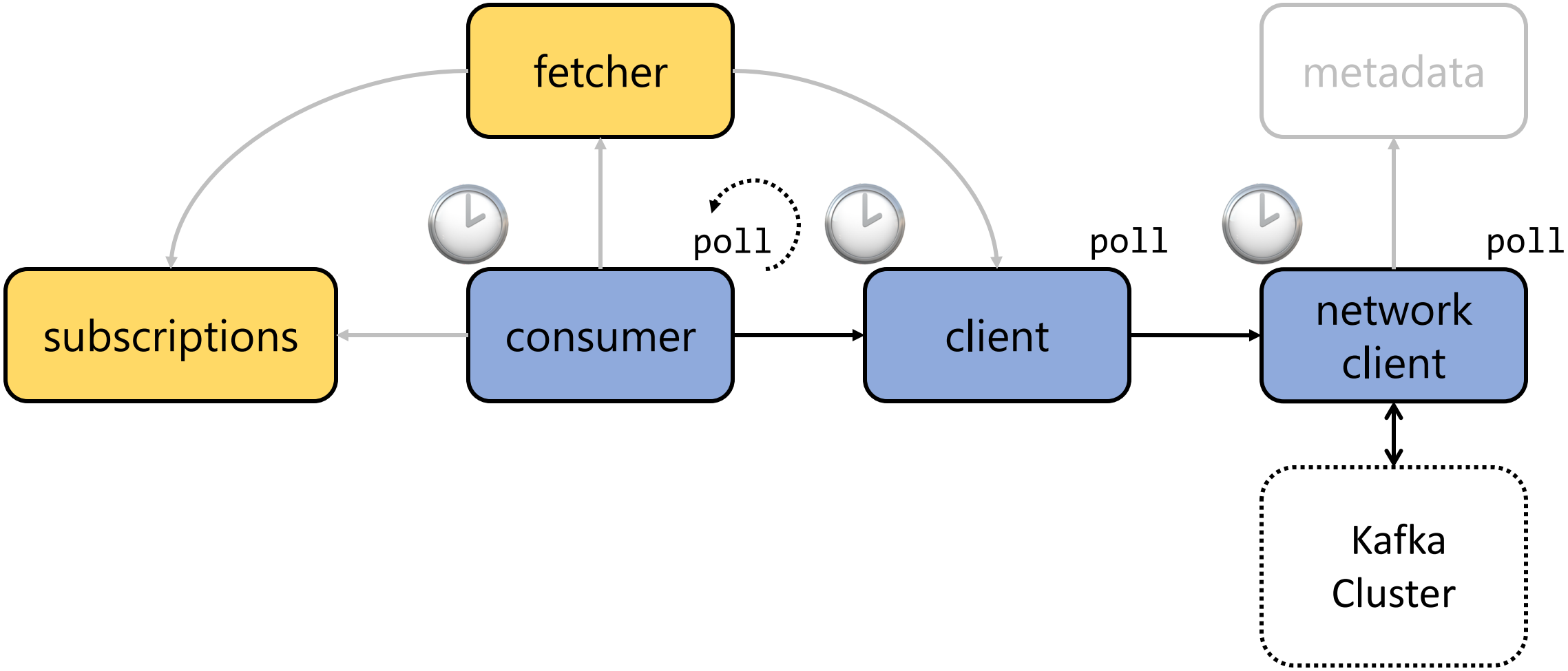
Consumer под капотом



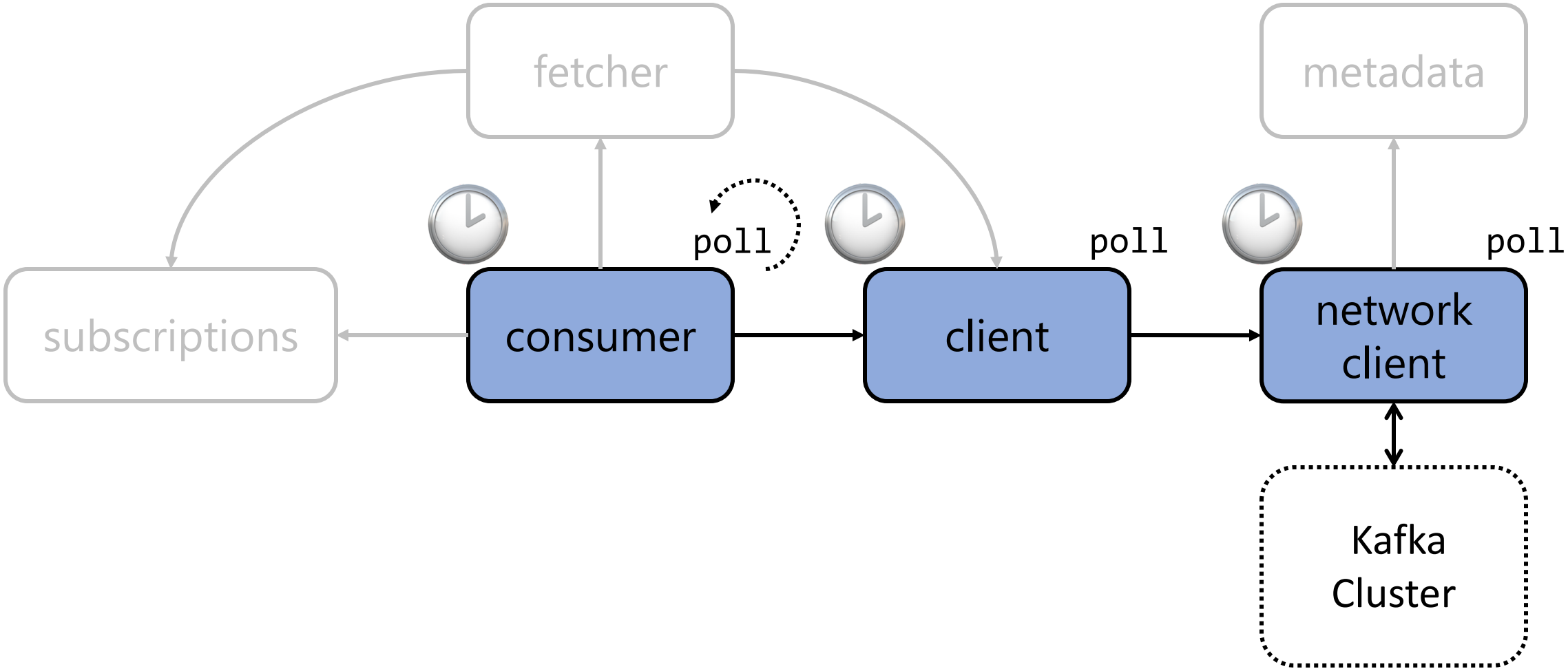
Consumer под капотом



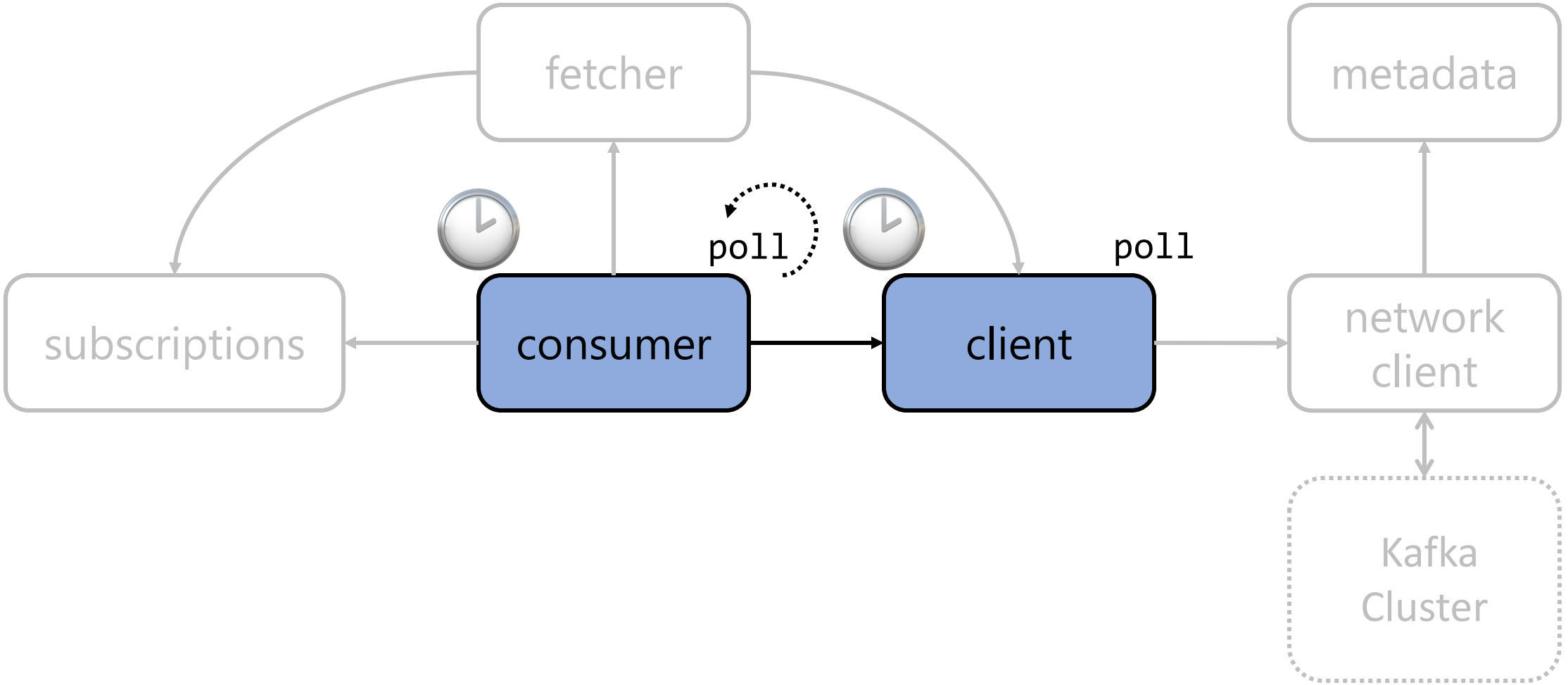
Consumer под капотом



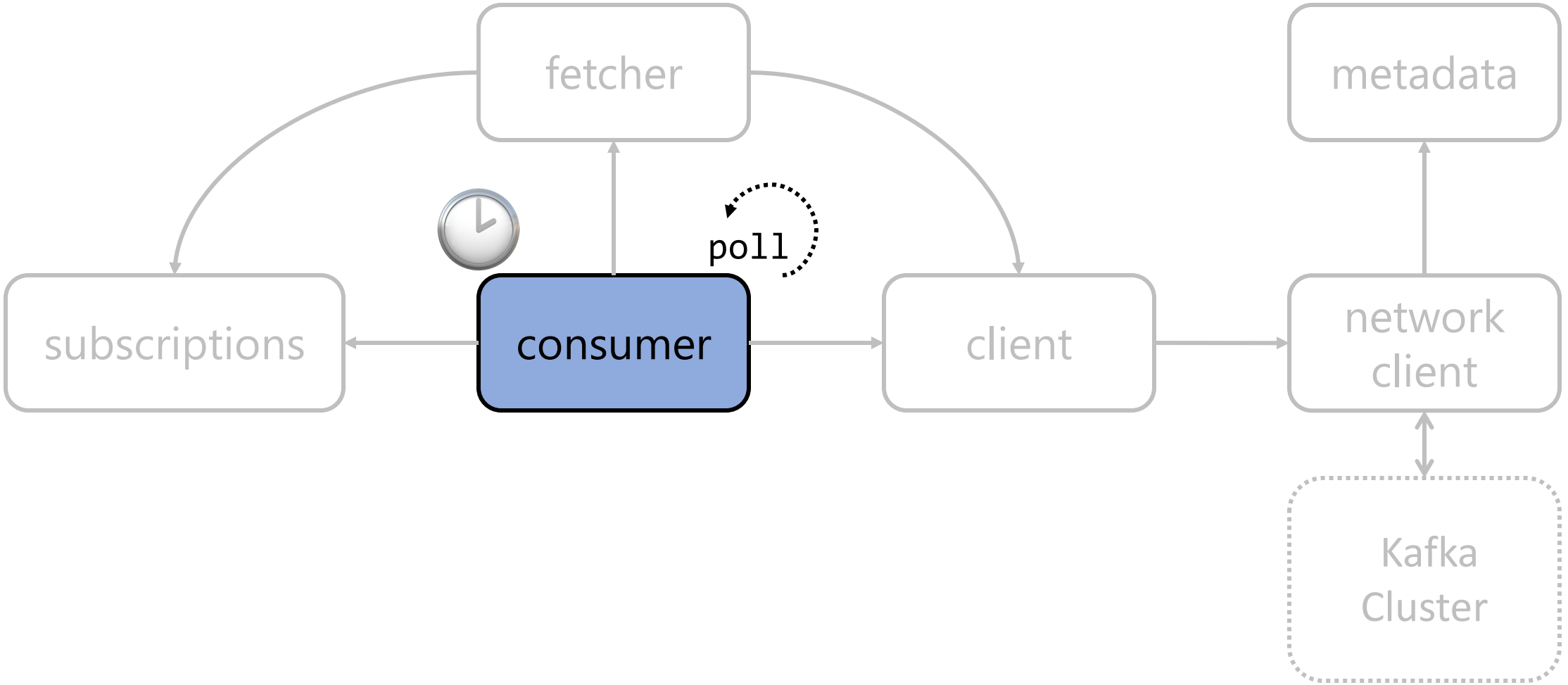
Consumer под капотом



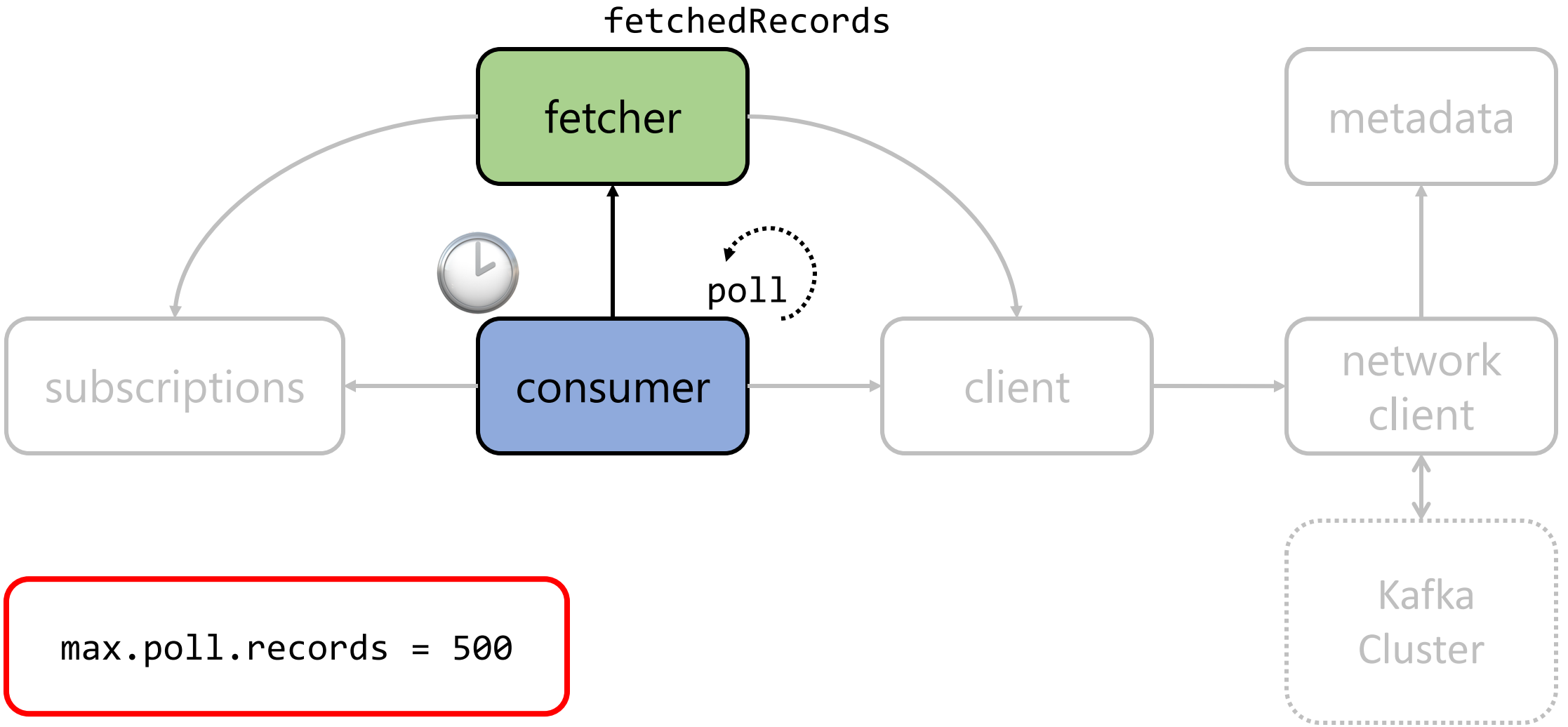
Consumer под капотом



Consumer под капотом

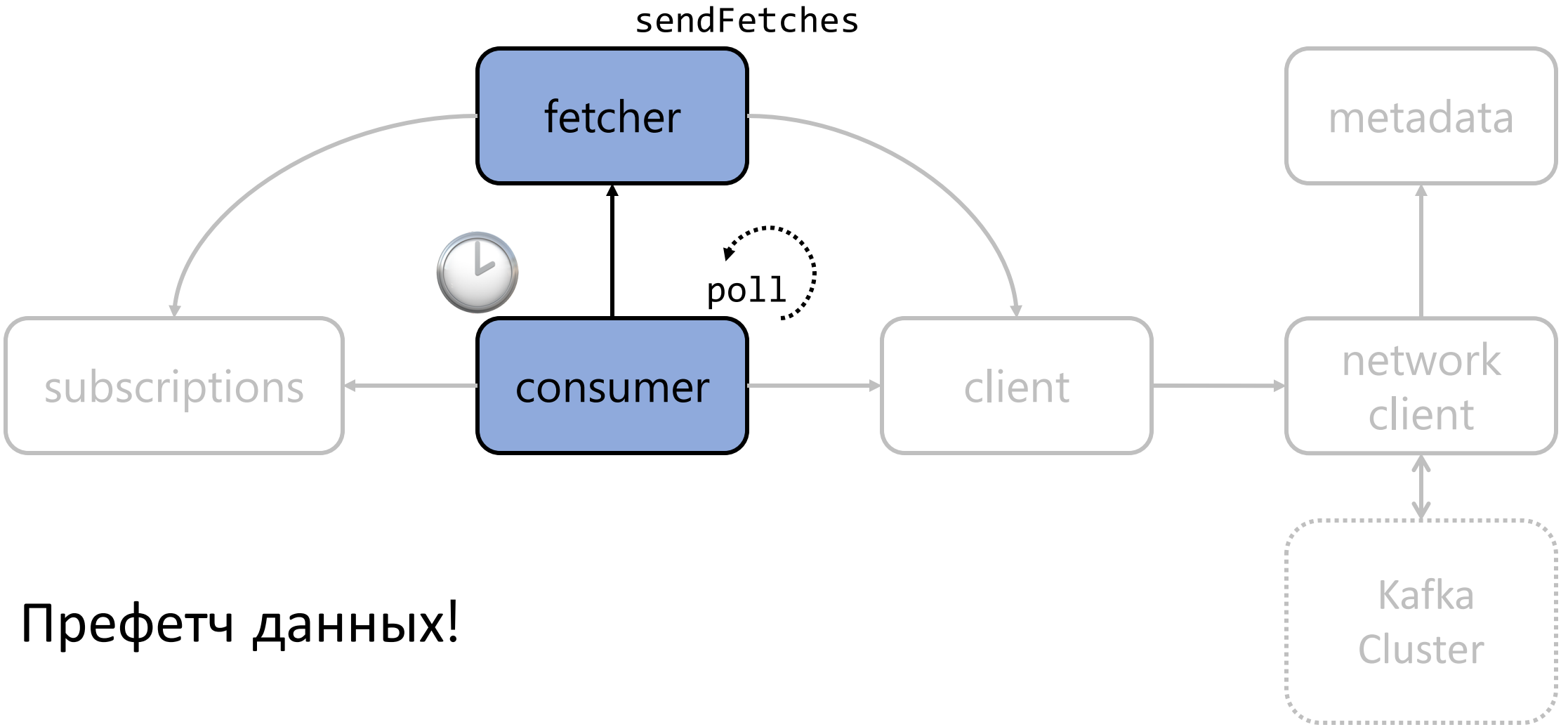


Consumer под капотом



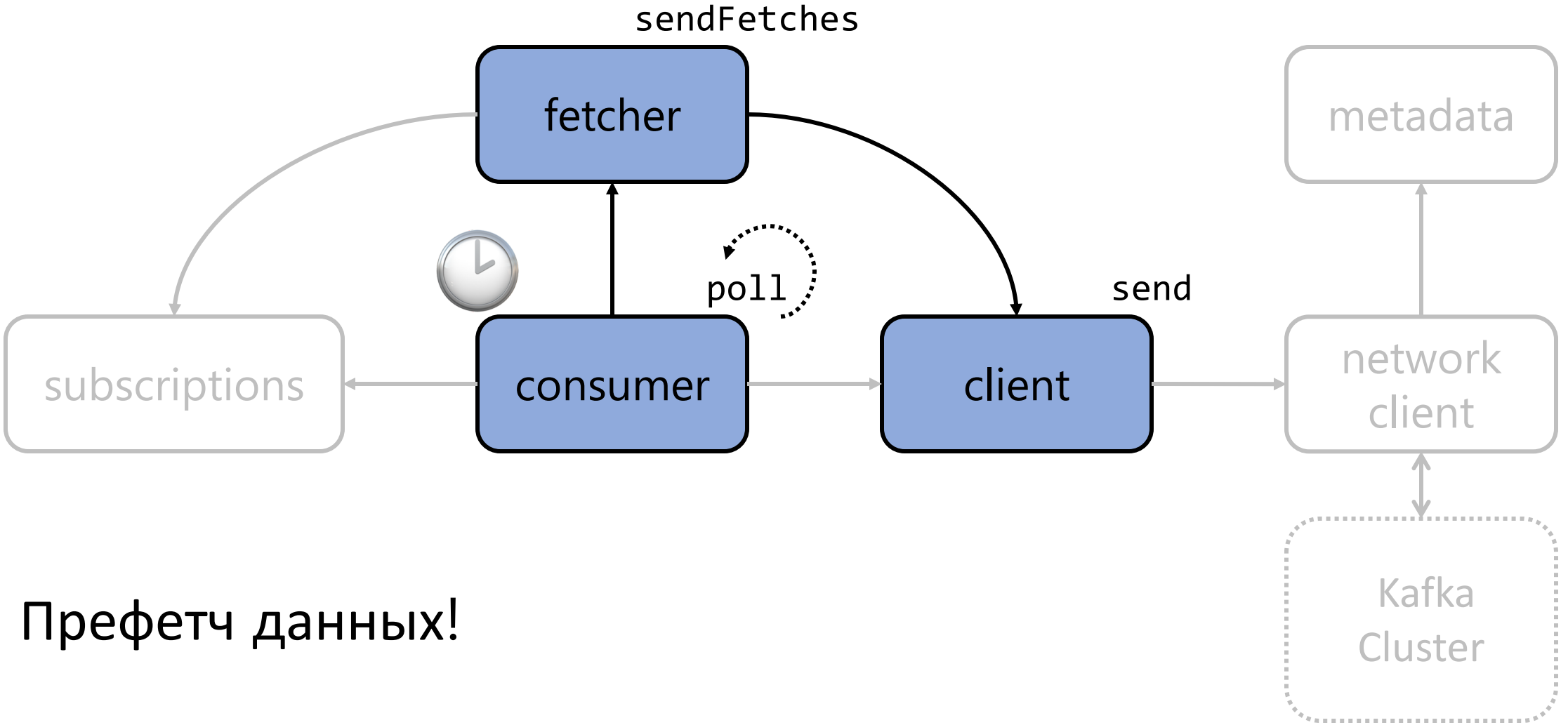

```
max.poll.records = 500
```

Consumer под капотом



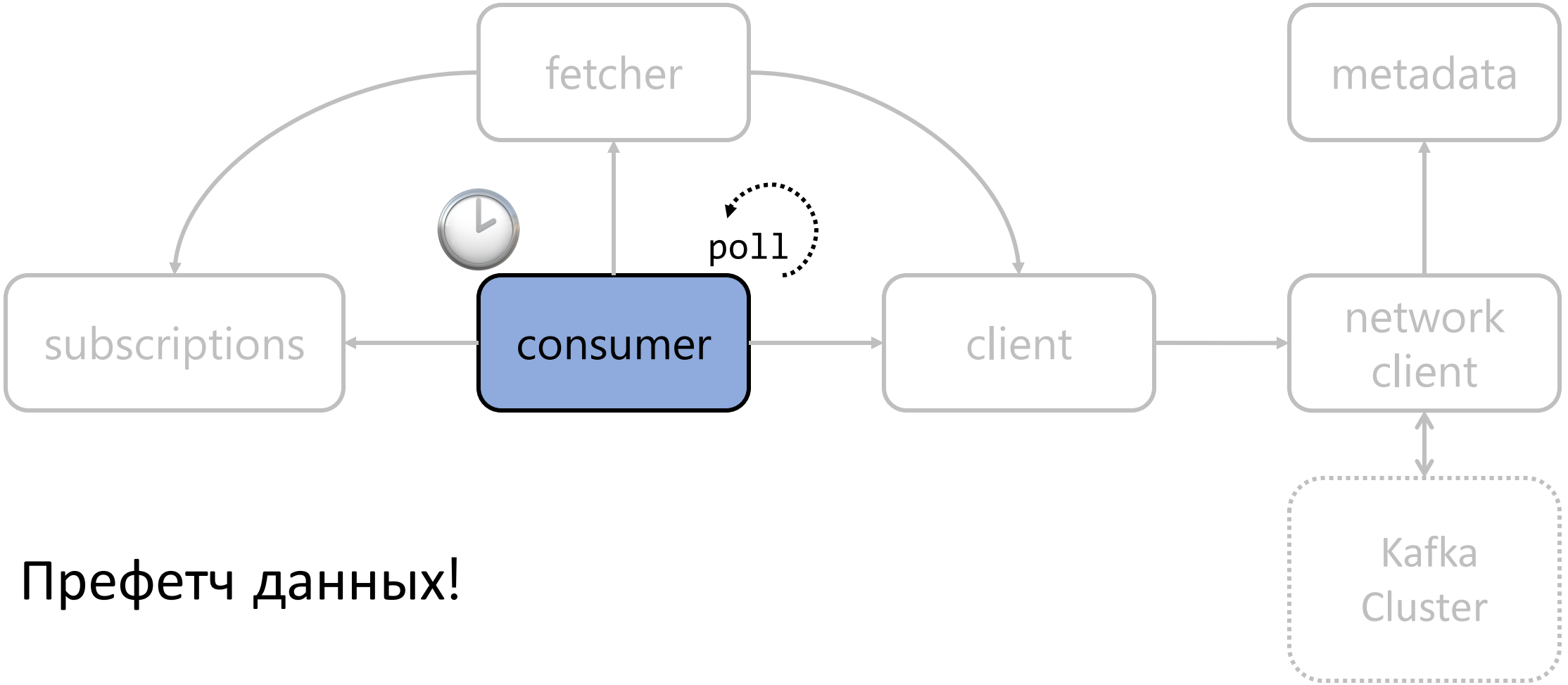
Префетч данных!

Consumer под капотом



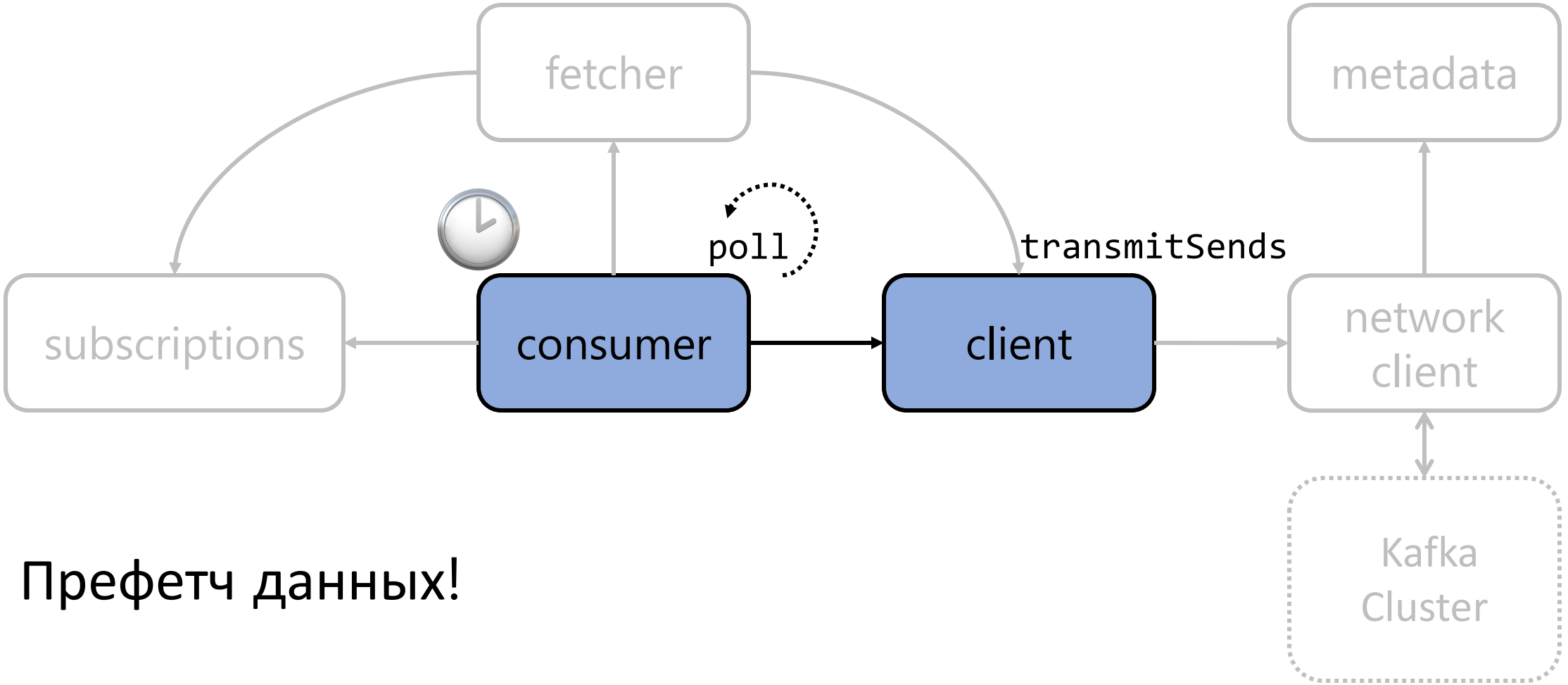
Префетч данных!

Consumer под капотом



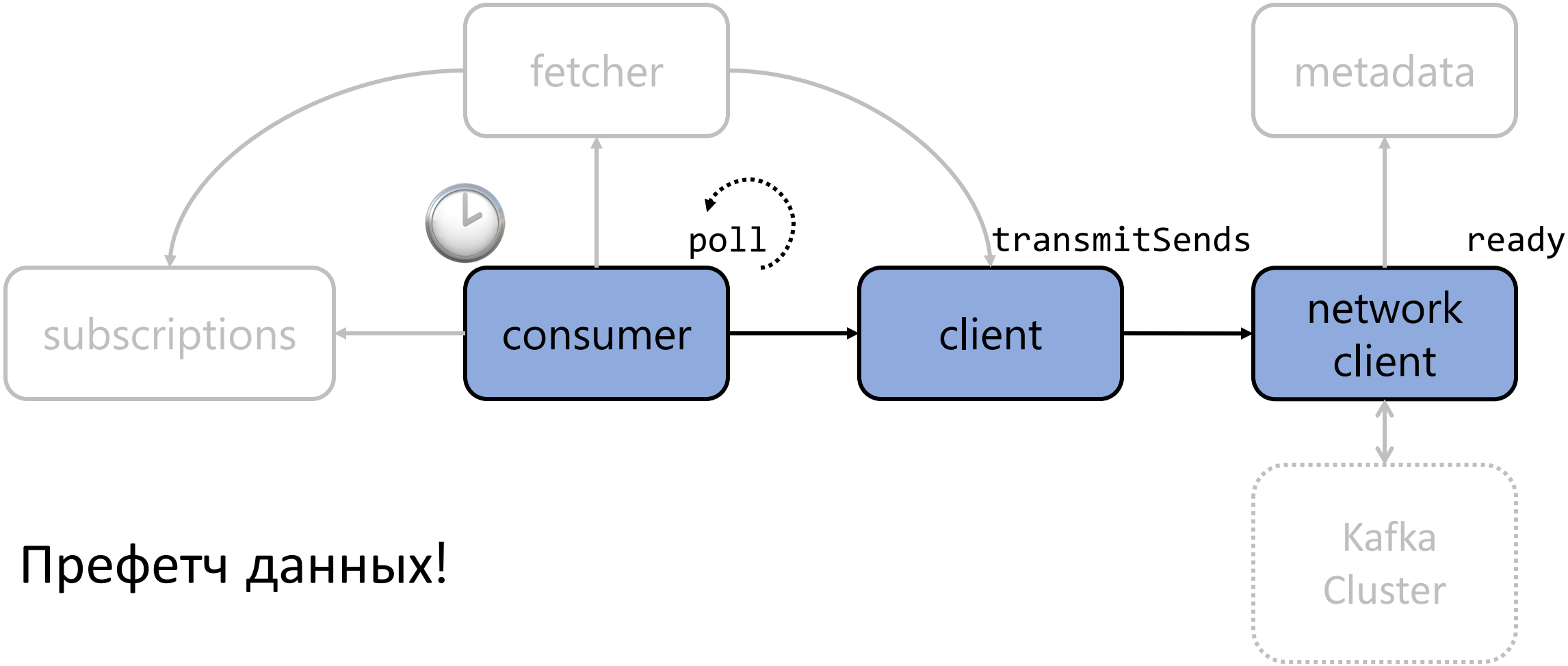
Префетч данных!

Consumer под капотом



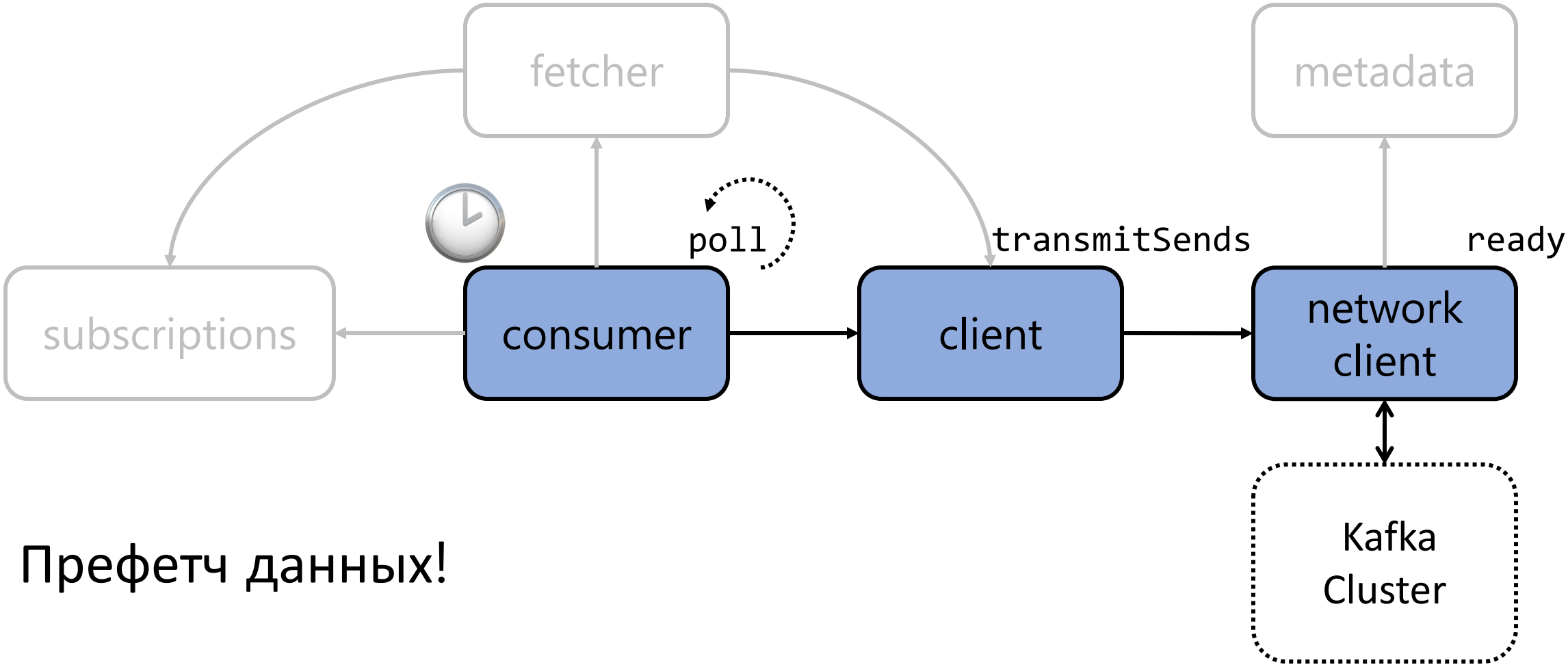
Префетч данных!

Consumer под капотом



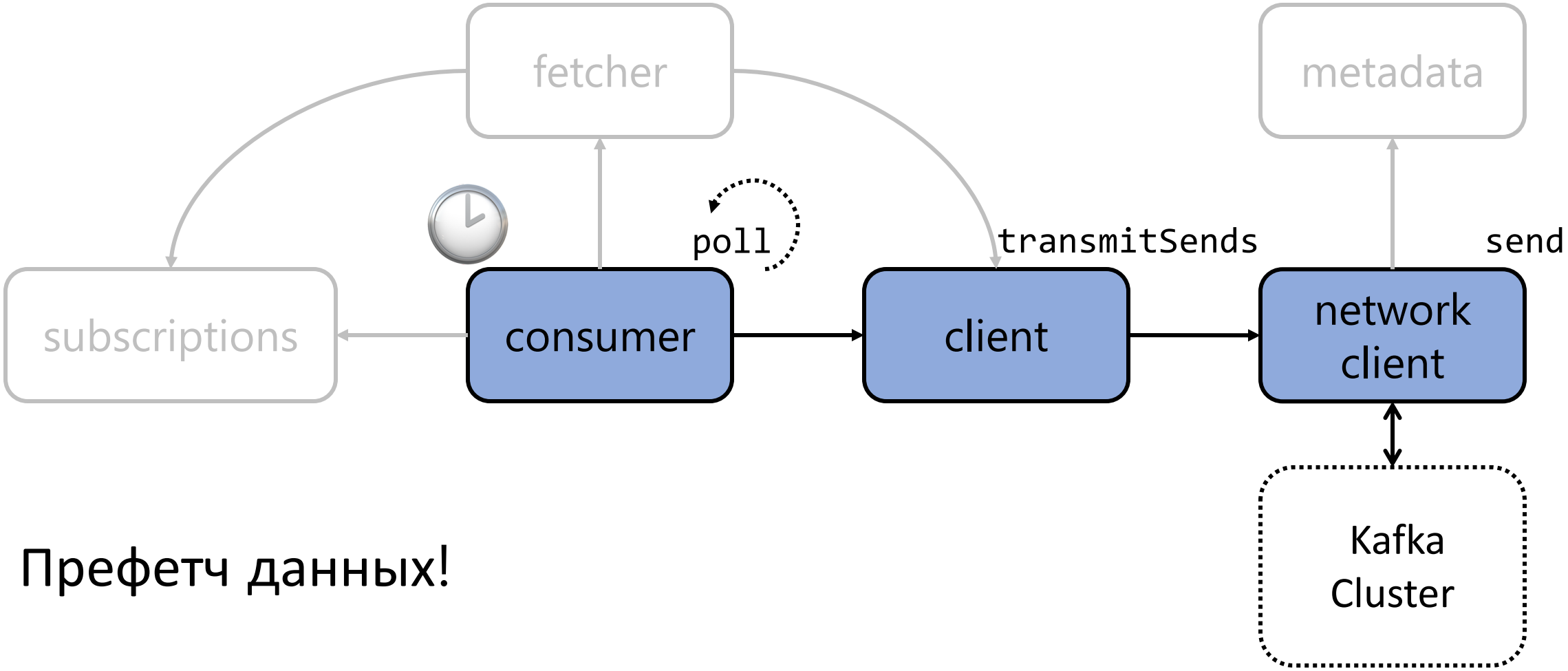
Префетч данных!

Consumer под капотом



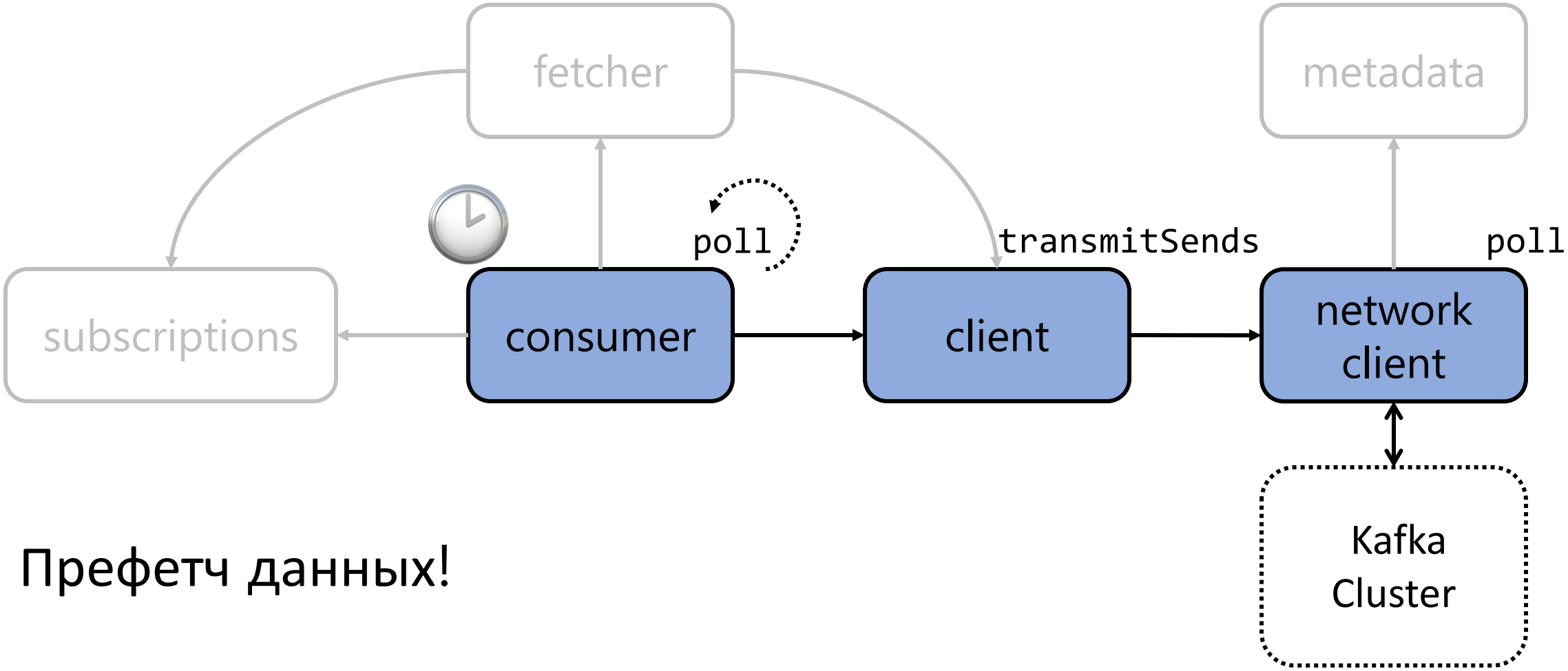
Префетч данных!

Consumer под капотом



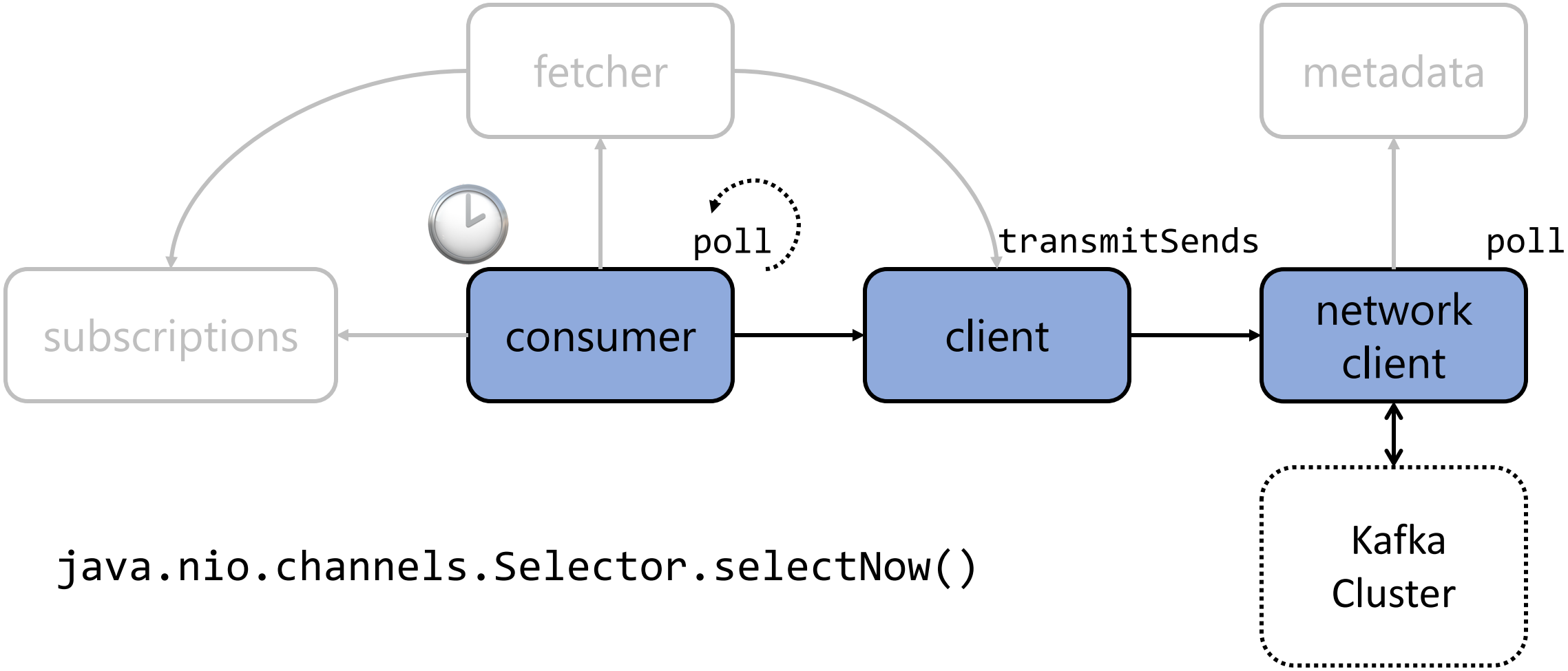
Префетч данных!

Consumer под капотом



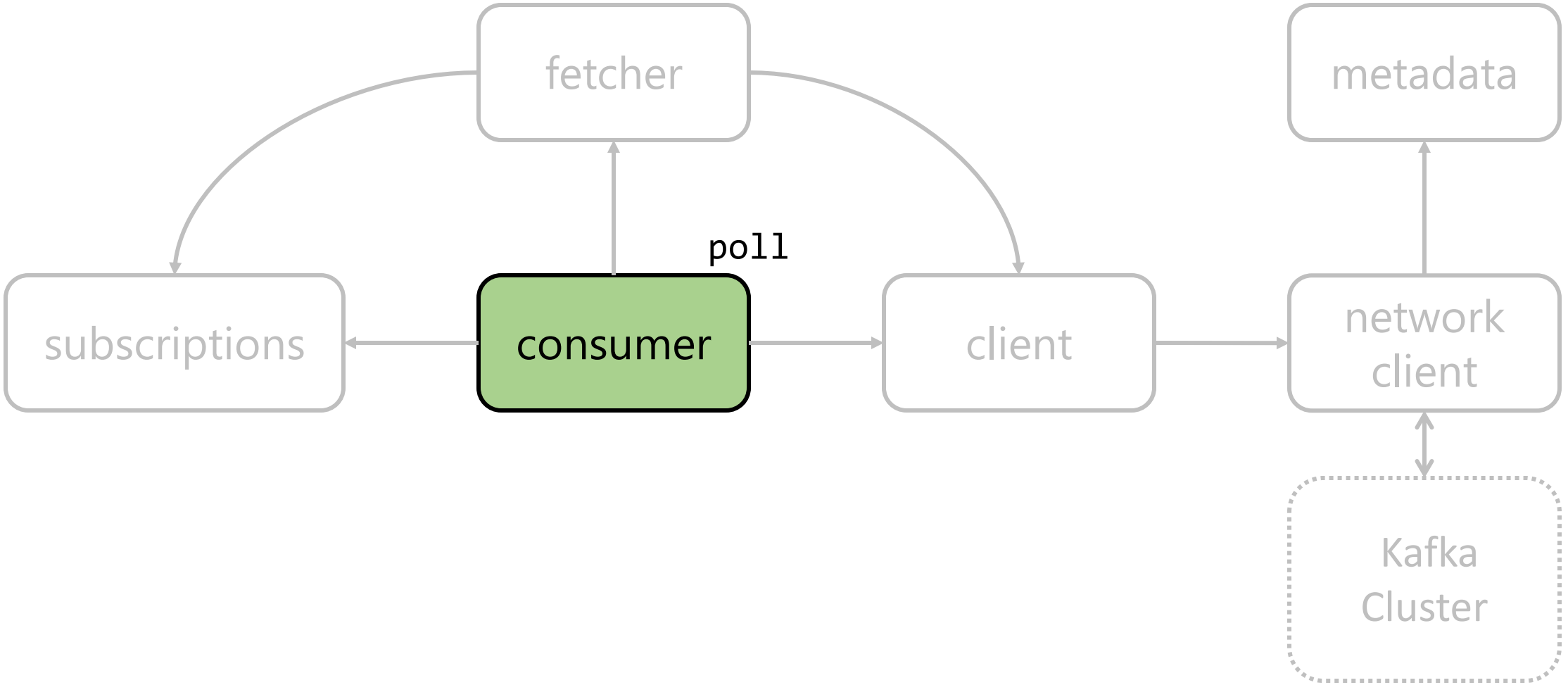
Префетч данных!

Consumer под капотом

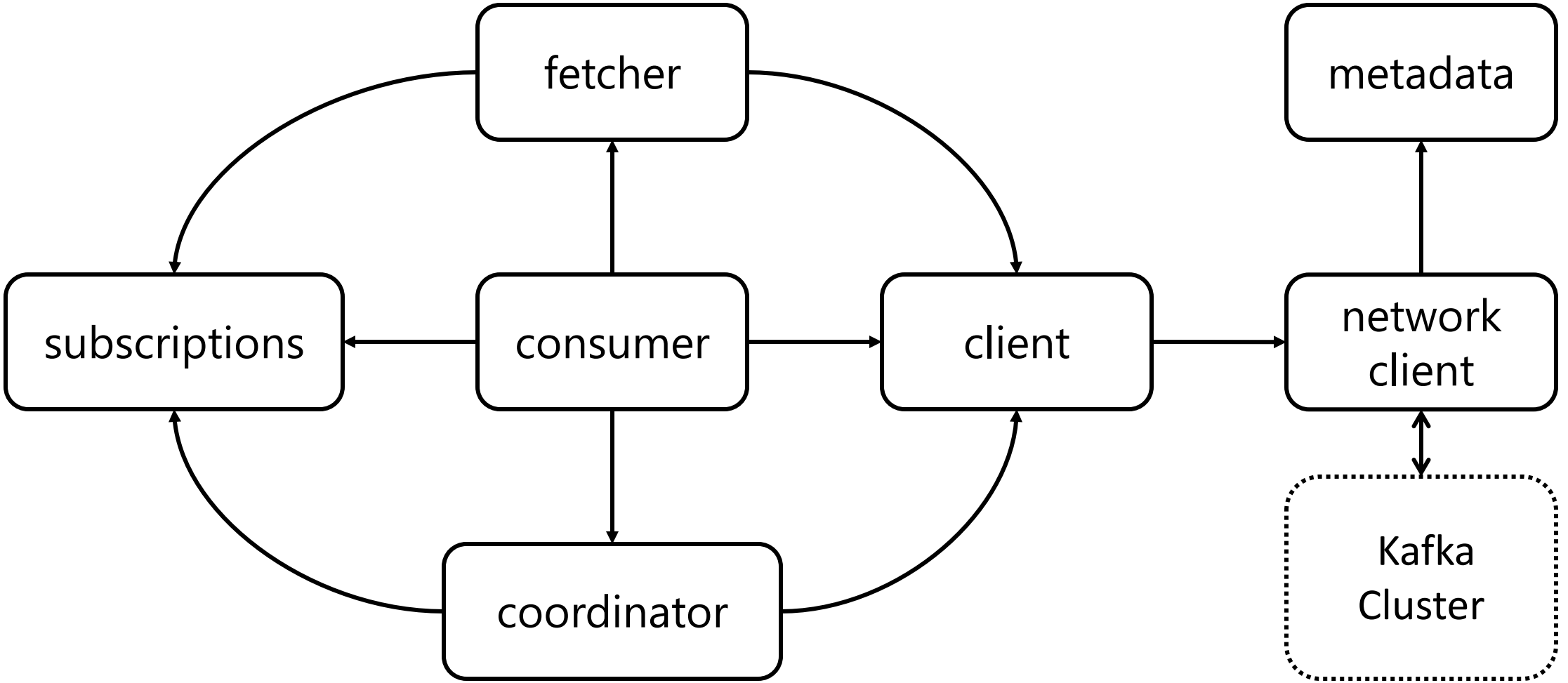



```
java.nio.channels.Selector.selectNow()
```

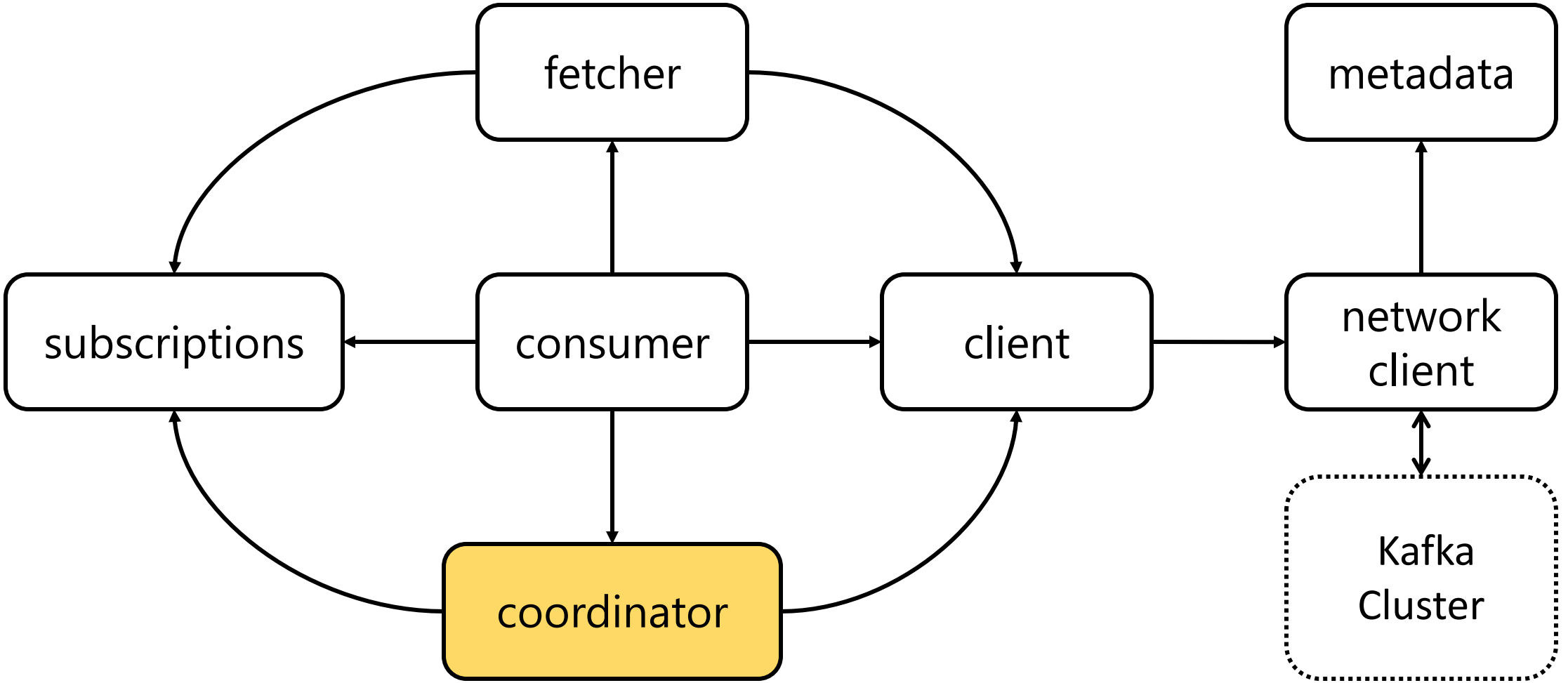
Consumer под капотом



Consumer под капотом — coordinator



Consumer под капотом — coordinator



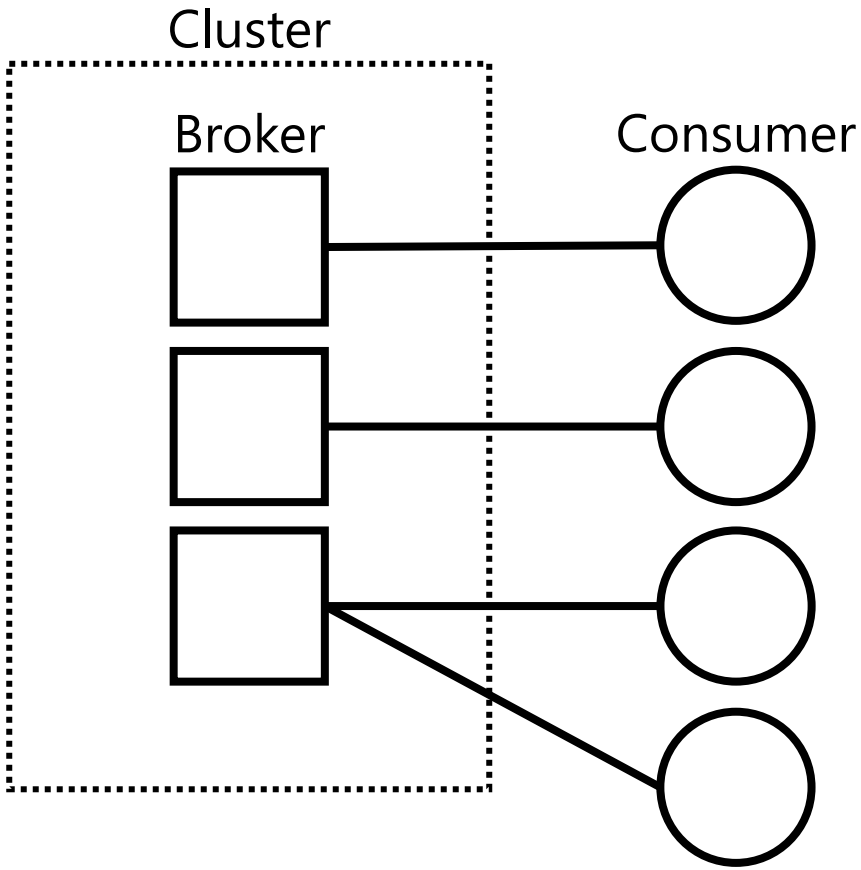
Consumer под капотом — coordinator

Consumer под капотом — coordinator

lookup
Group
Coordinator

Consumer под капотом — coordinator

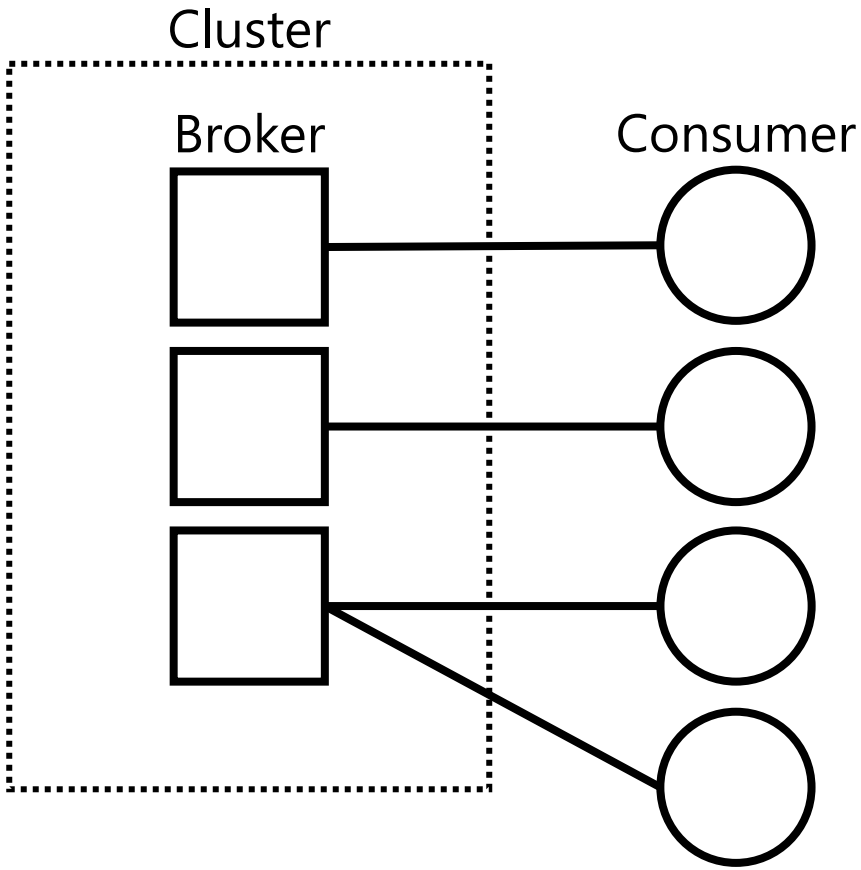
lookup
Group
Coordinator



`group.id`

Consumer под капотом — coordinator

lookup
Group
Coordinator

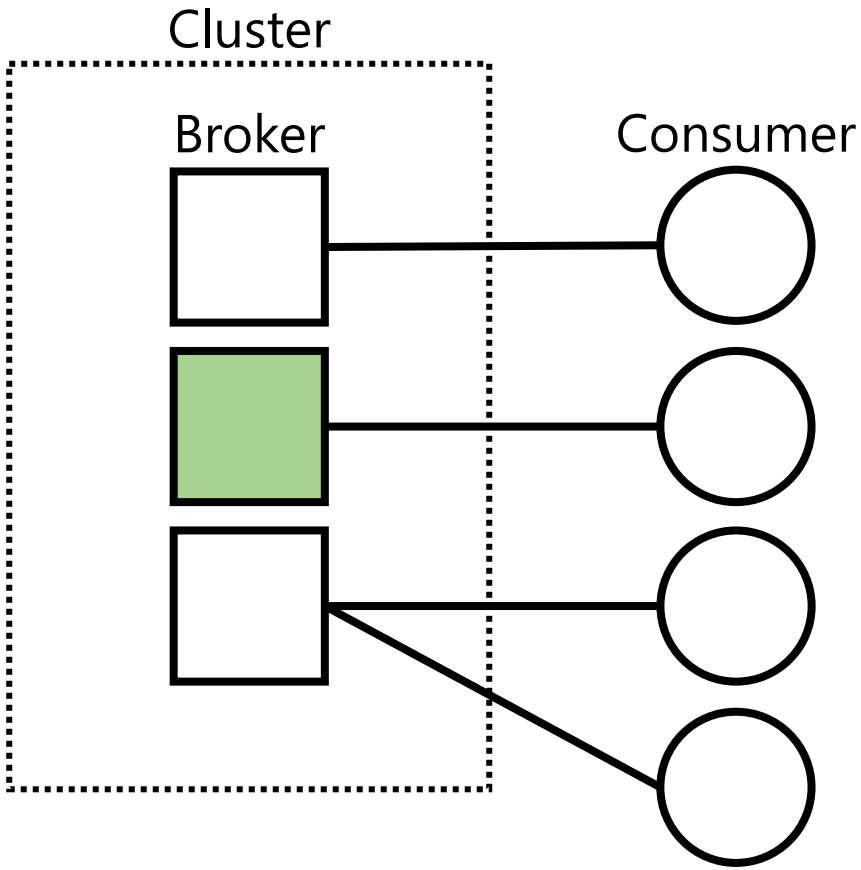


`group.id`

`__consumer_offsets`

Consumer под капотом — coordinator

lookup
Group
Coordinator

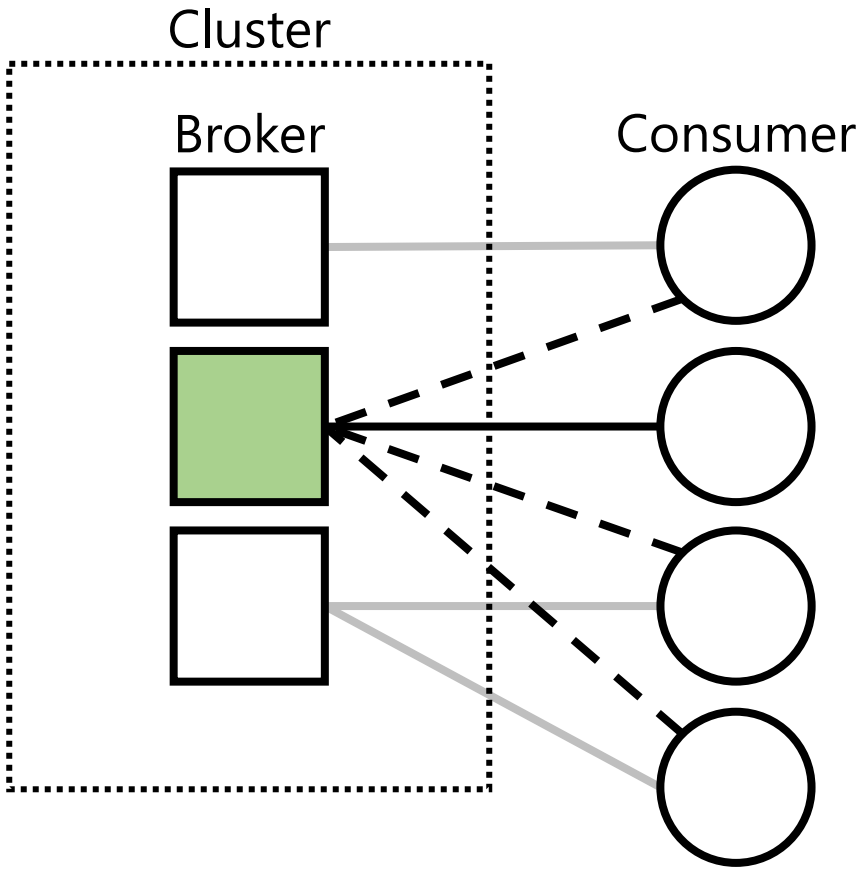


group.id

__consumer_offsets

Consumer под капотом — coordinator

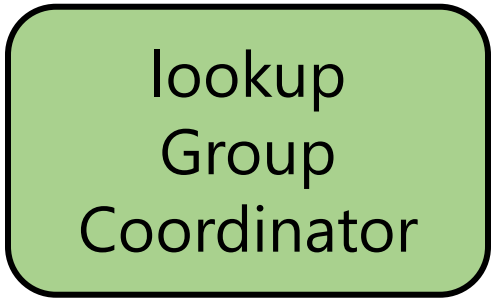
lookup
Group
Coordinator



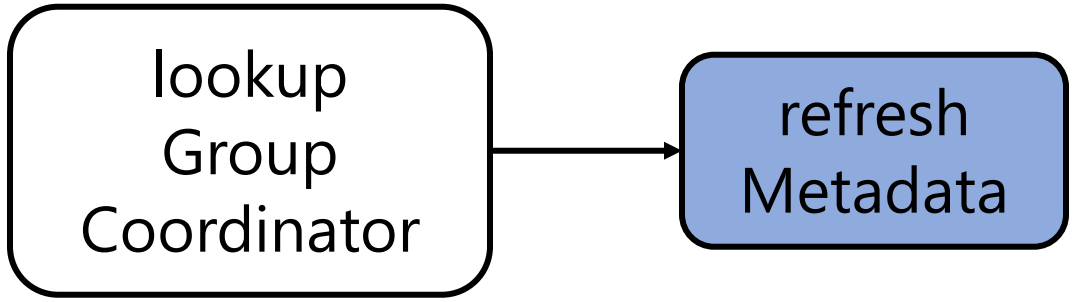
group.id

__consumer_offsets

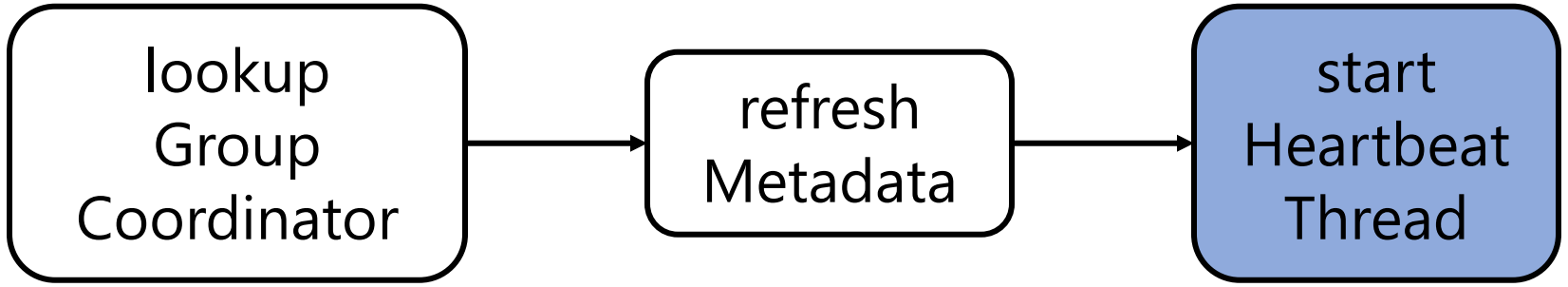
Consumer под капотом — coordinator



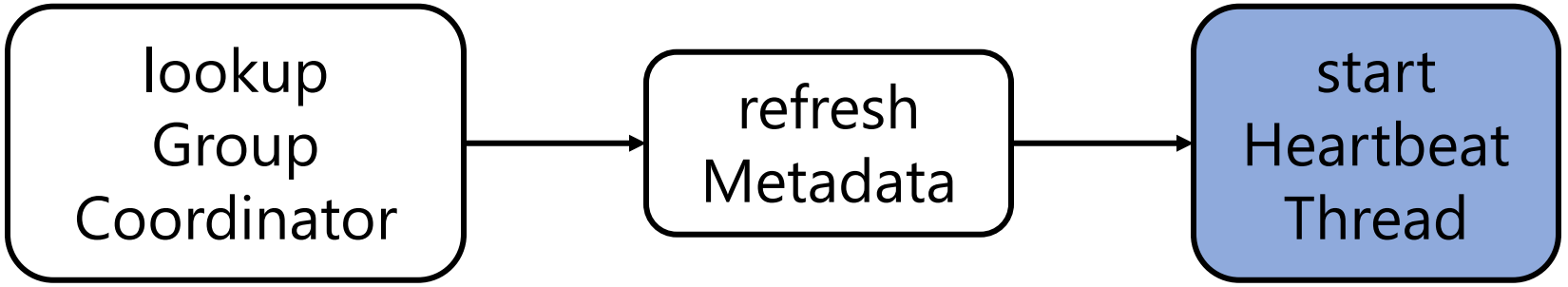
Consumer под капотом — coordinator

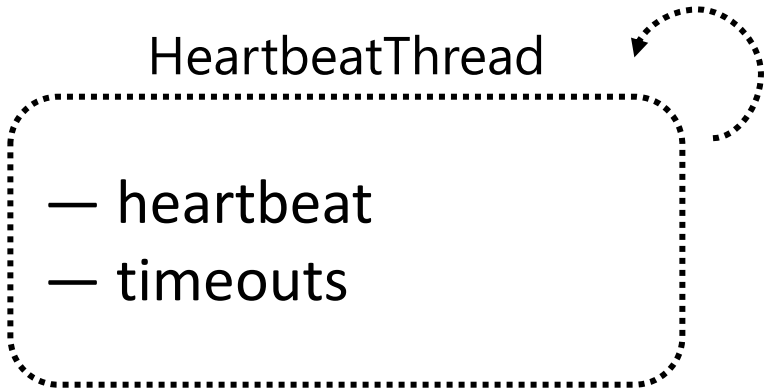


Consumer под капотом — coordinator

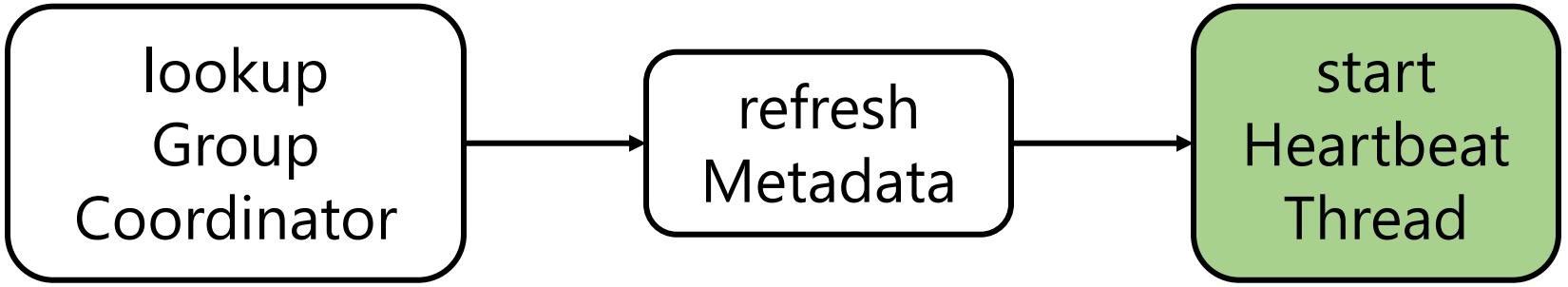


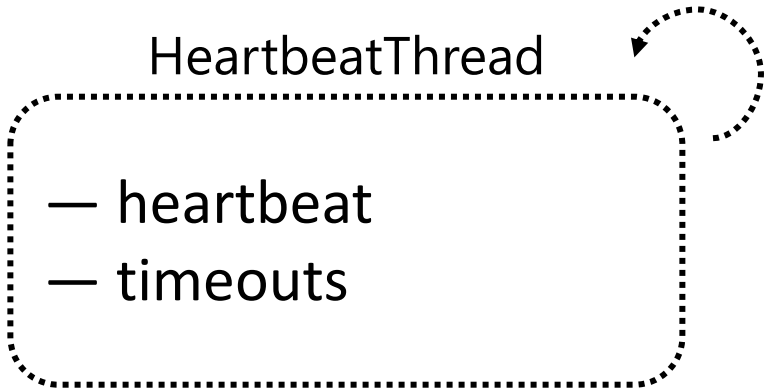
Consumer под капотом — coordinator



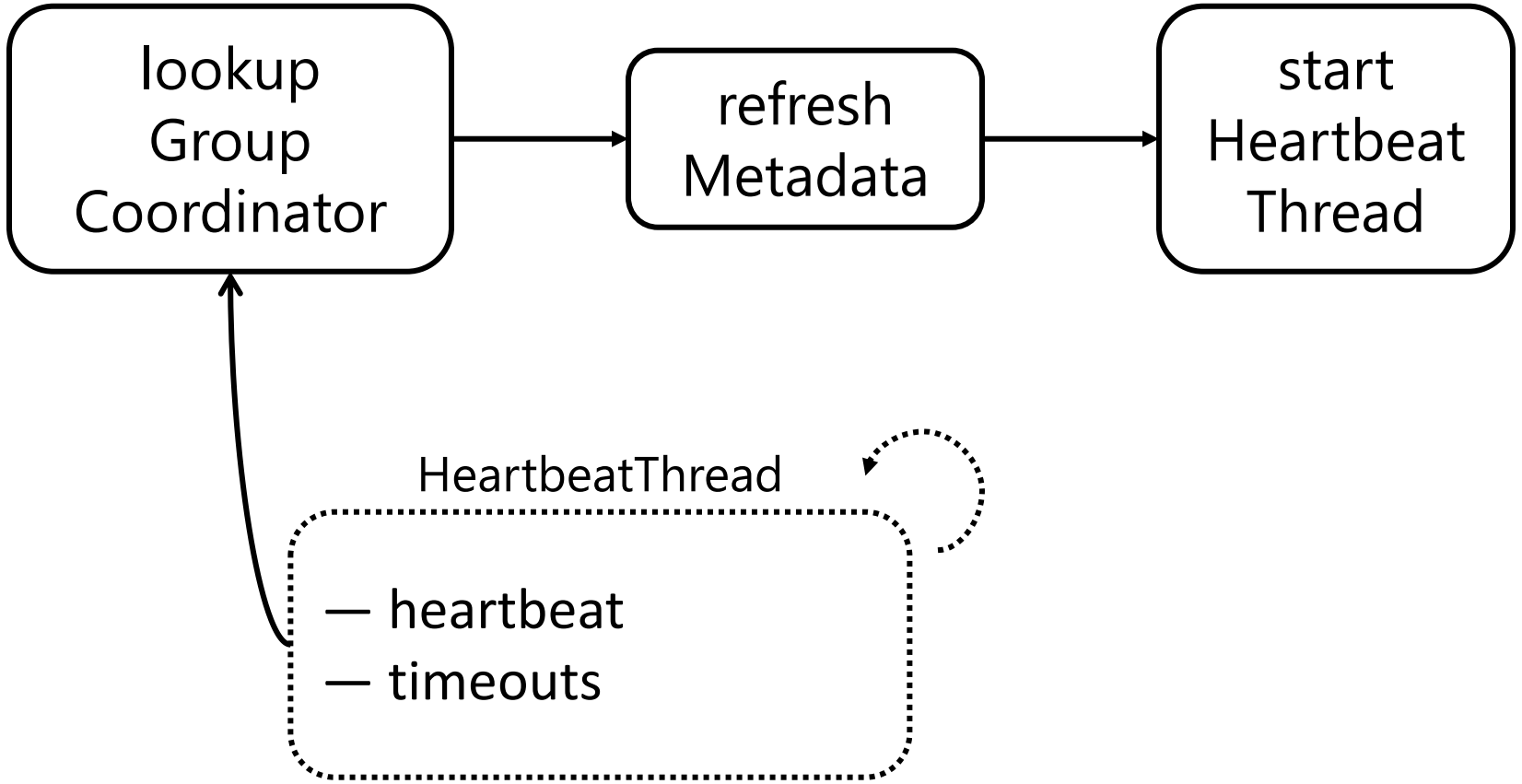


Consumer под капотом — coordinator

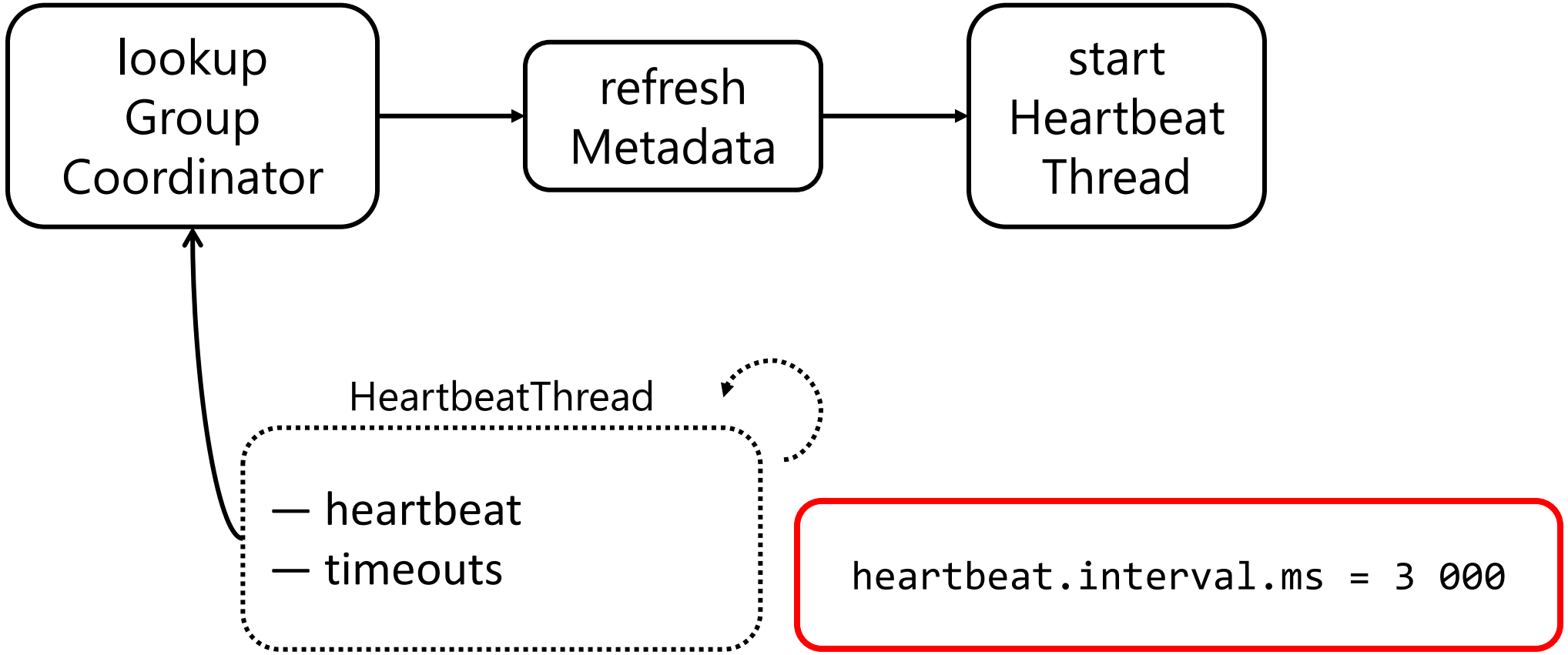




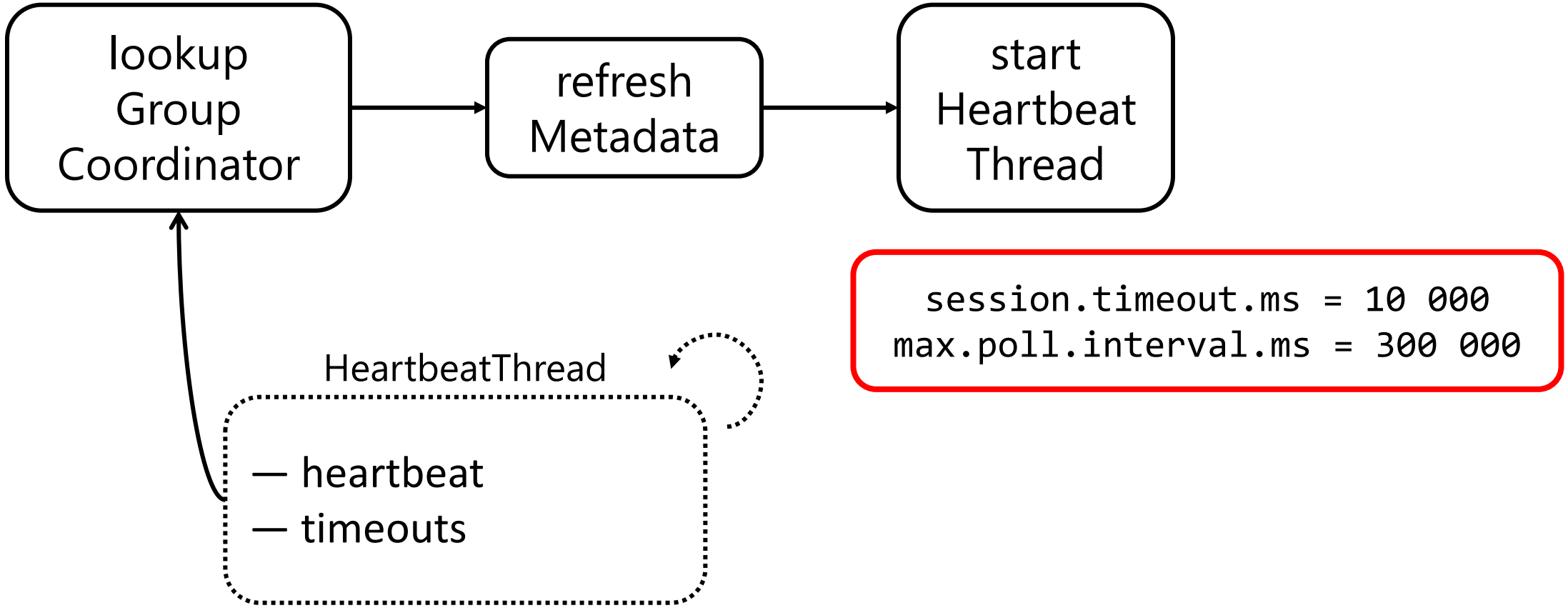
Consumer под капотом — coordinator



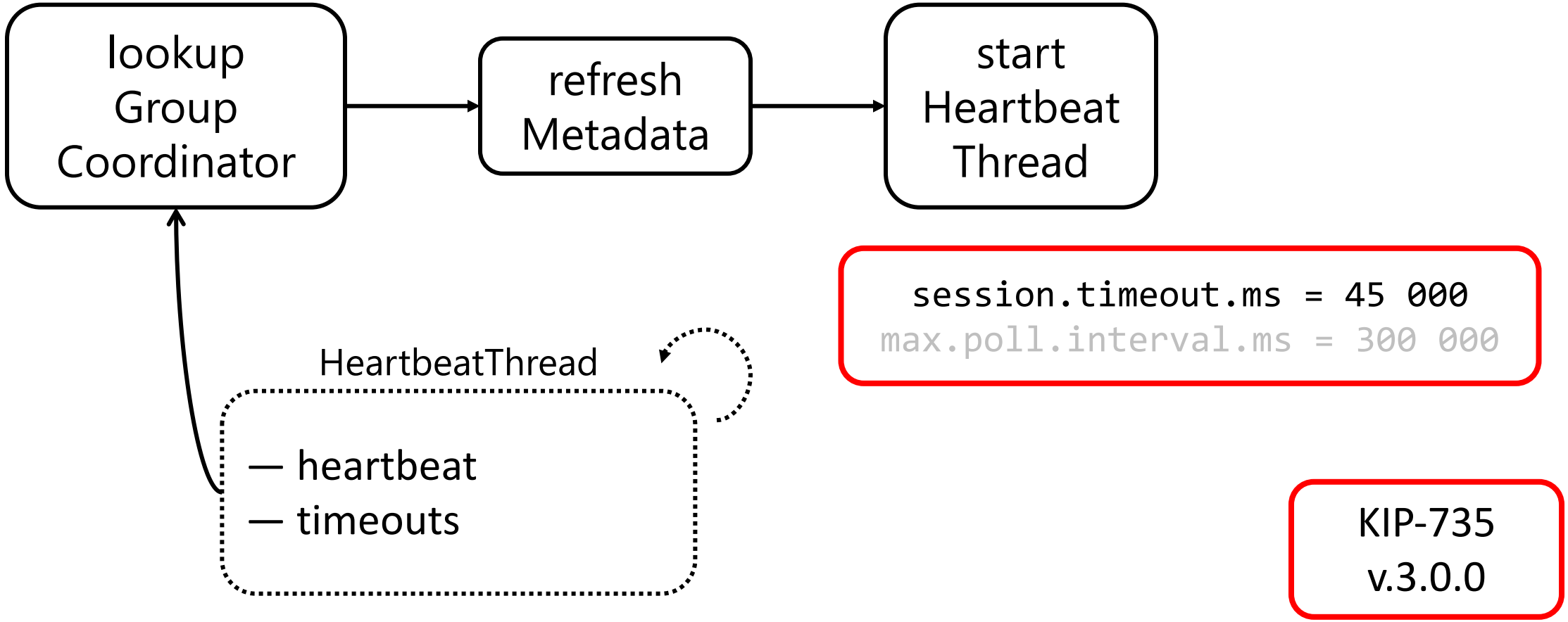
Consumer под капотом — coordinator



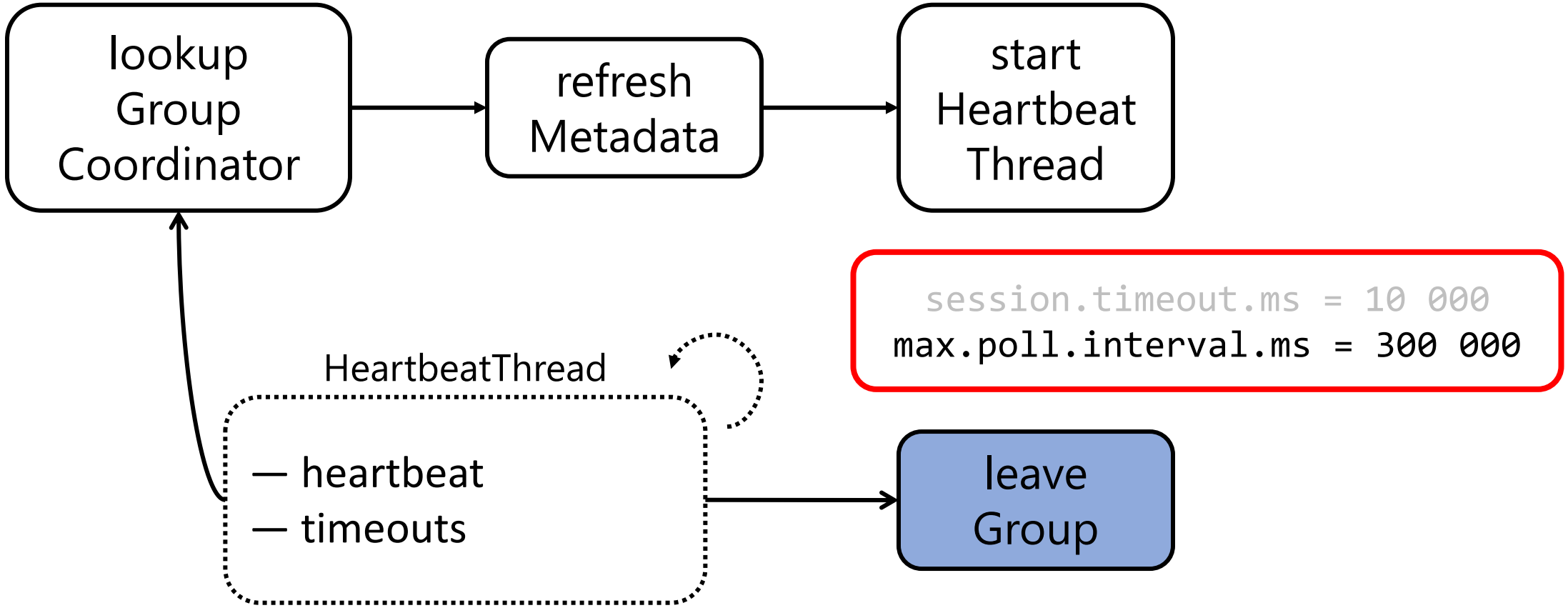
Consumer под капотом — coordinator



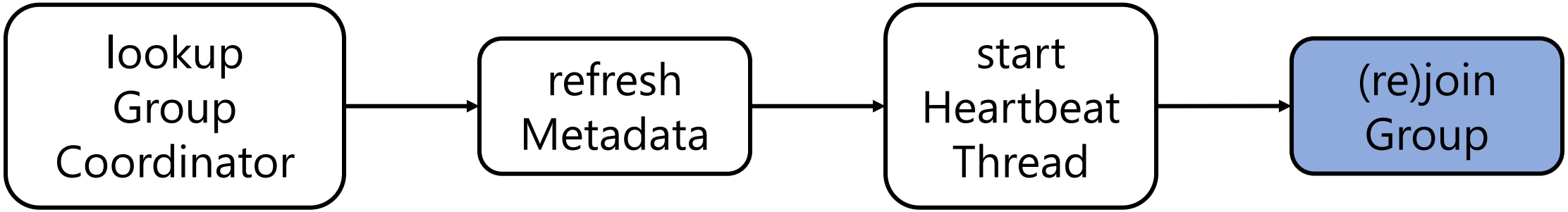
Consumer под капотом — coordinator



Consumer под капотом — coordinator

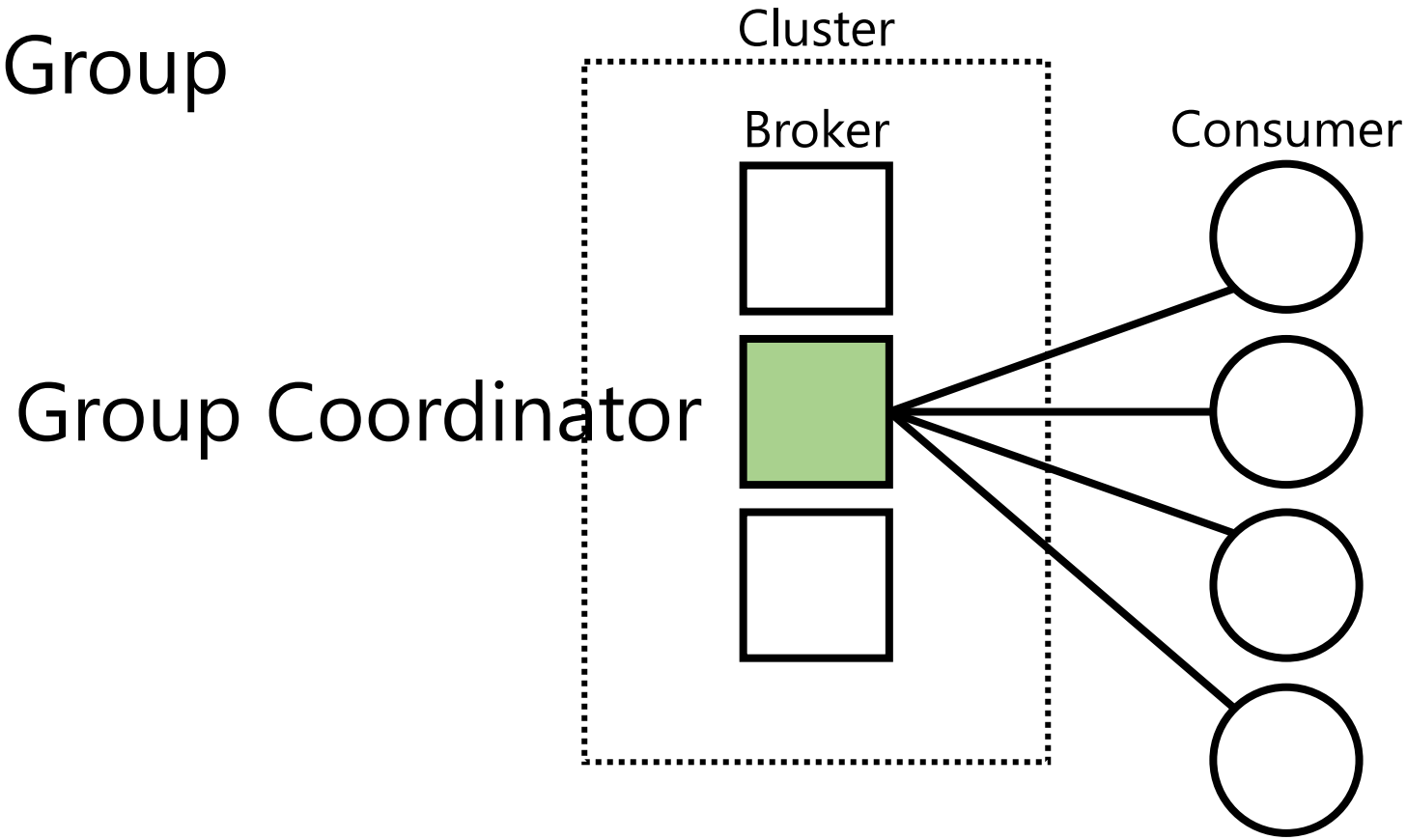


Consumer под капотом — coordinator



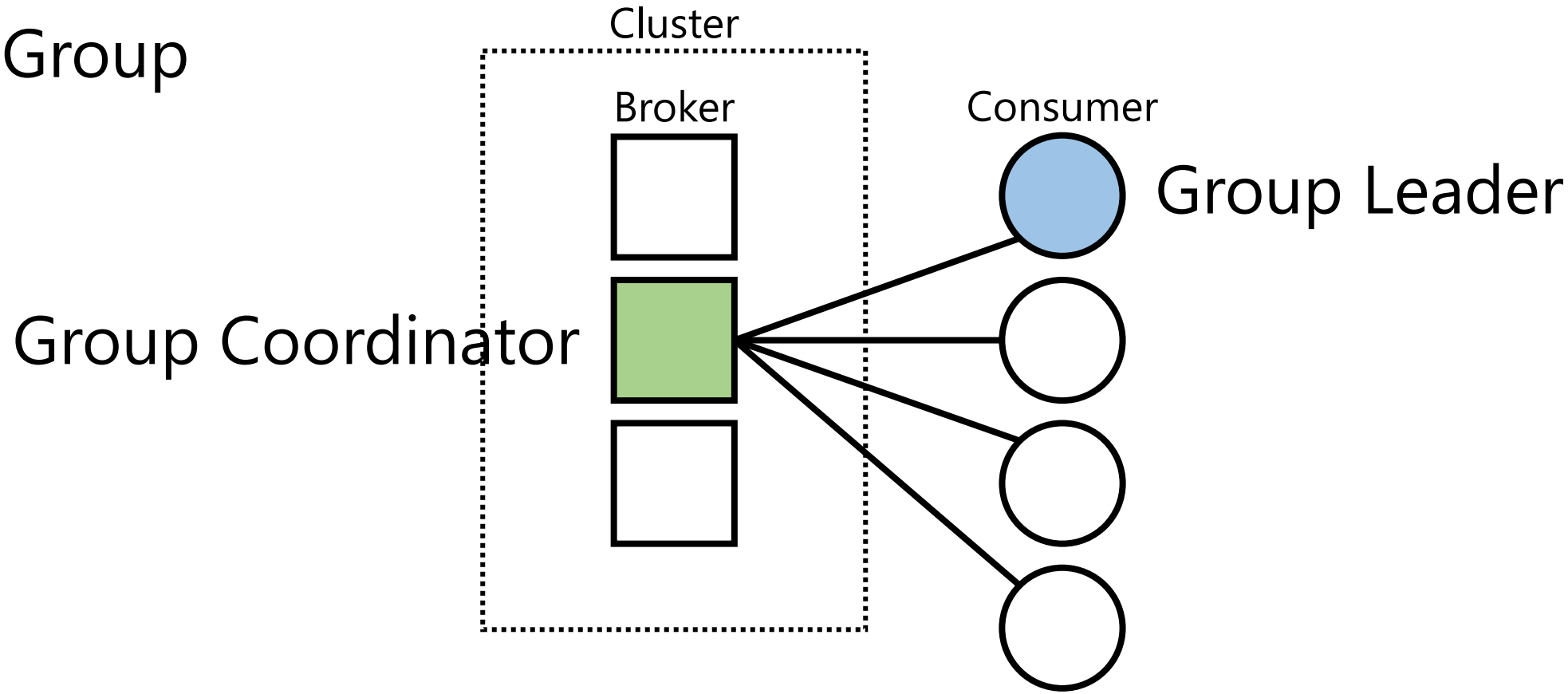
Consumer под капотом — coordinator

(re)join Group



Consumer под капотом — coordinator

(re)join Group



RebalanceProtocol: EAGER vs COOPERATIVE

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

— Партиции изымаются перед балансировкой, потом распределяются заново

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER
— Партиции изымаются перед балансировкой,
потом распределяются заново

COOPERATIVE
— Назначенные партиции не трогаются,
алгоритм балансировки может поменять распределение

KIP-429
v.2.4.0

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

— RangeAssigner

— RoundRobinAssigner

— StickyAssigner

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

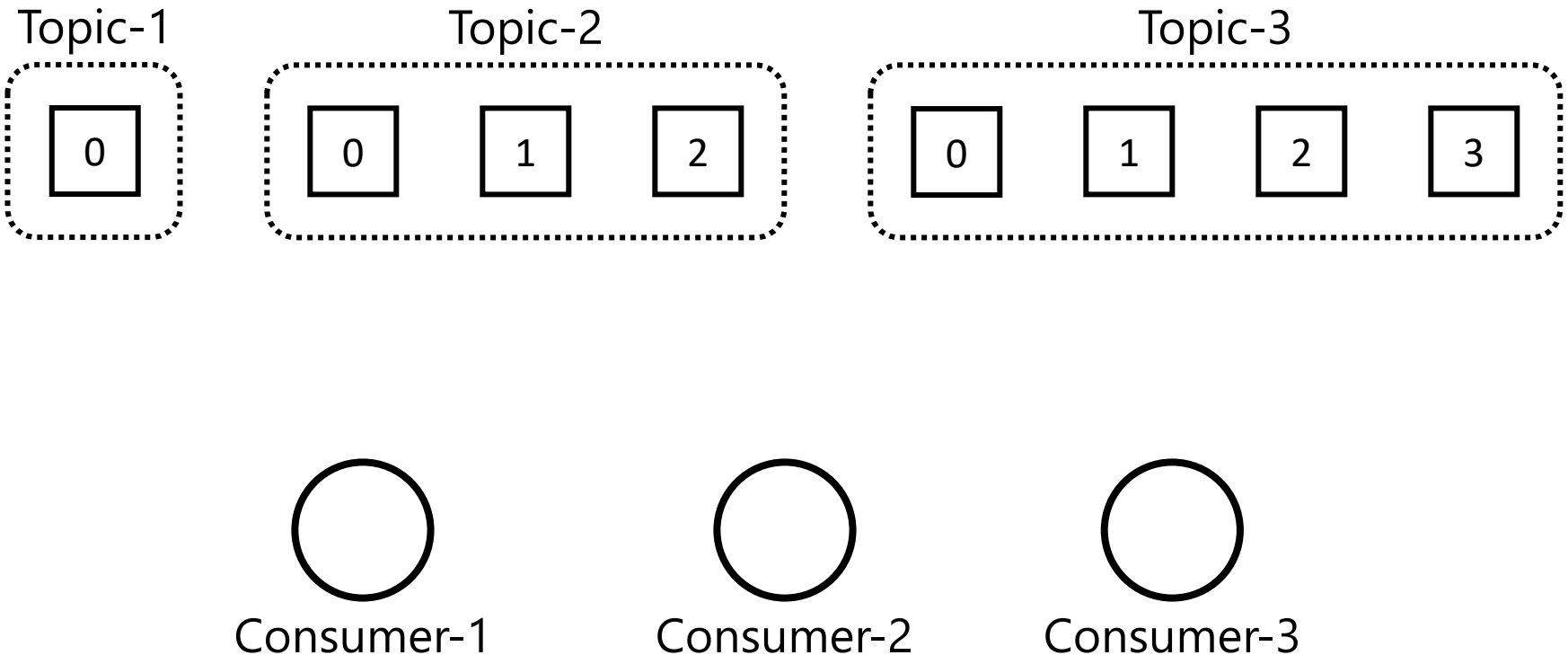
— RangeAssigner

— RoundRobinAssigner

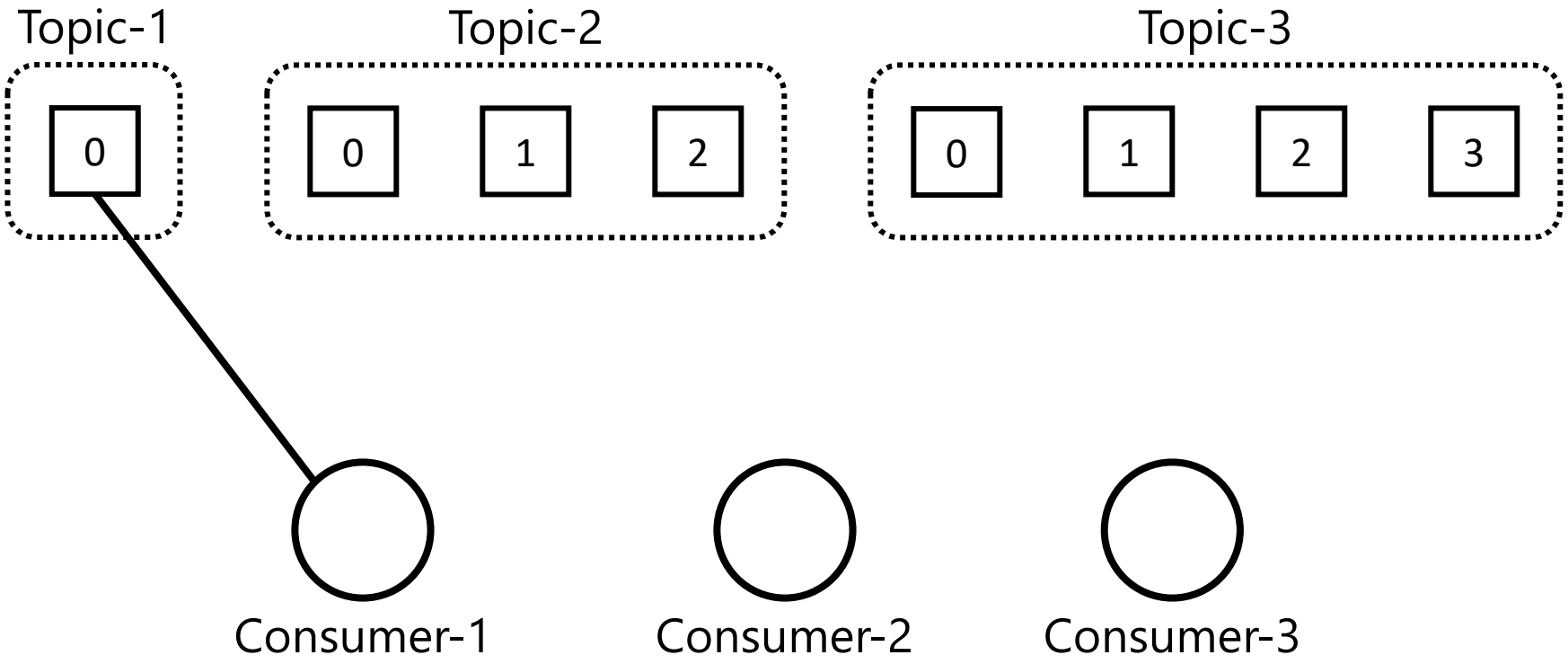
— StickyAssigner

```
partition.assignment.strategy =
org.apache.kafka.clients.consumer.RangeAssignor
```

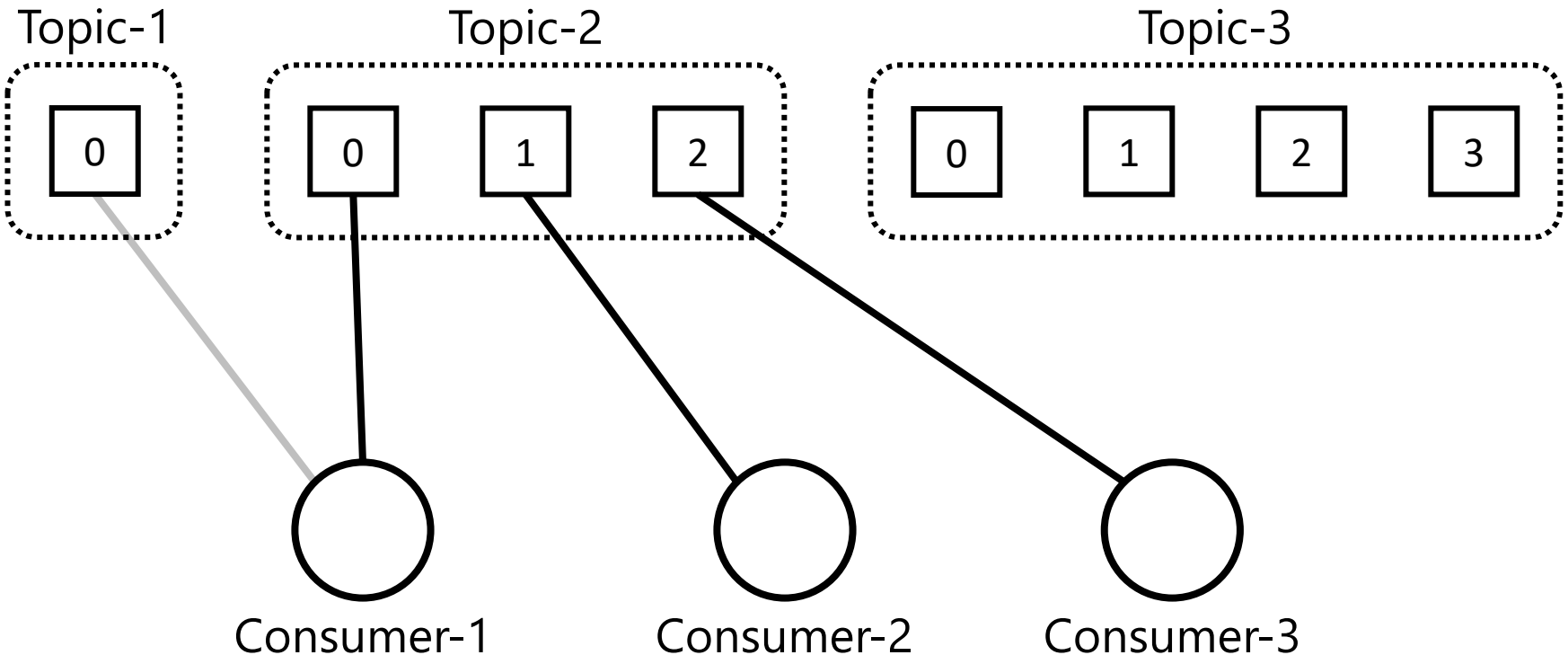
RangeAssigner



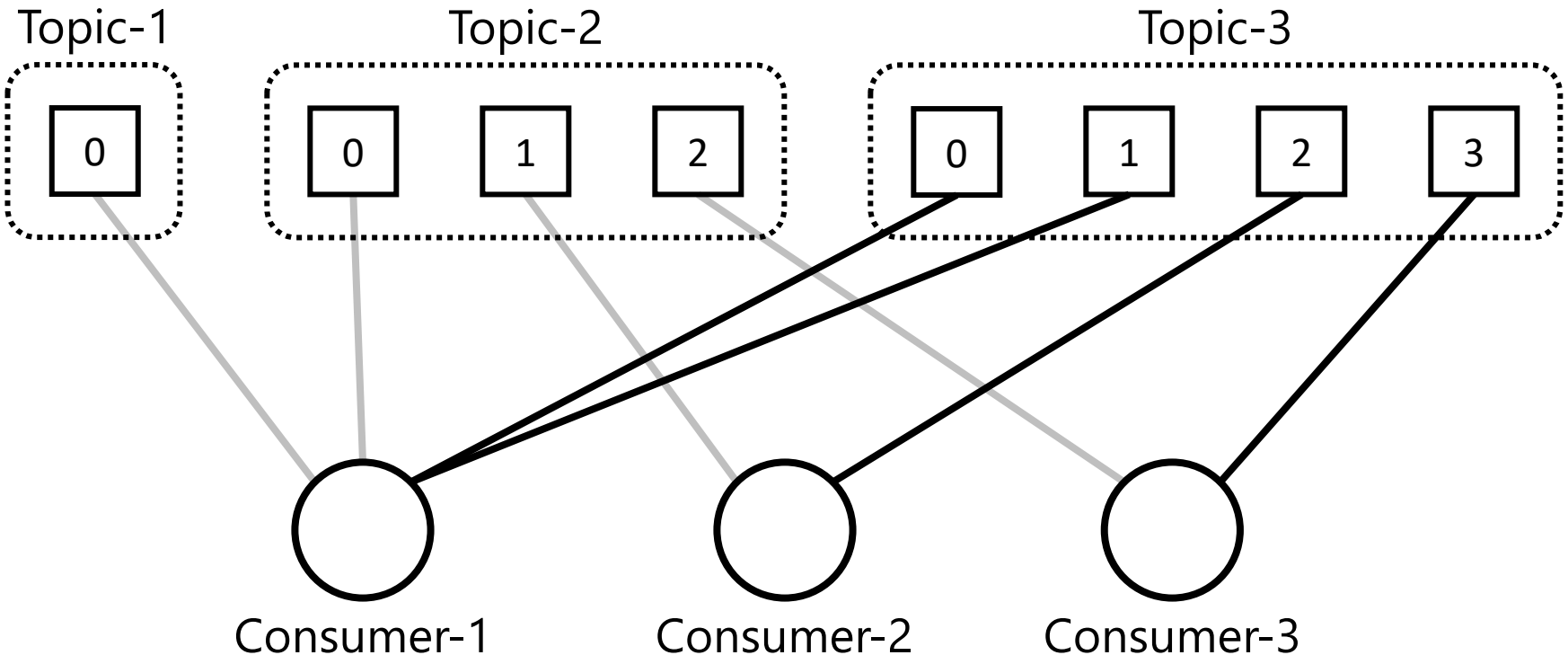
RangeAssigner



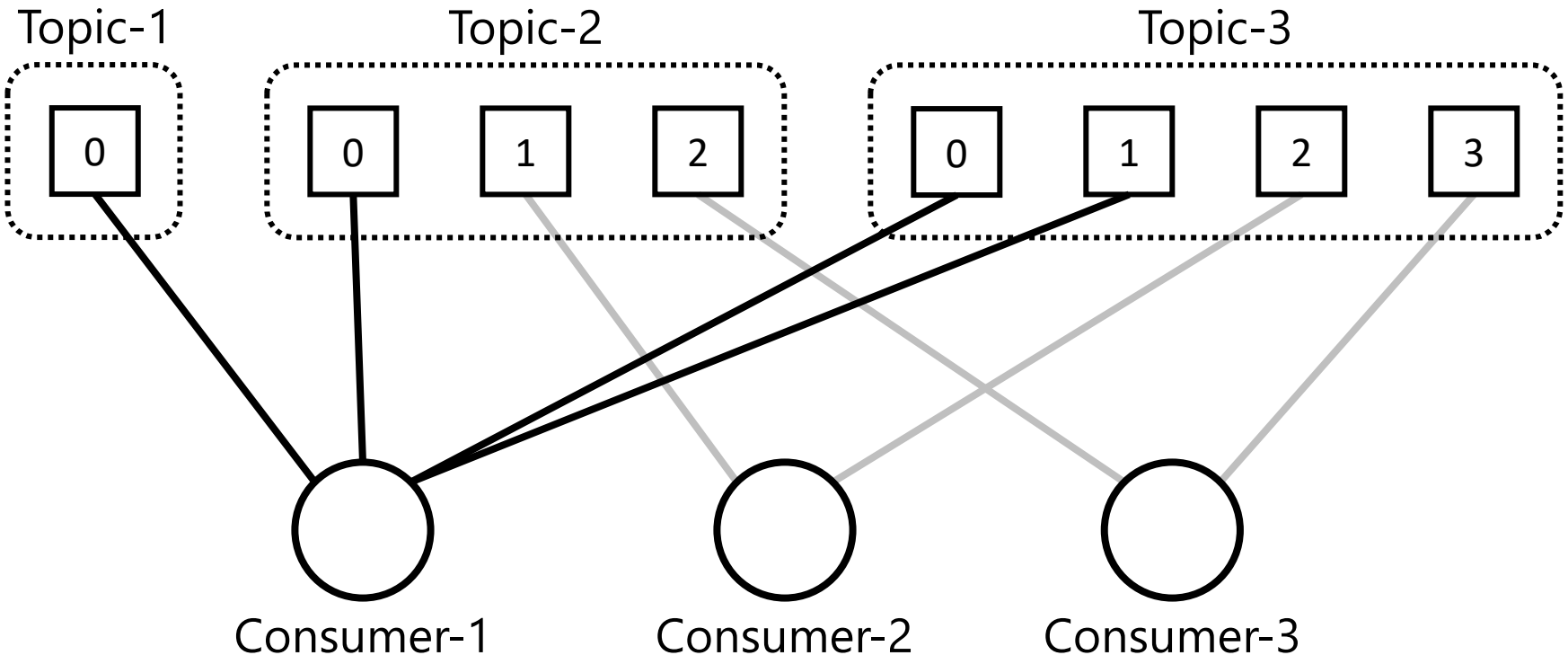
RangeAssigner



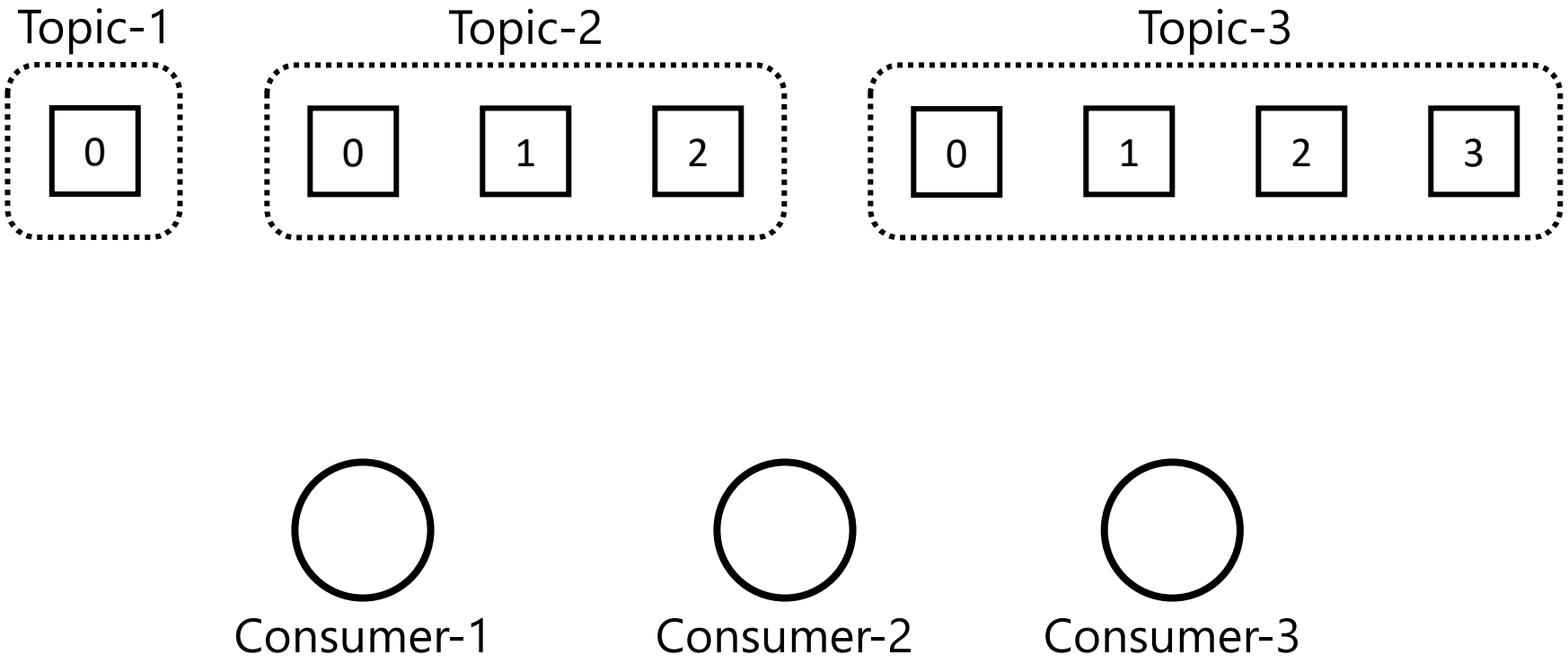
RangeAssigner



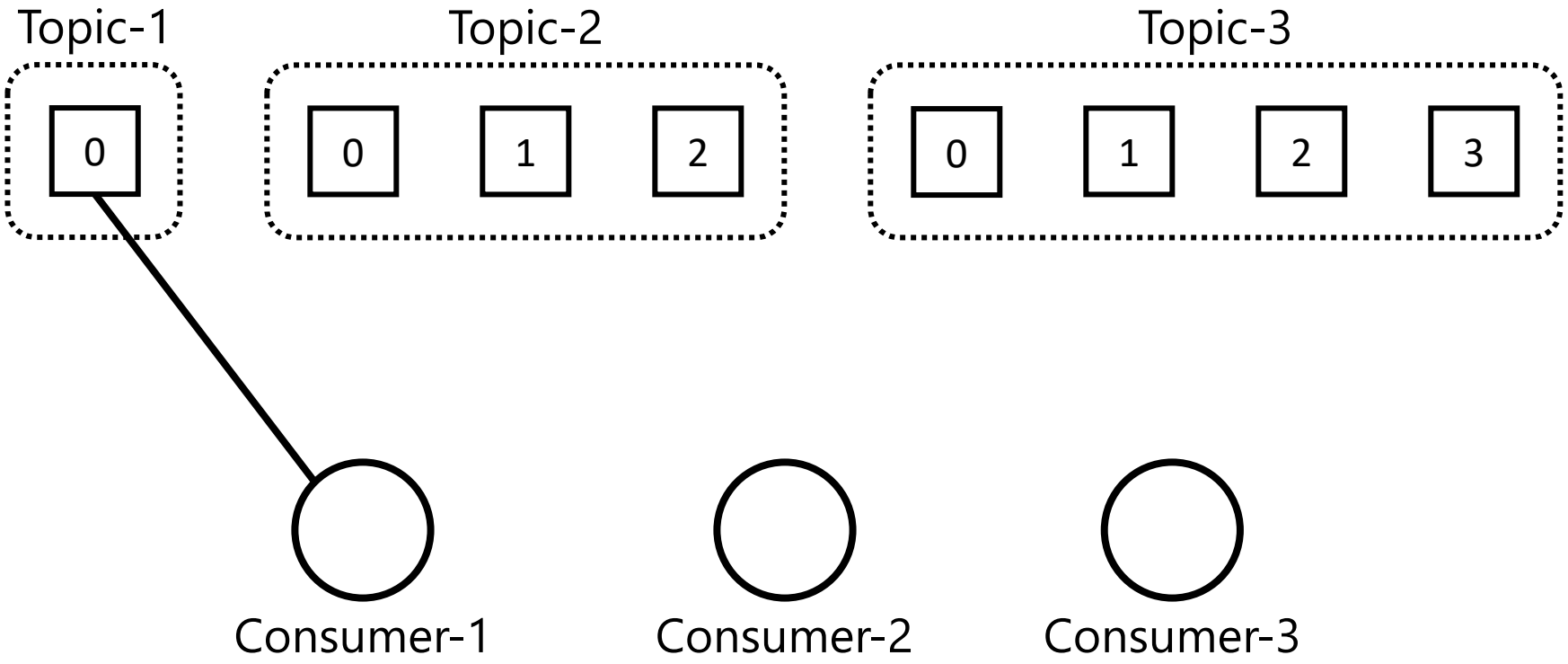
RangeAssigner



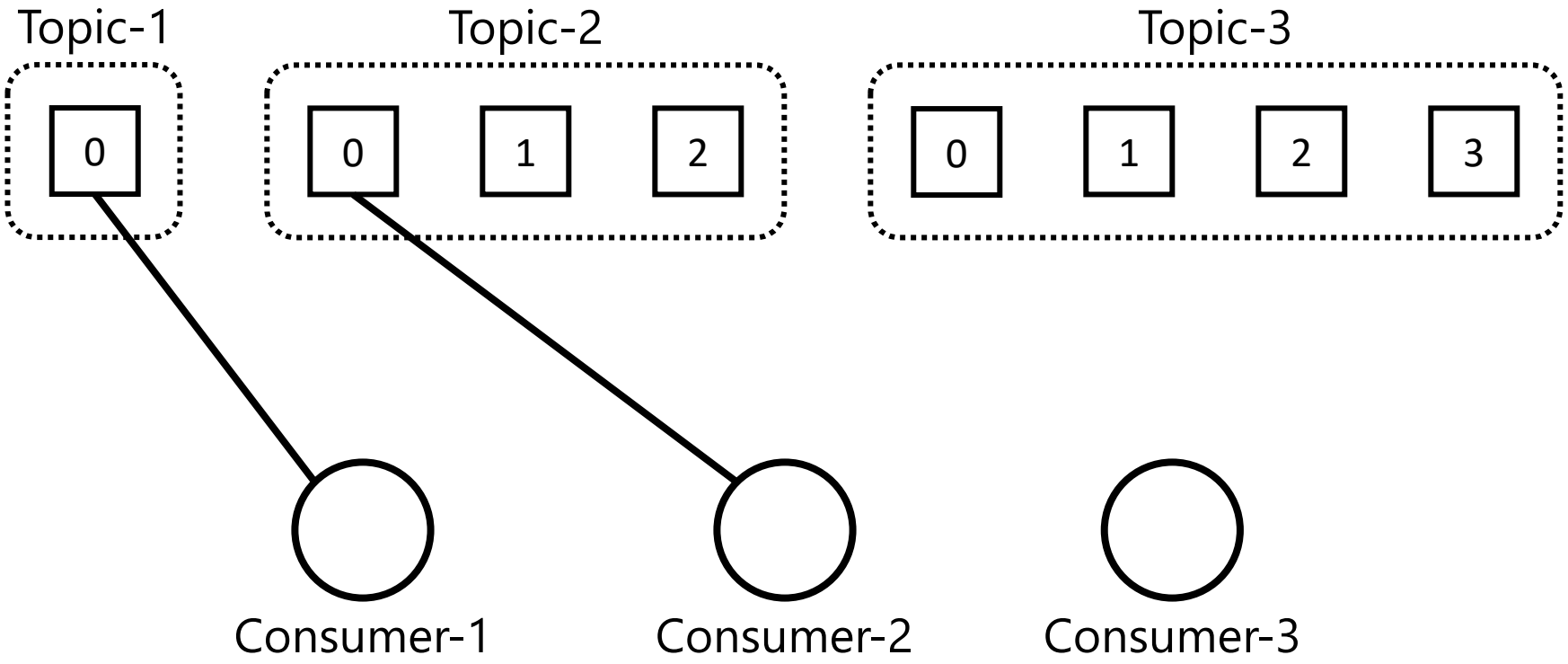
RoundRobinAssigner



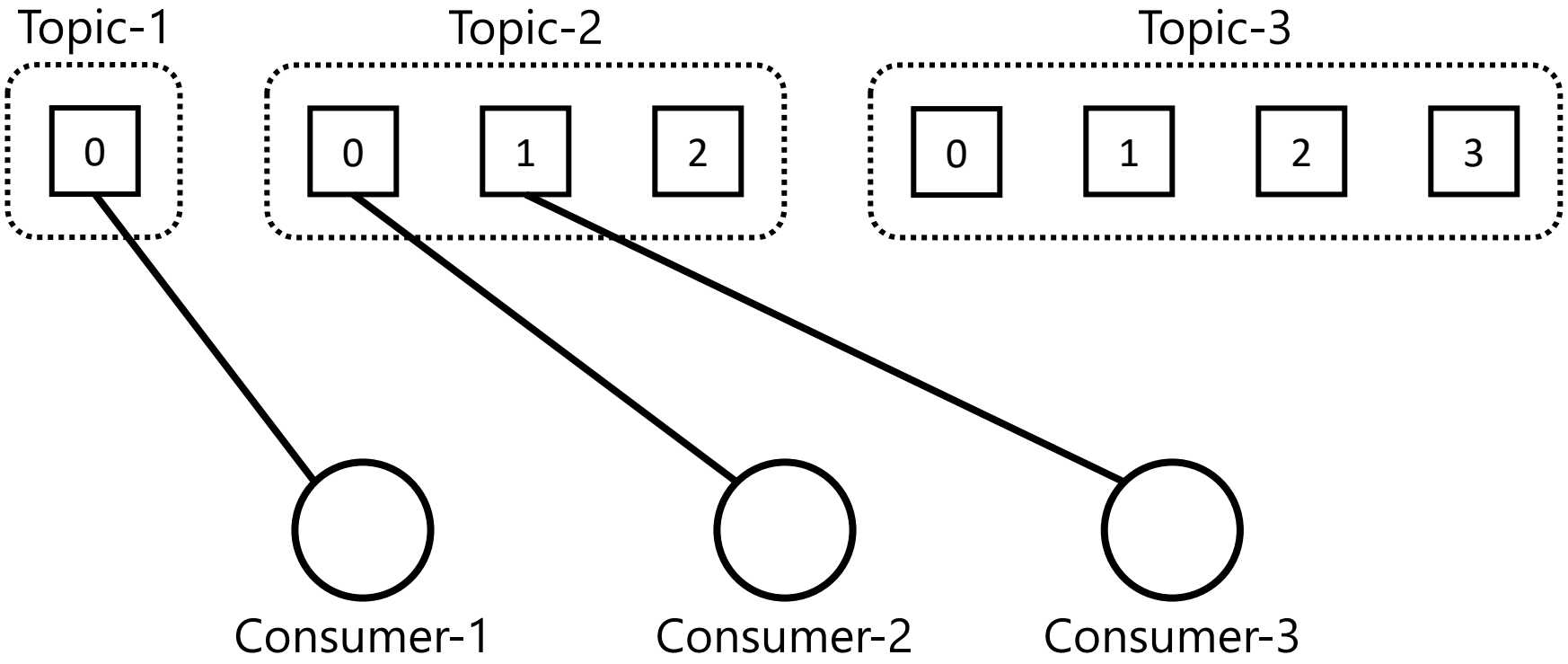
RoundRobinAssigner



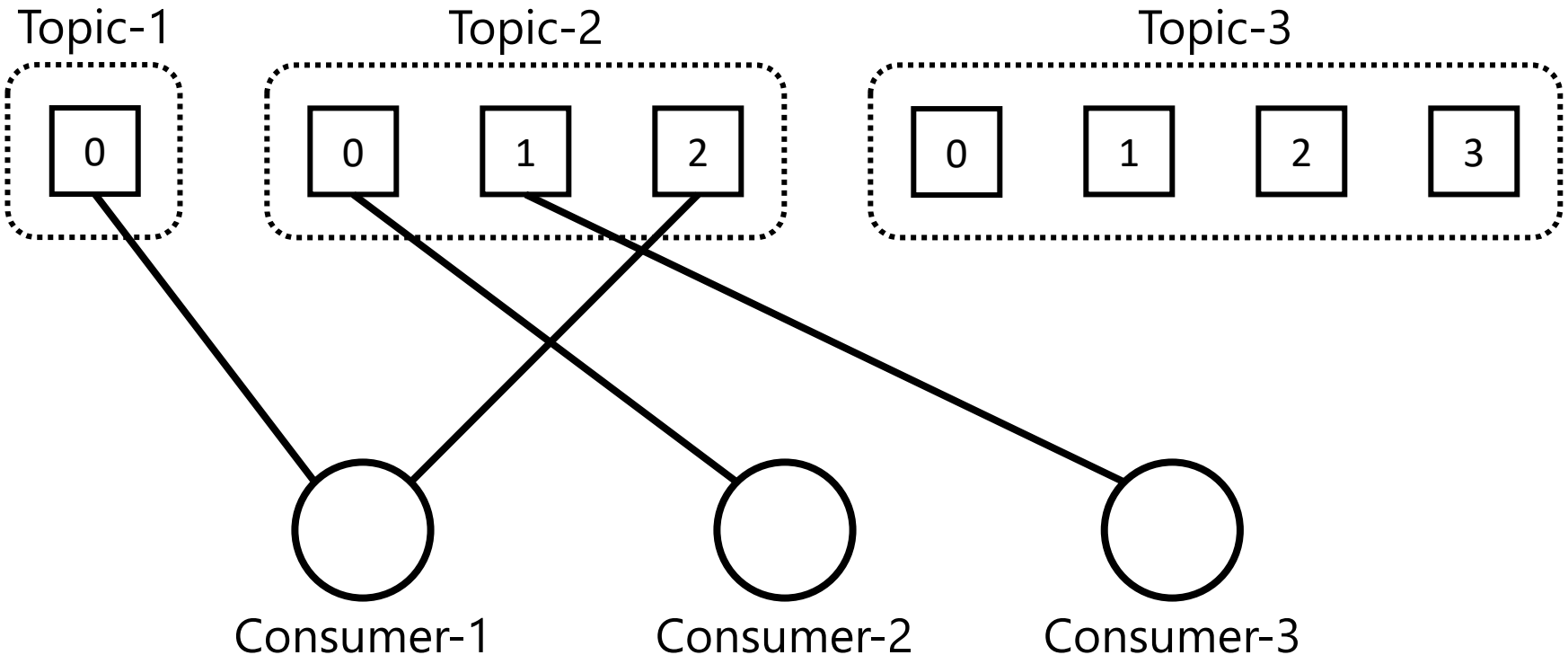
RoundRobinAssigner



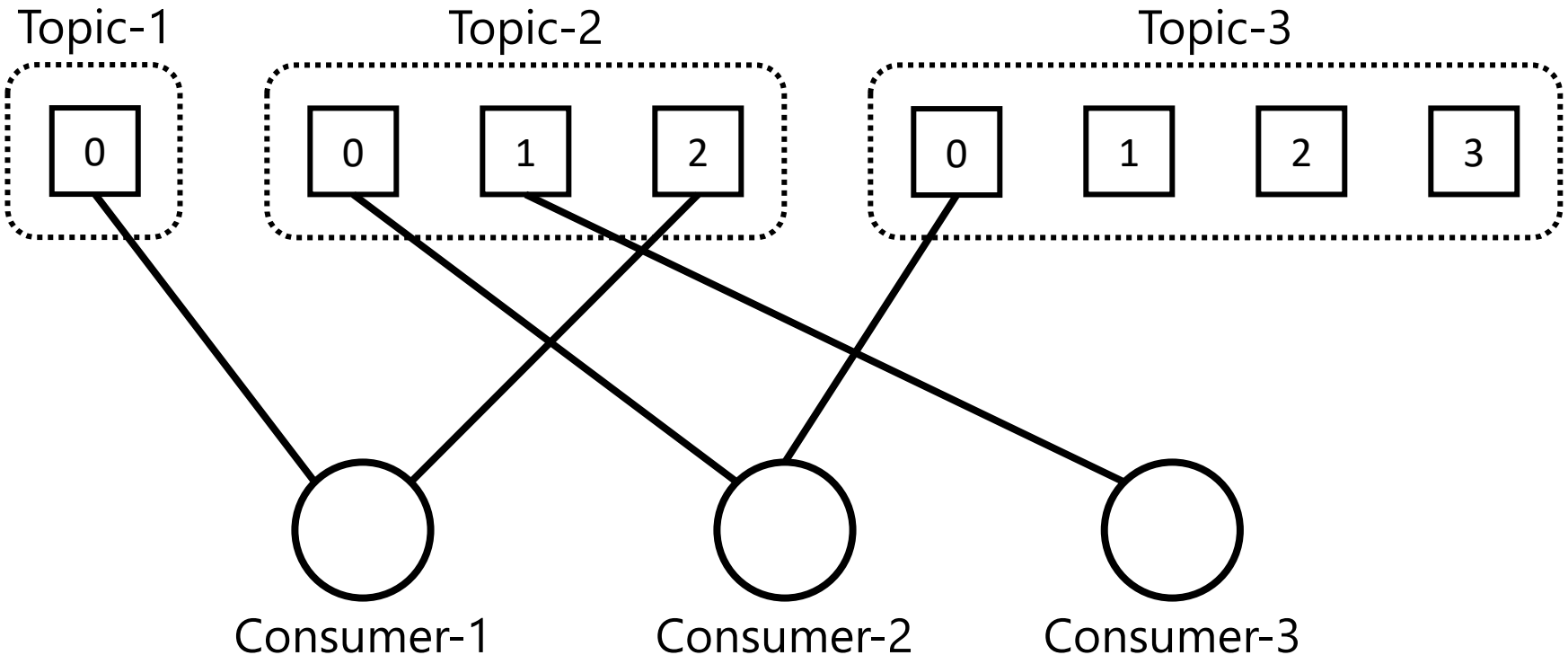
RoundRobinAssigner



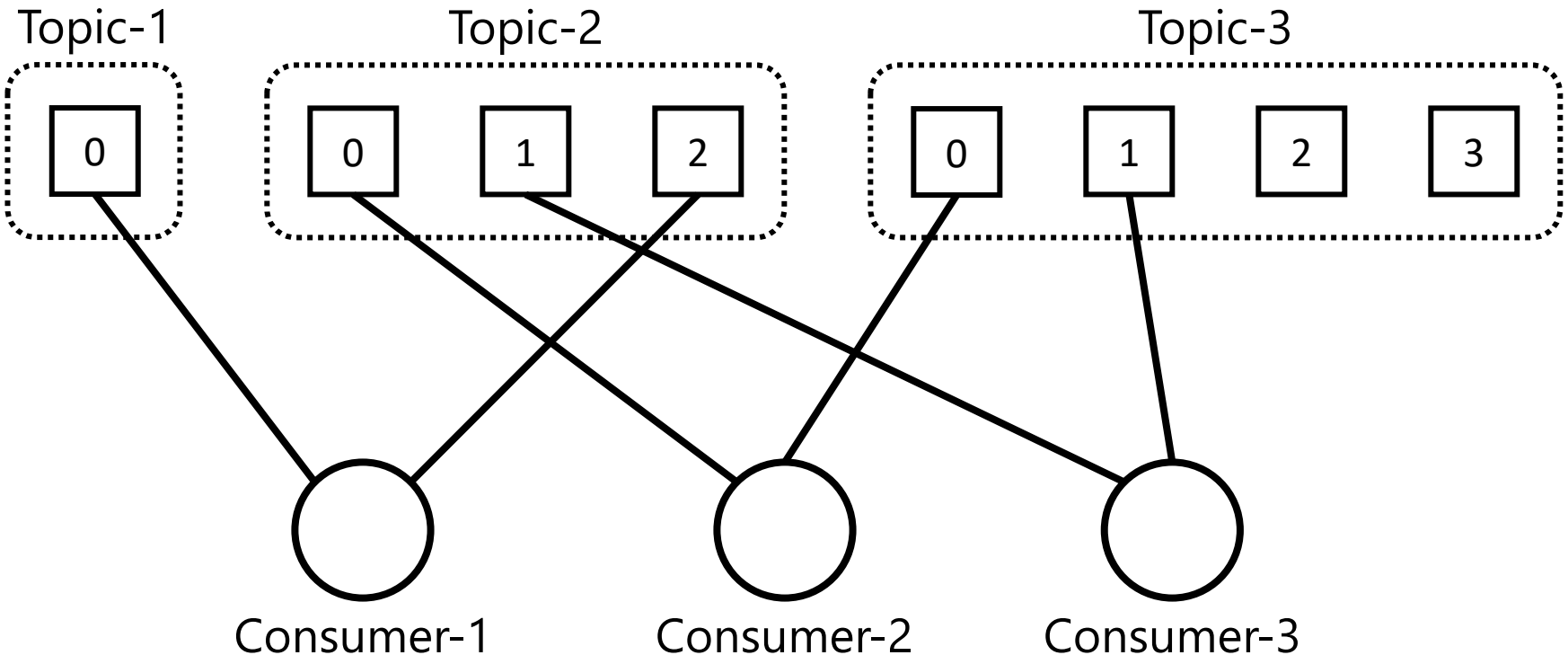
RoundRobinAssigner



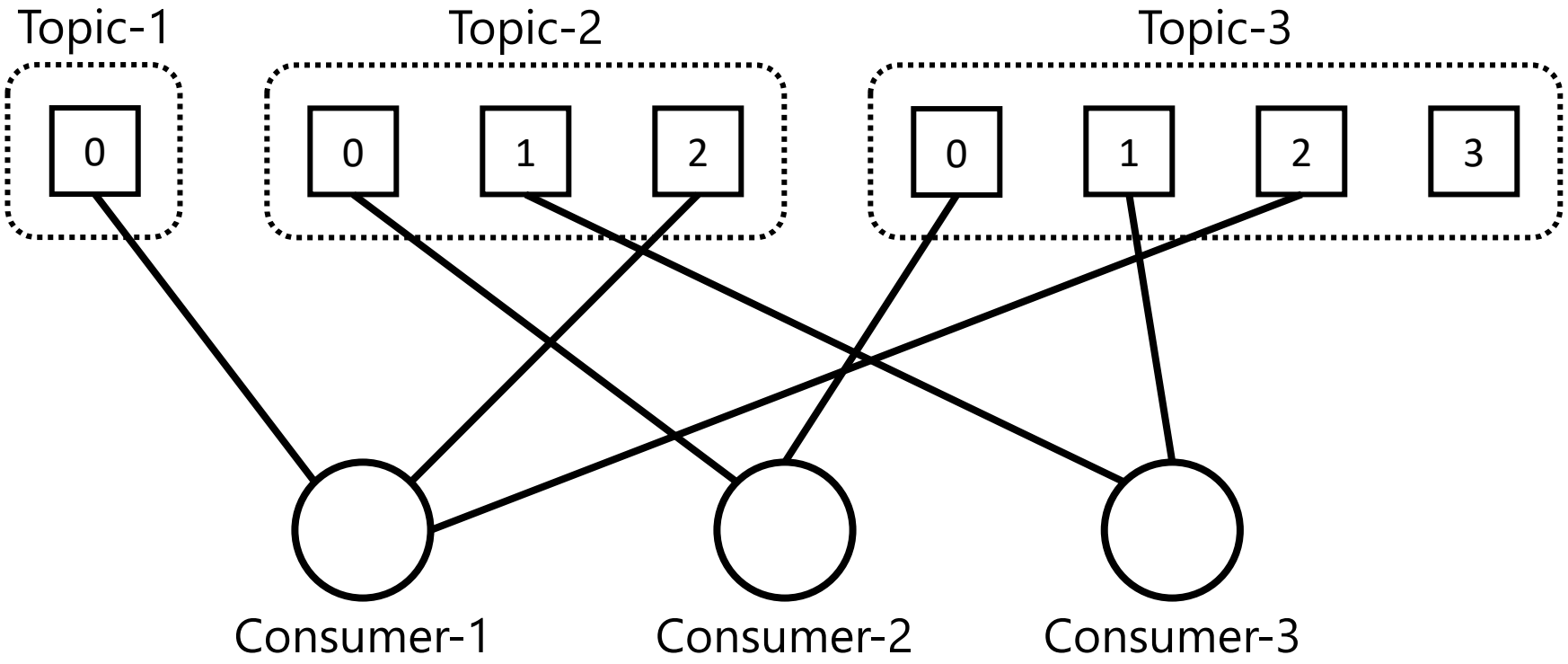
RoundRobinAssigner



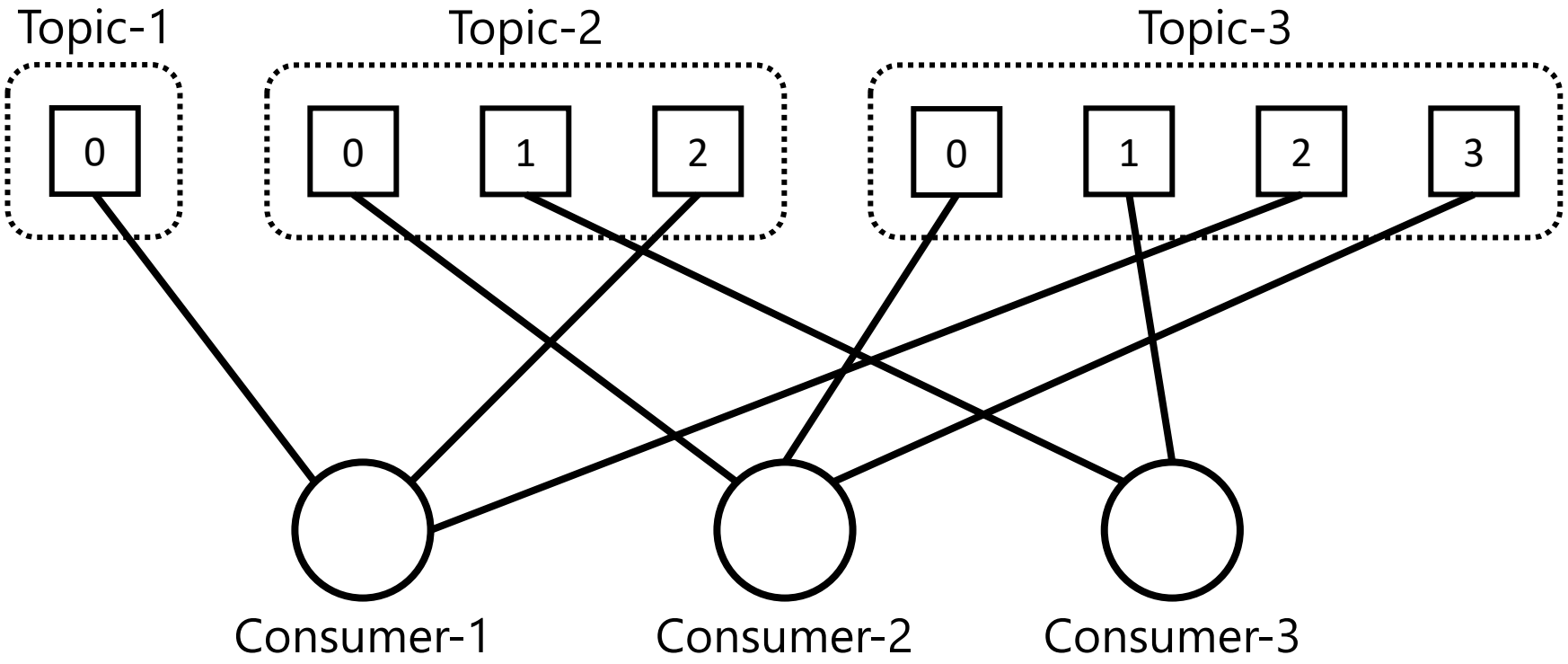
RoundRobinAssigner



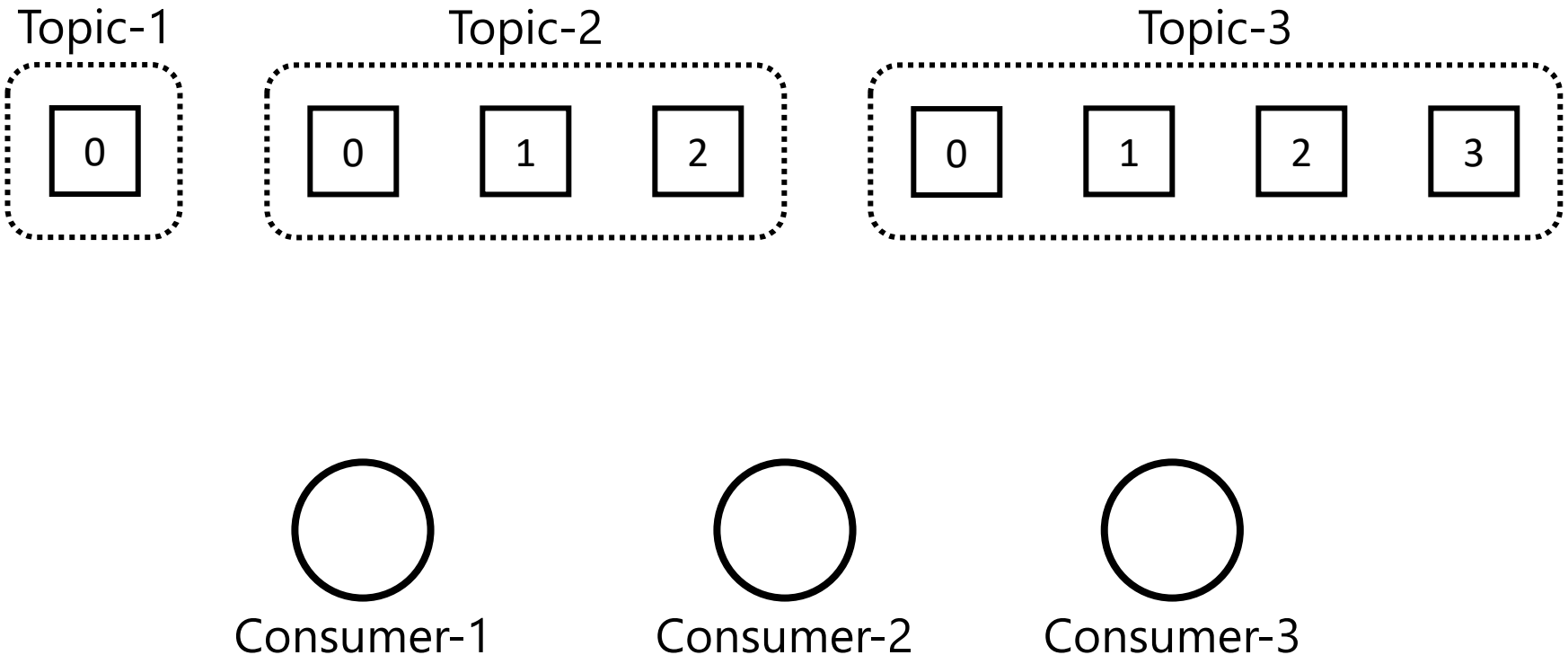
RoundRobinAssigner



RoundRobinAssigner

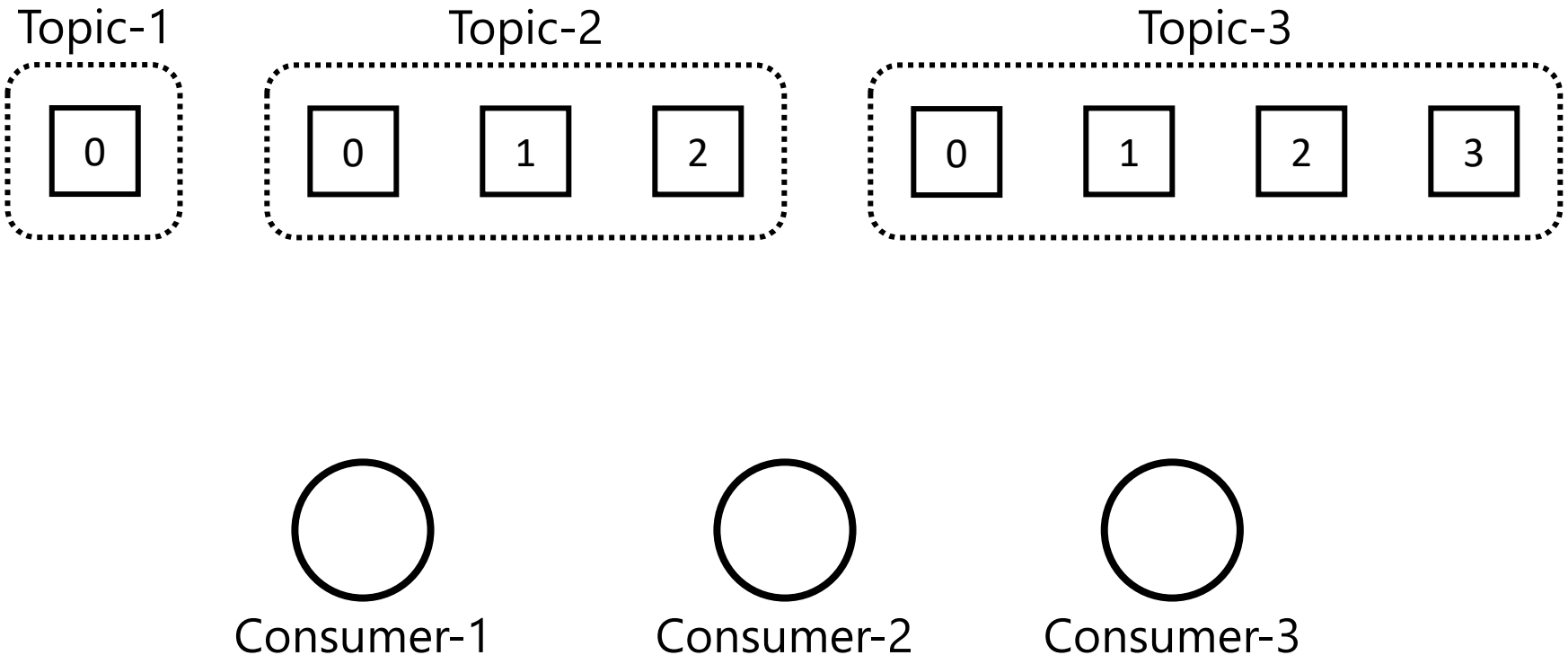


StickyAssigner



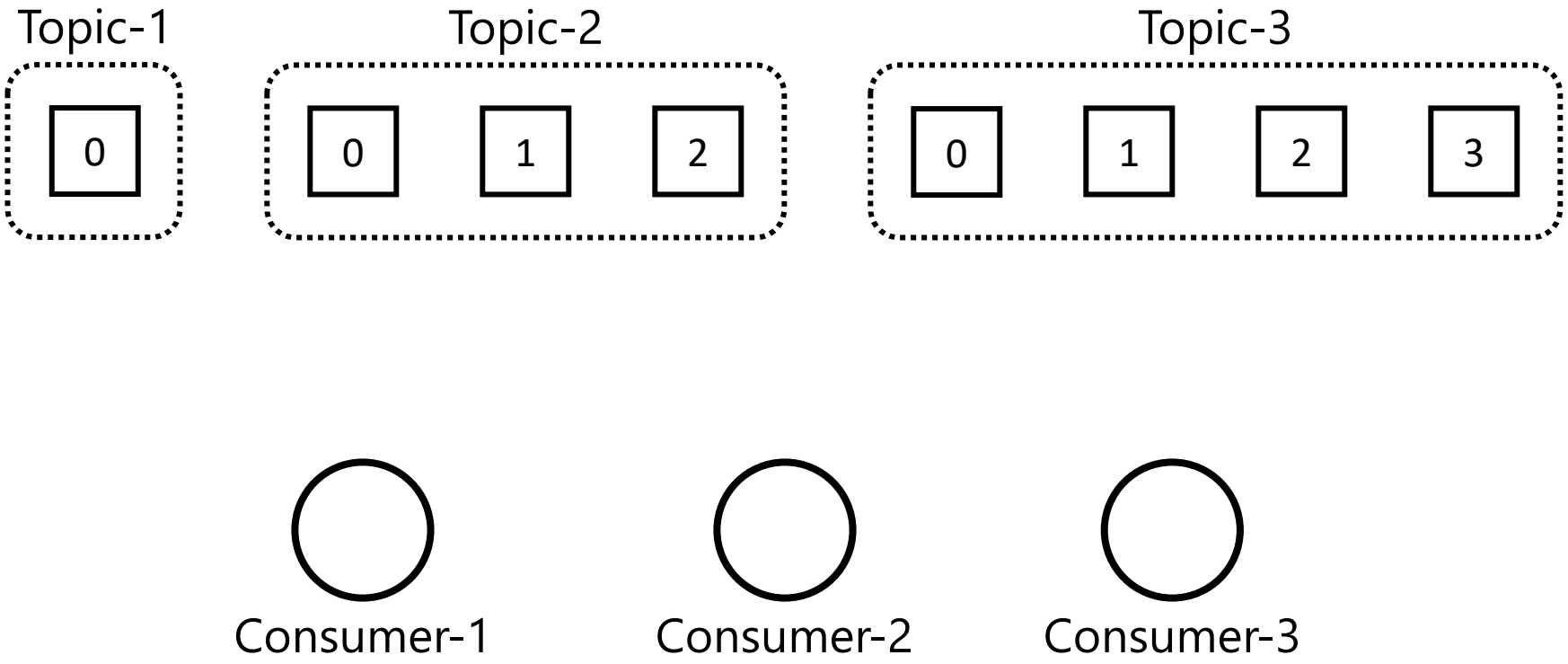
StickyAssigner

От 2 до 3 партиций на 1 consumer



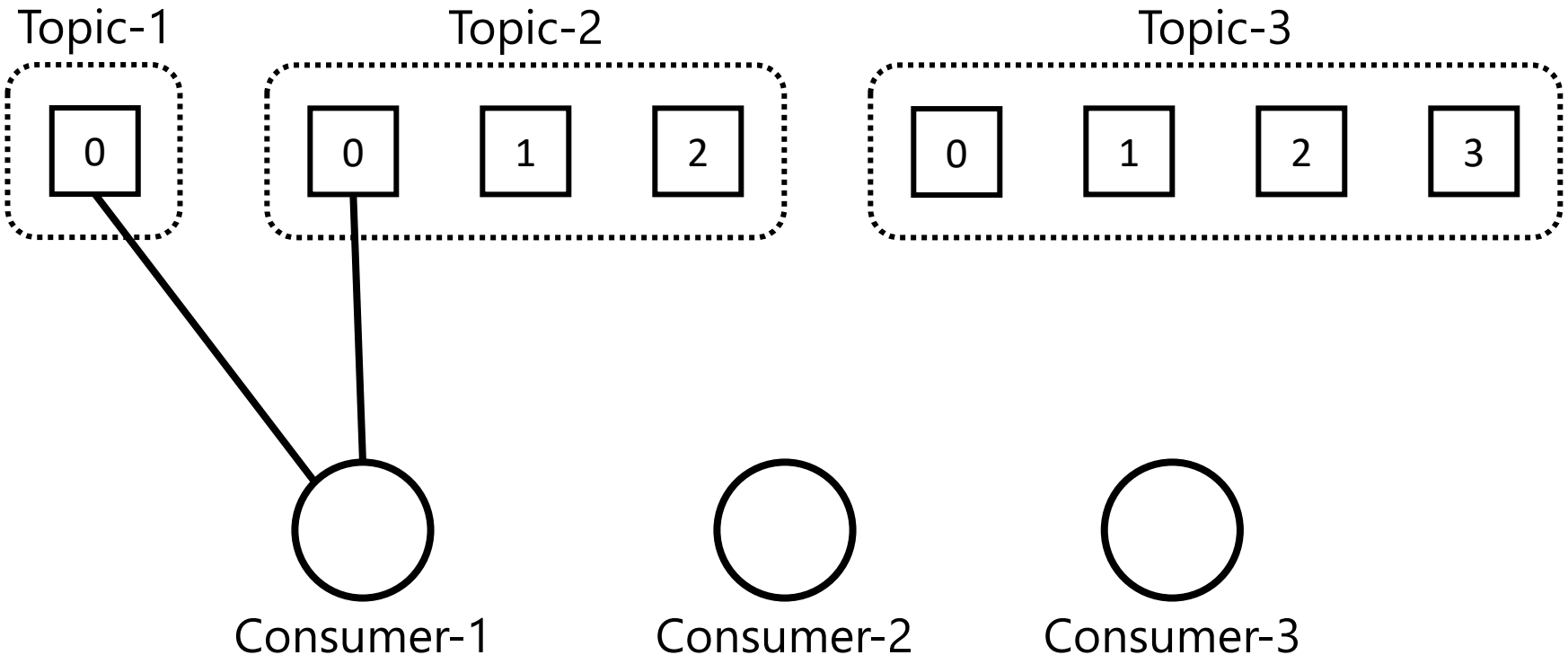
StickyAssigner

От 2 до 3 партиций на 1 consumer



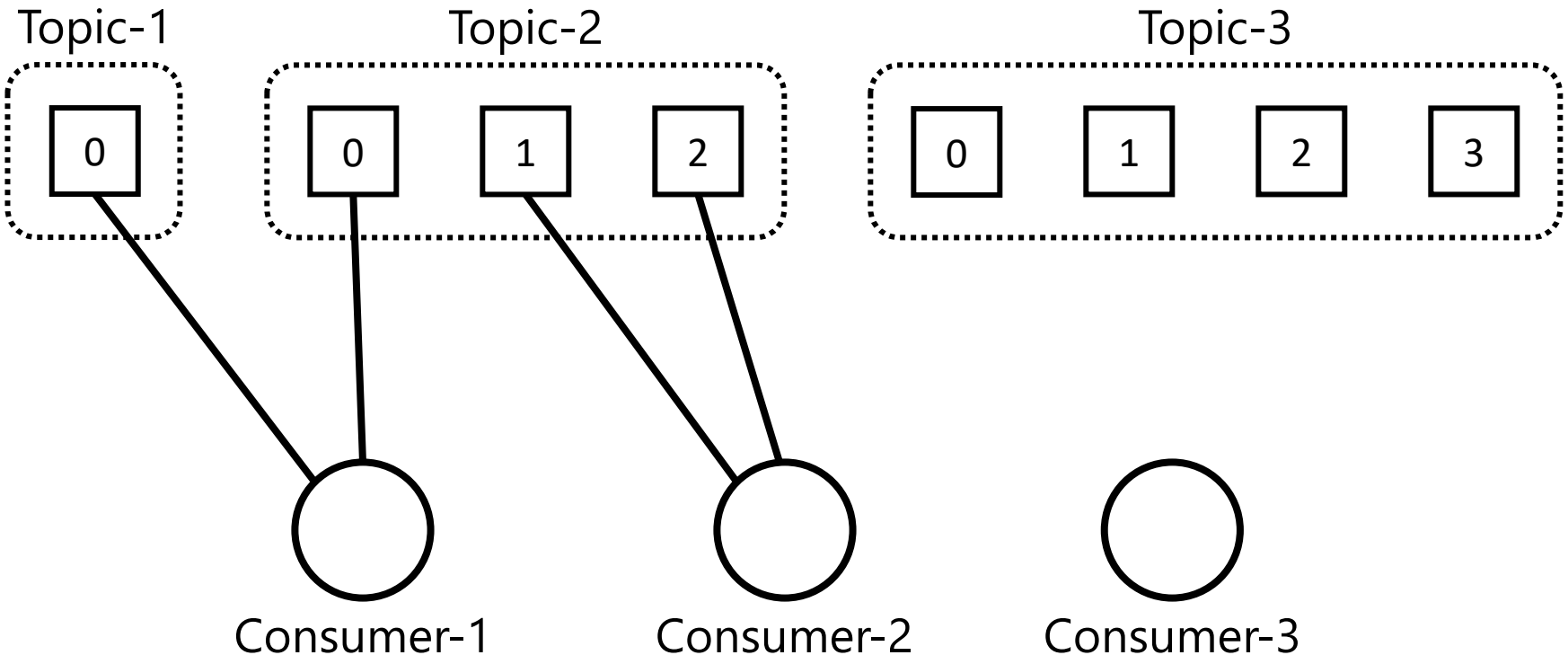
StickyAssigner

От 2 до 3 партиций на 1 consumer



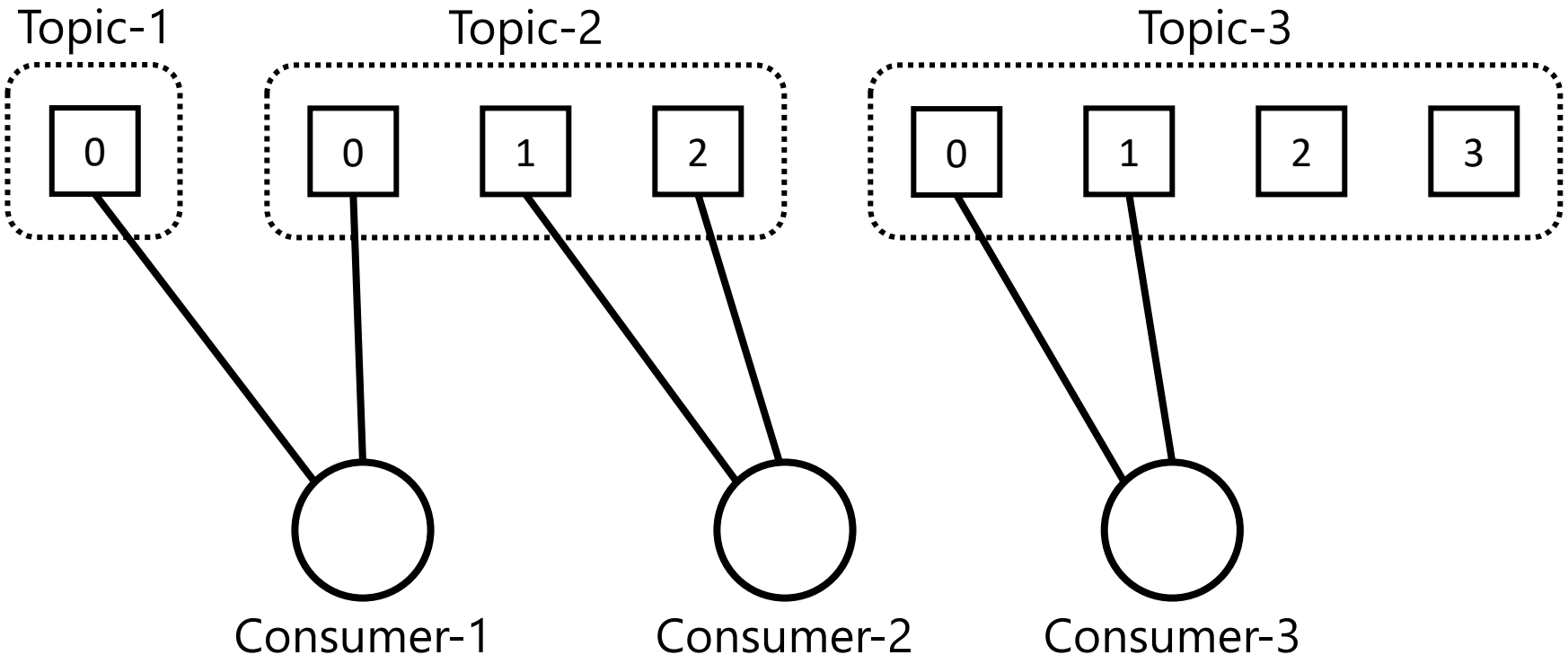
StickyAssigner

От 2 до 3 партиций на 1 consumer



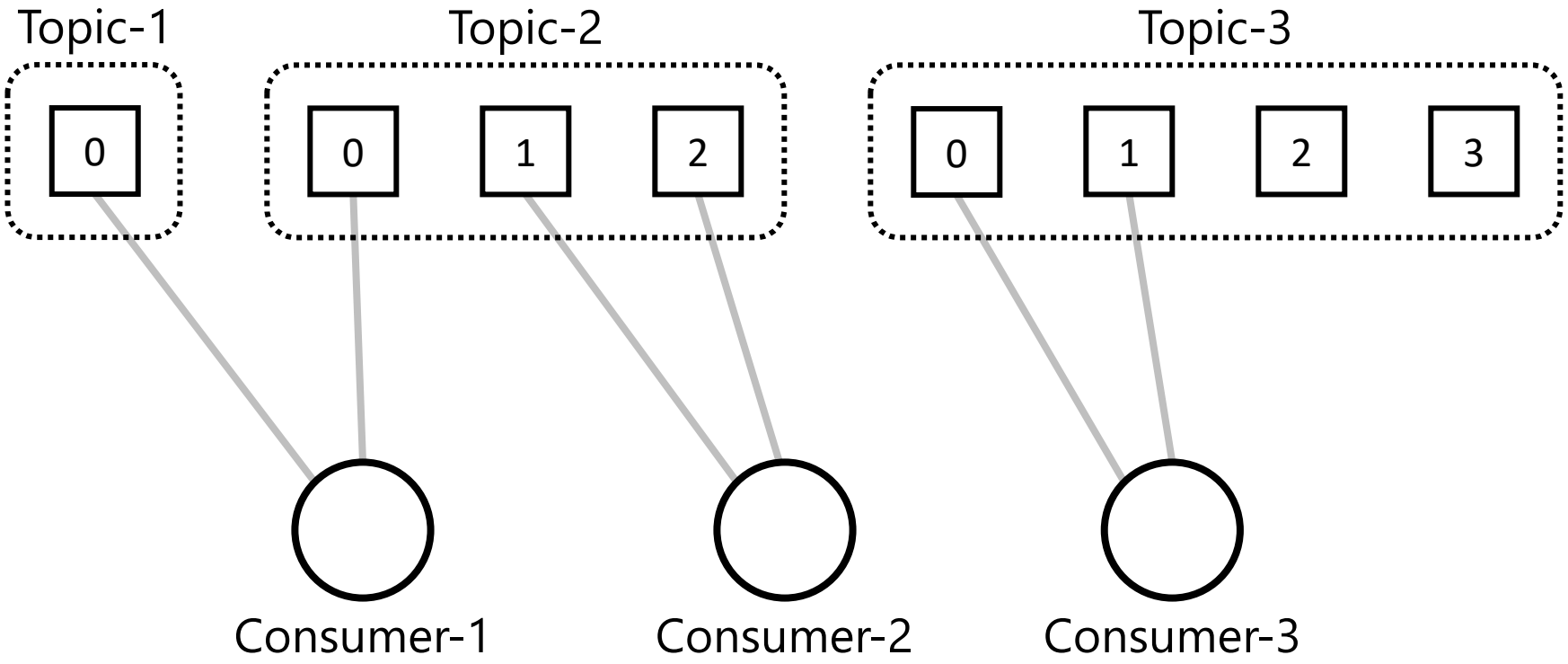
StickyAssigner

От 2 до 3 партиций на 1 consumer



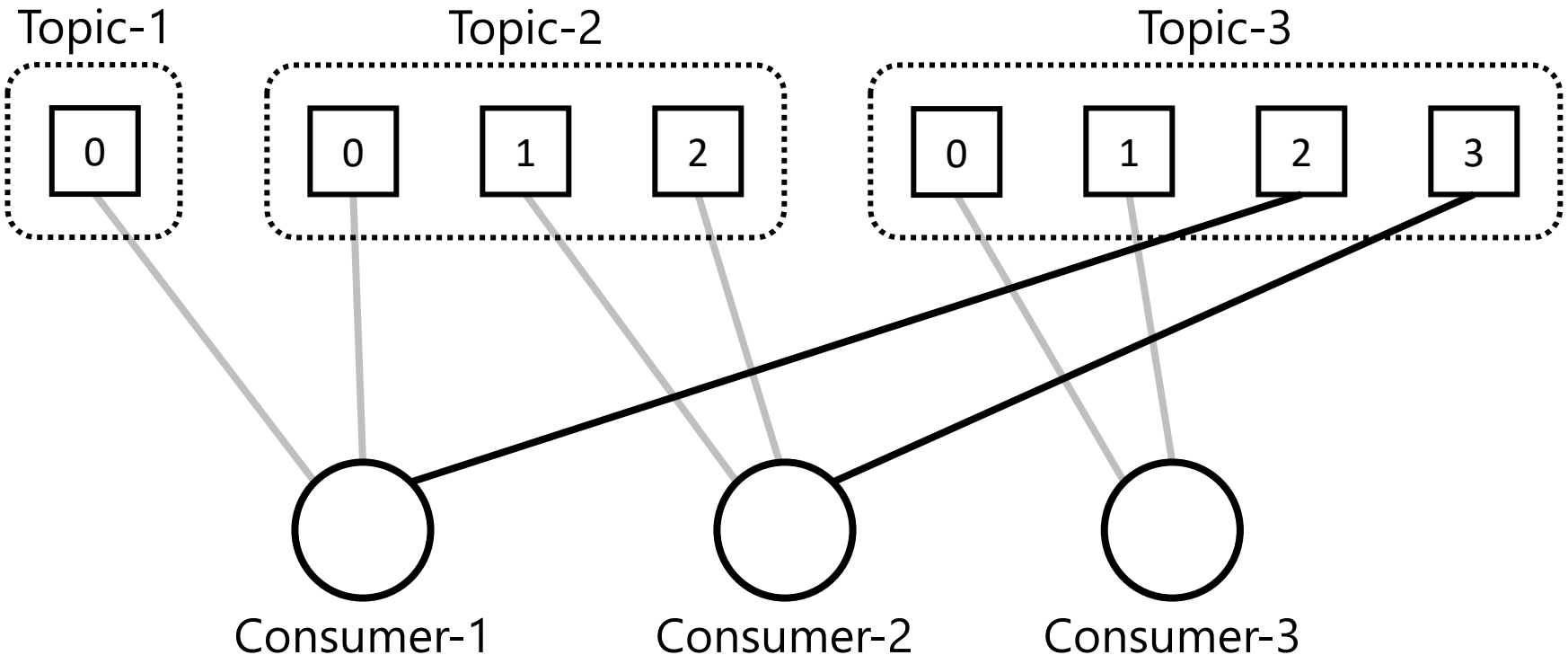
StickyAssigner

От 2 до 3 партиций на 1 consumer



StickyAssigner

От 2 до 3 партиций на 1 consumer



RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

— RangeAssigner

— RoundRobinAssigner

— StickyAssigner

```
partition.assignment.strategy =
org.apache.kafka.clients.consumer.RangeAssignor
```

COOPERATIVE

— CooperativeStickyAssigner

KIP-429
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-429%3A+Kafka+Consumer+Incremental+Rebalance+Protocol

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

— **RangeAssigner**

— RoundRobinAssigner

— StickyAssigner

```
partition.assignment.strategy =
org.apache.kafka.clients.consumer.RangeAssignor,
o.a.k.clients.consumer.CooperativeStickyAssignor
```

COOPERATIVE

— **CooperativeStickyAssigner**

v.3.0.0

RebalanceProtocol: EAGER vs COOPERATIVE

EAGER

— RangeAssigner

— RoundRobinAssigner

— StickyAssigner

```
partition.assignment.strategy =
o.a.k.clients.consumer.CooperativeStickyAssignor,
org.apache.kafka.clients.consumer.RangeAssignor
```

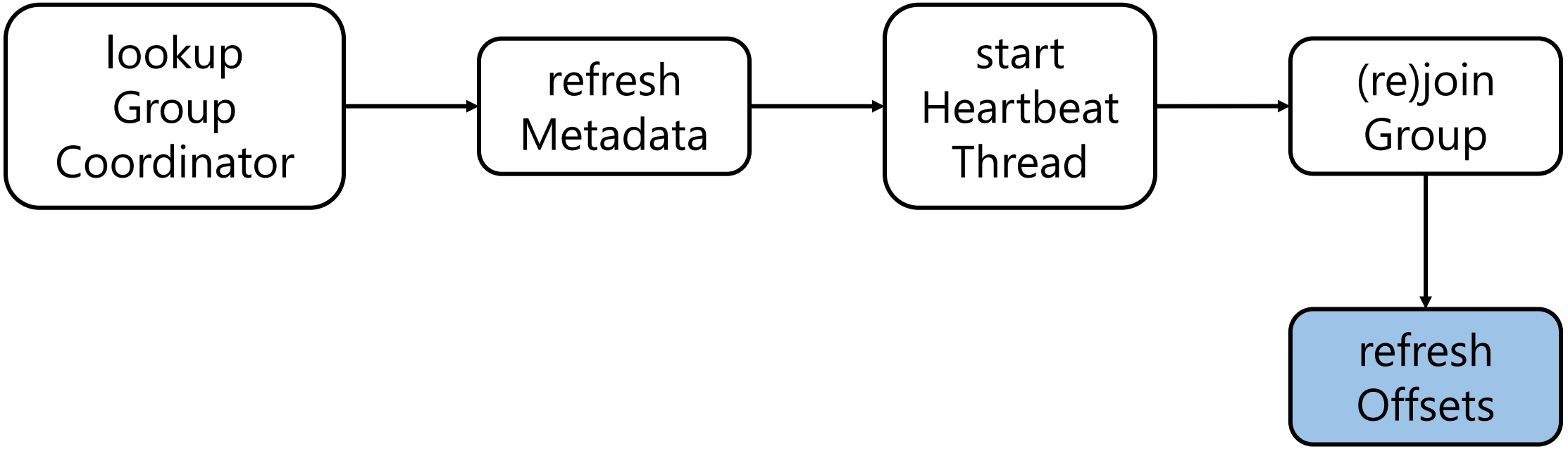
COOPERATIVE

— **CooperativeStickyAssigner**

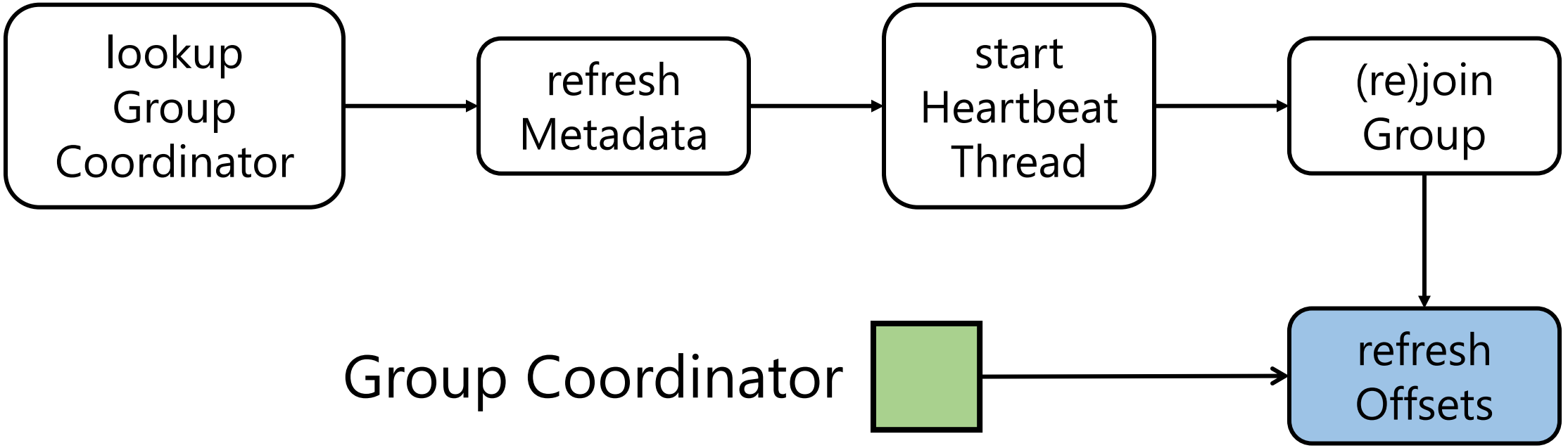
KIP-726
WIP

https://cwiki.apache.org/confluence/pages/viewpage.action?pageId=177048248

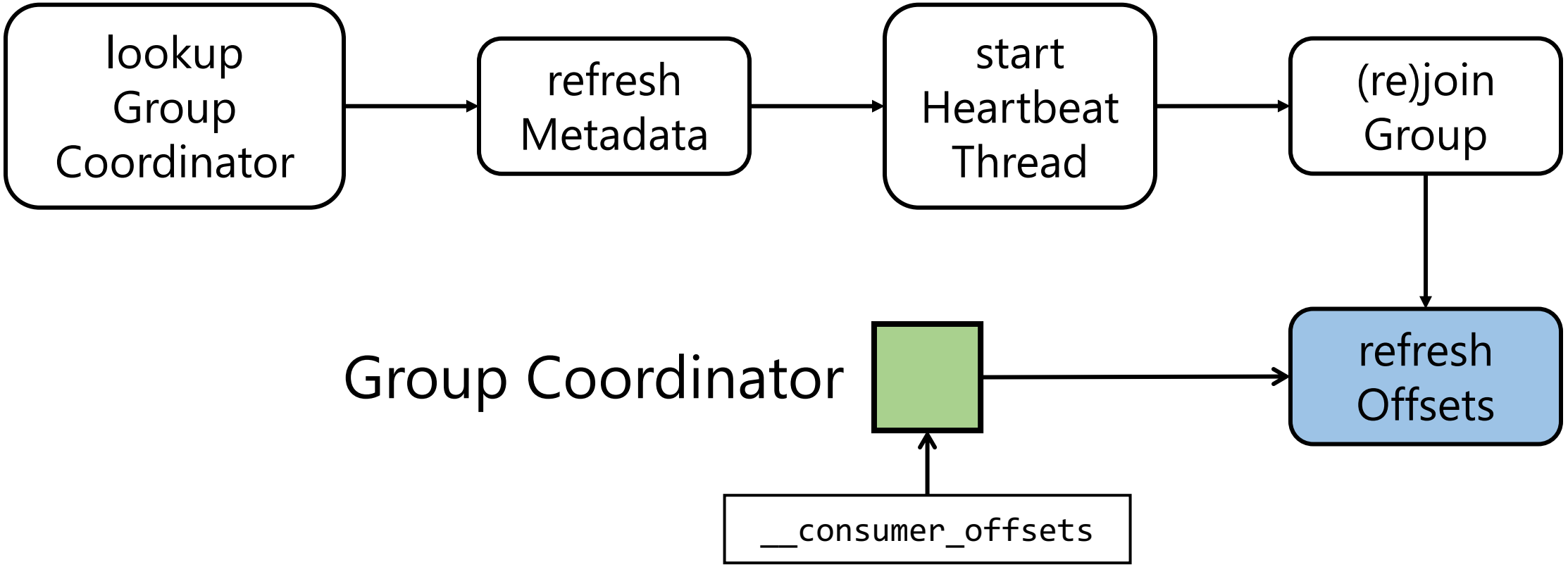
Consumer под капотом — coordinator



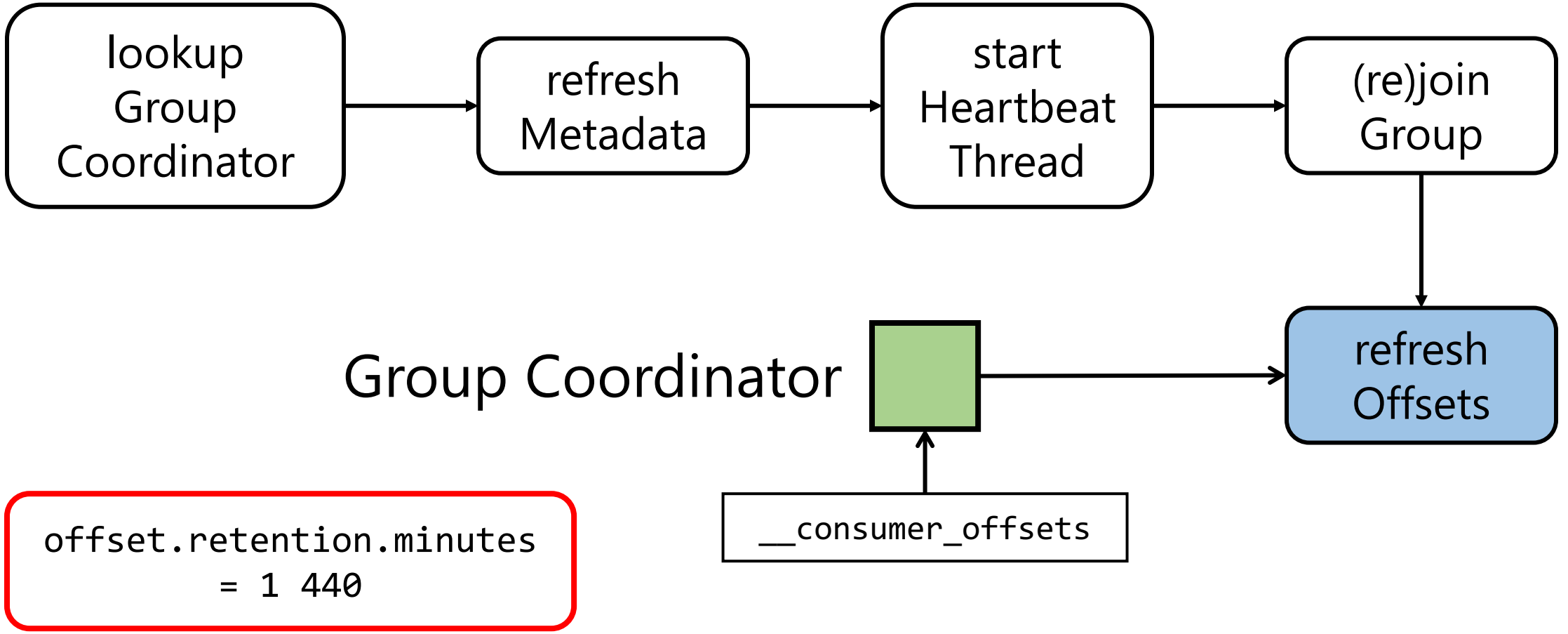
Consumer под капотом — coordinator



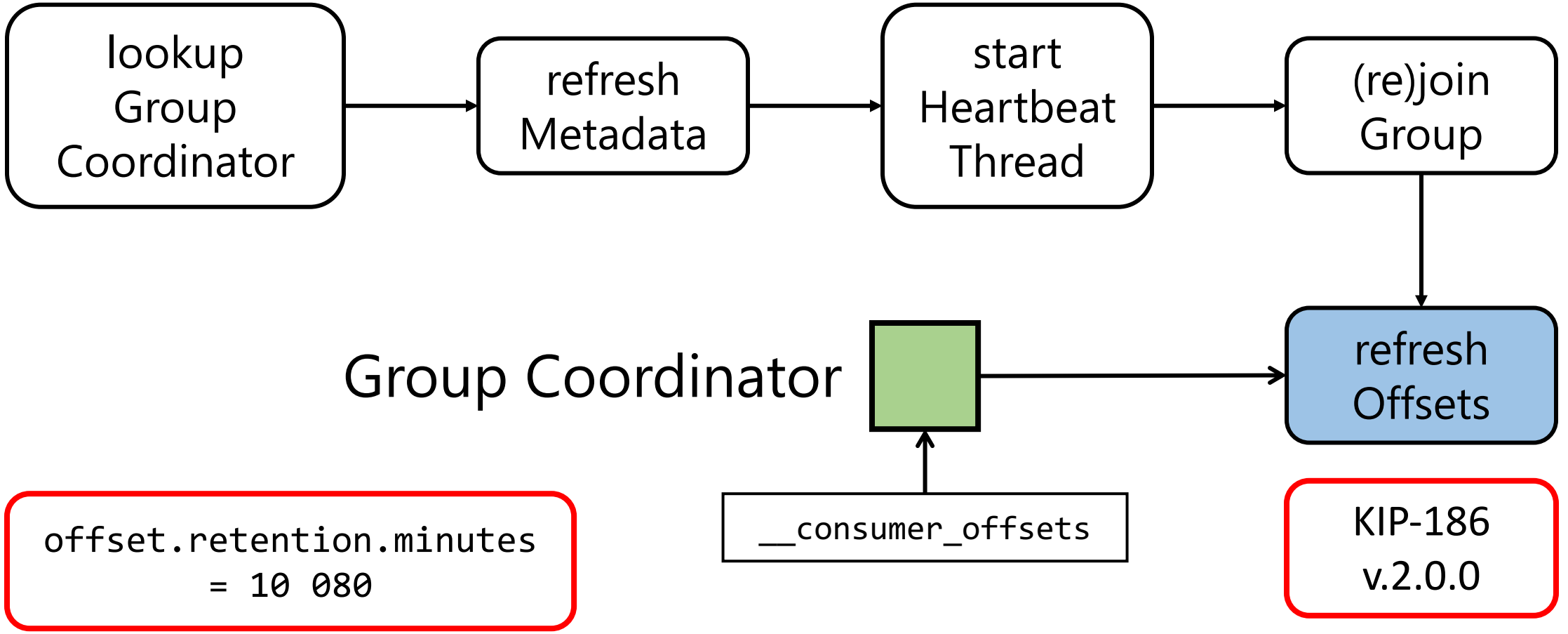
Consumer под капотом — coordinator



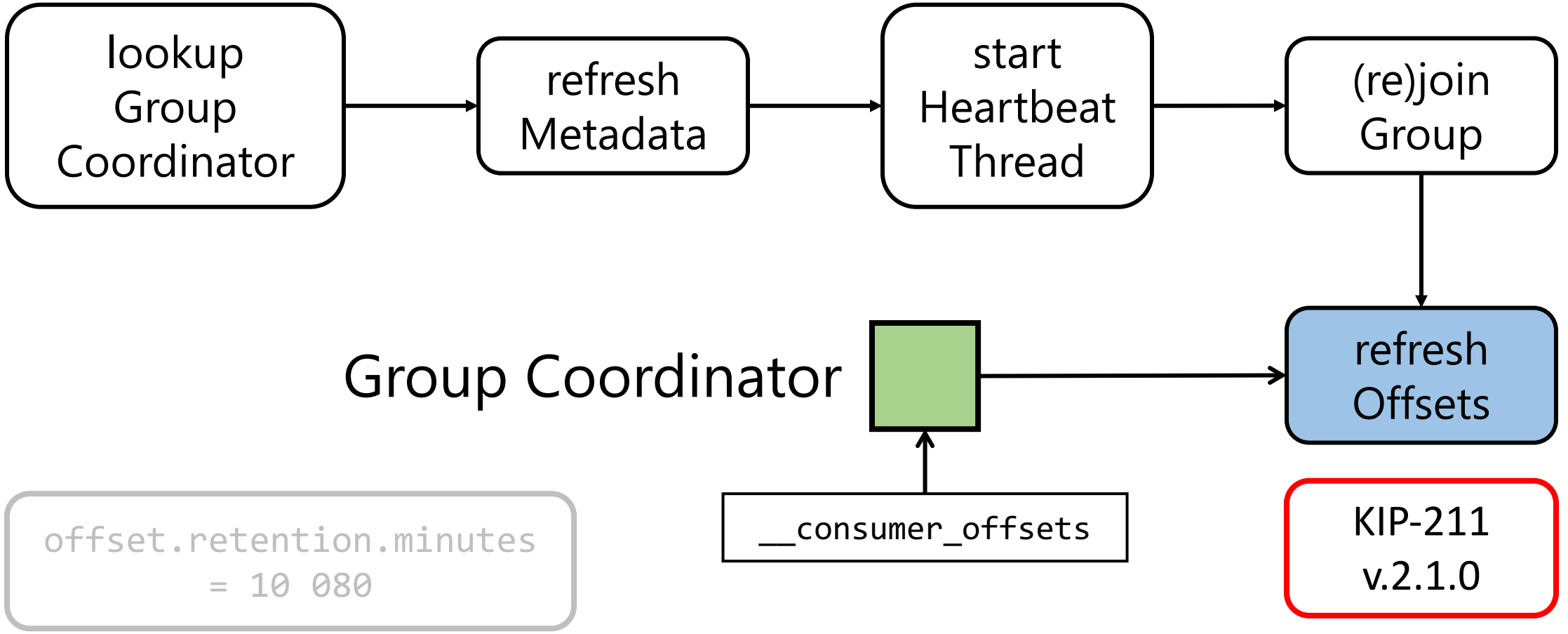
Consumer под капотом — coordinator



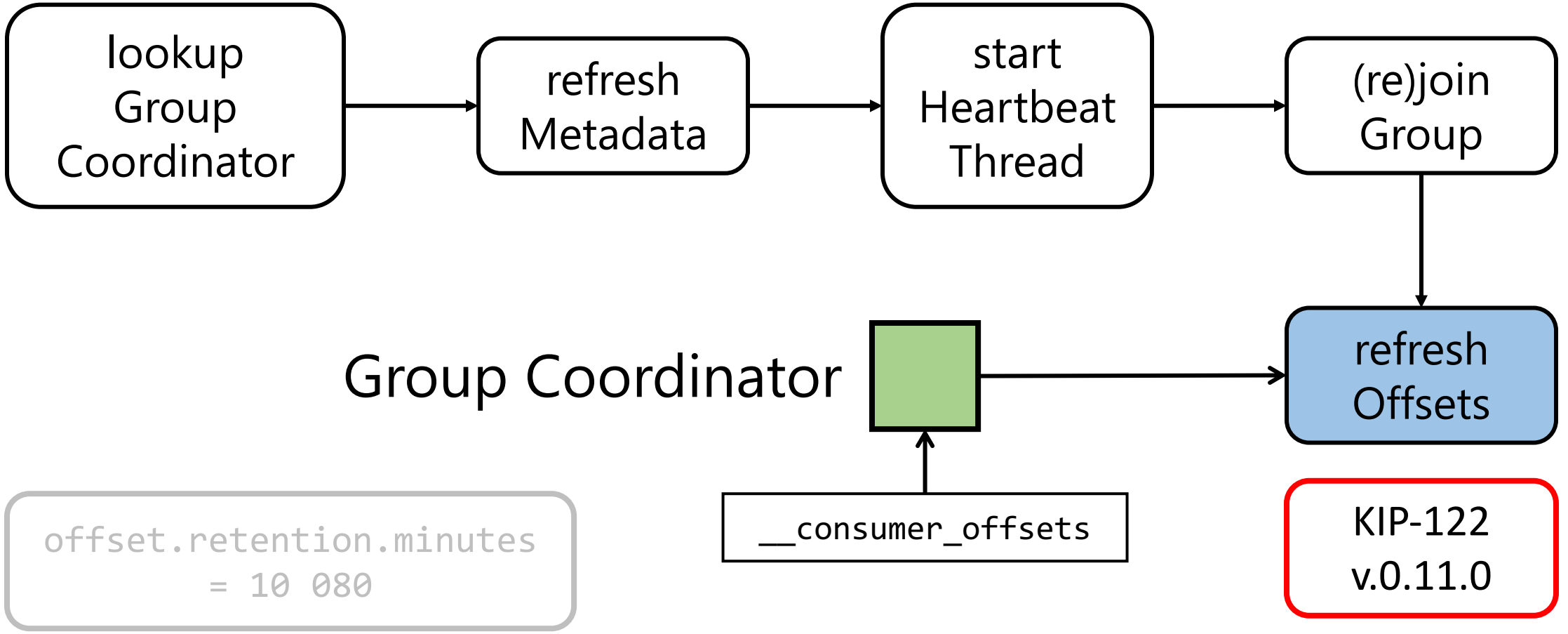
Consumer под капотом — coordinator



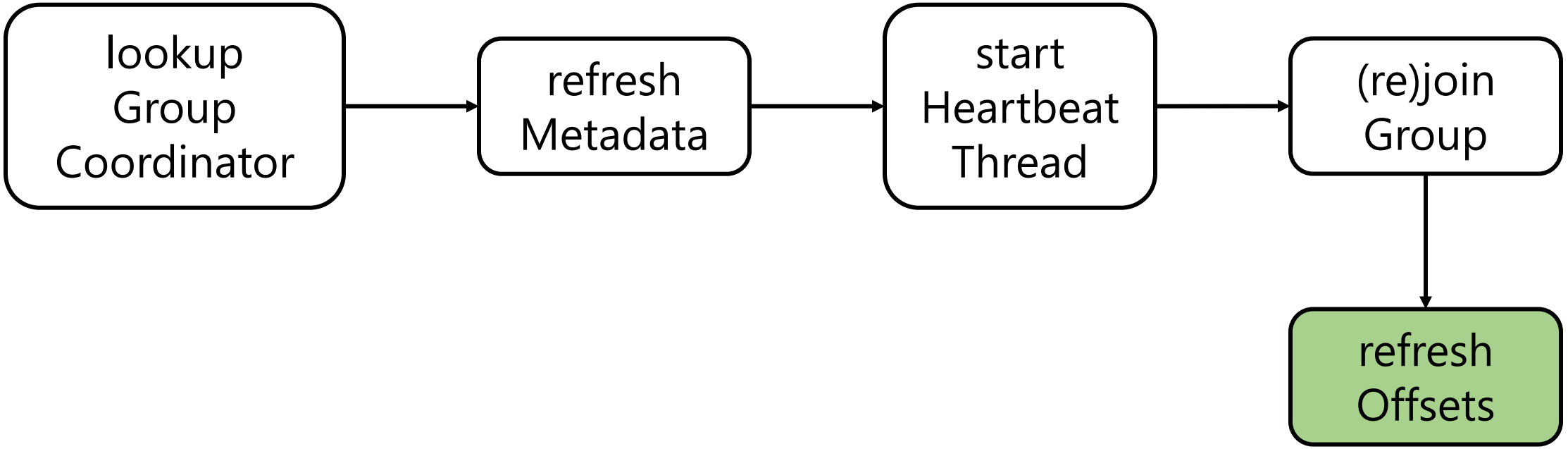
Consumer под капотом — coordinator



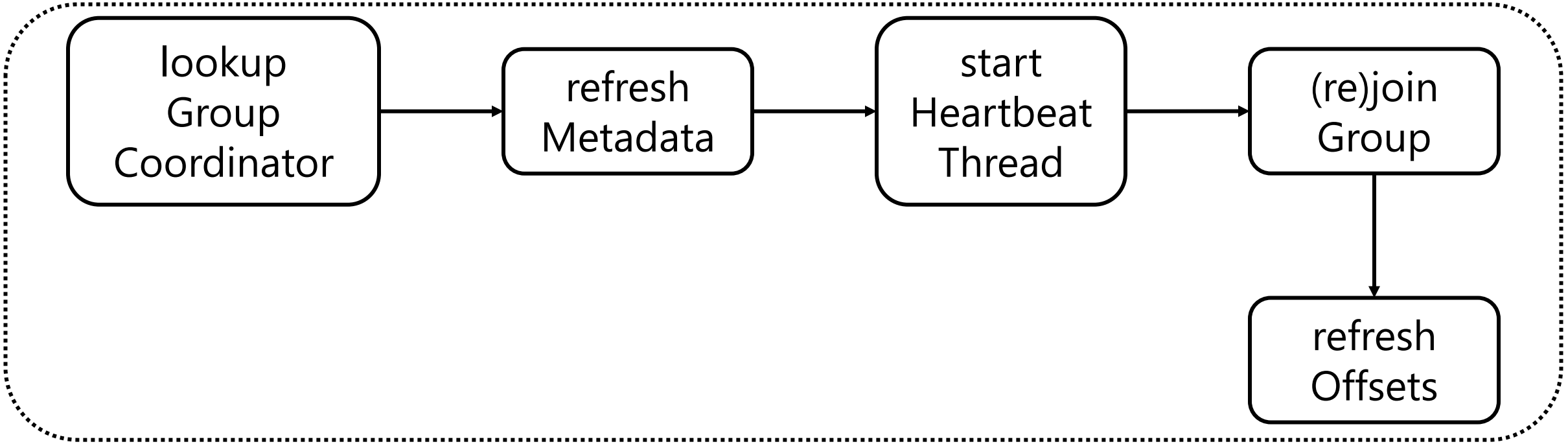
Consumer под капотом — coordinator



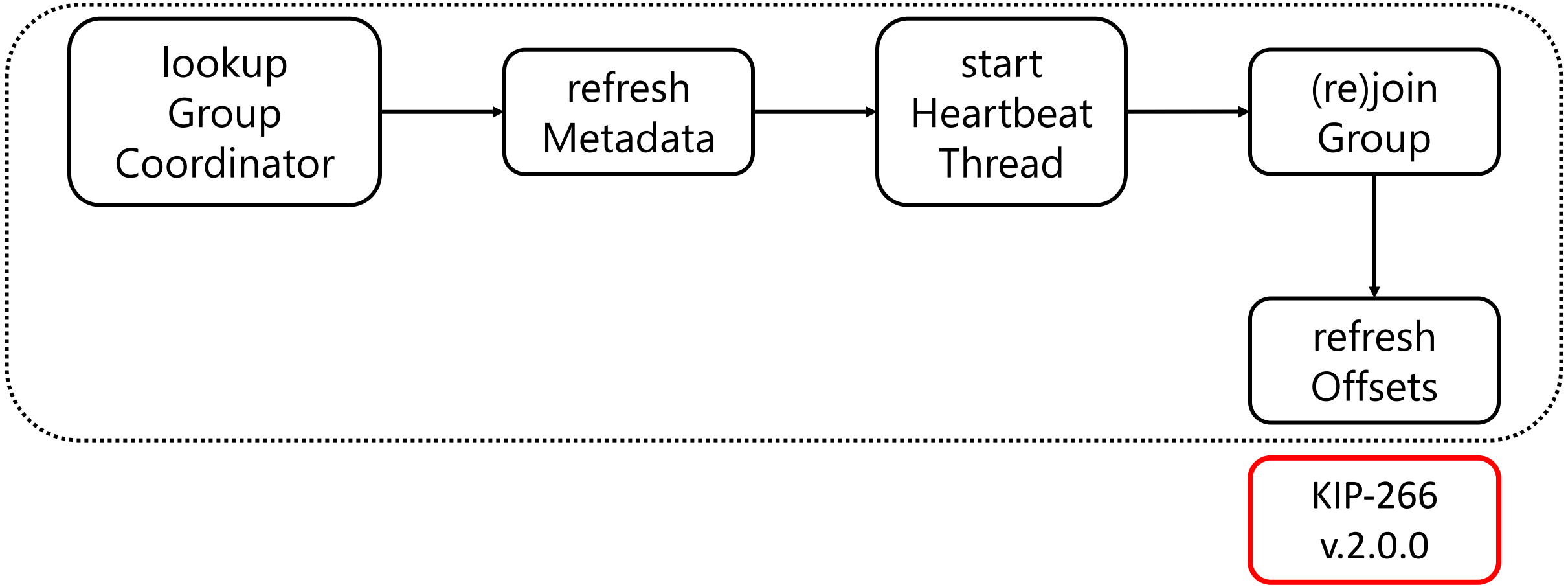
Consumer под капотом — coordinator



Consumer под капотом — coordinator

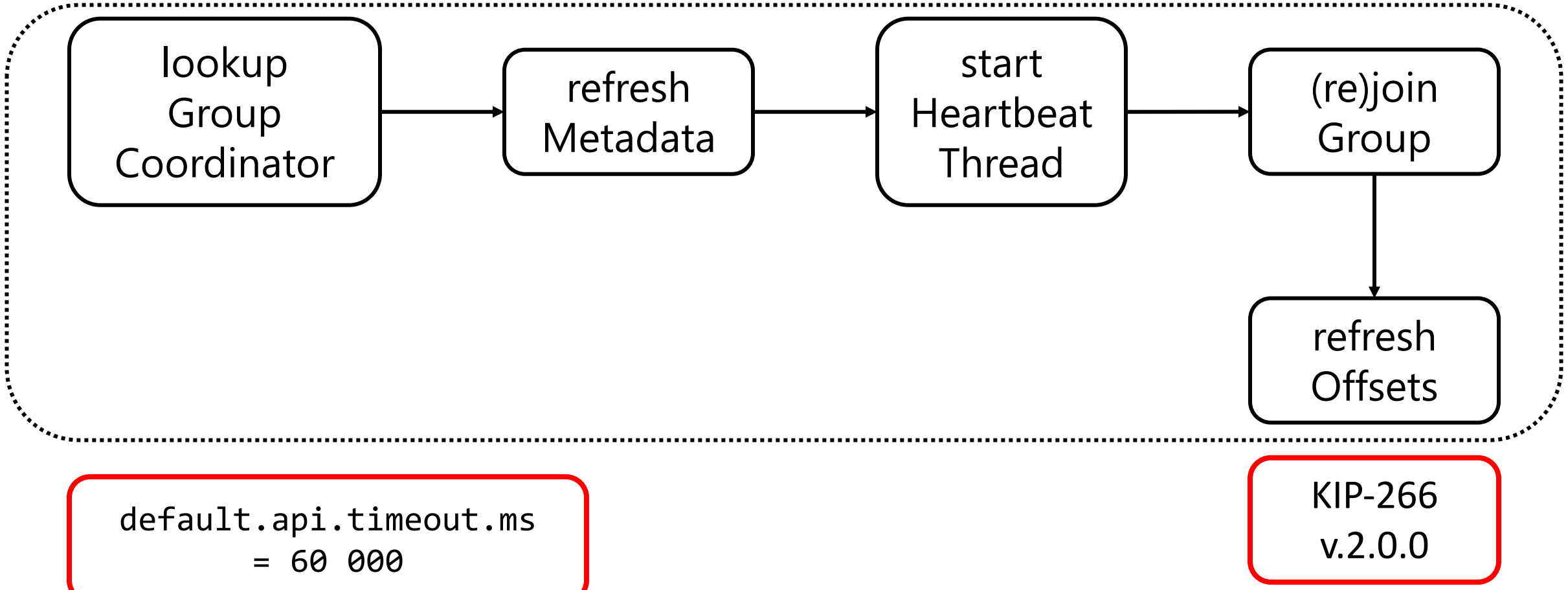


Consumer под капотом — coordinator

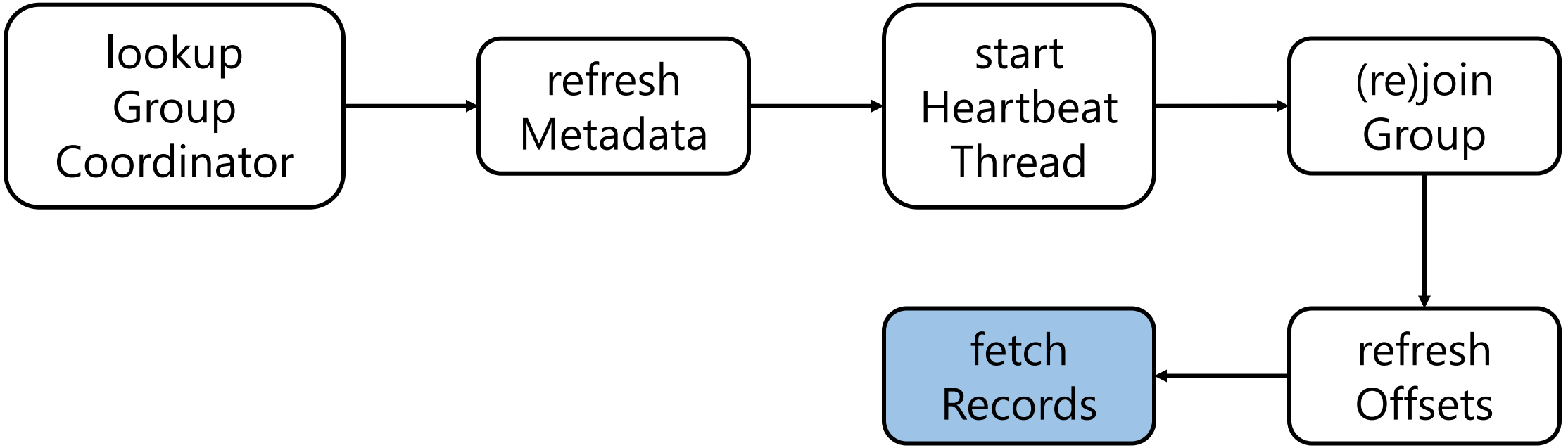


https://cwiki.apache.org/confluence/display/KAFKA/KIP-266%3A+Fix+consumer+indefinite+blocking+behavior

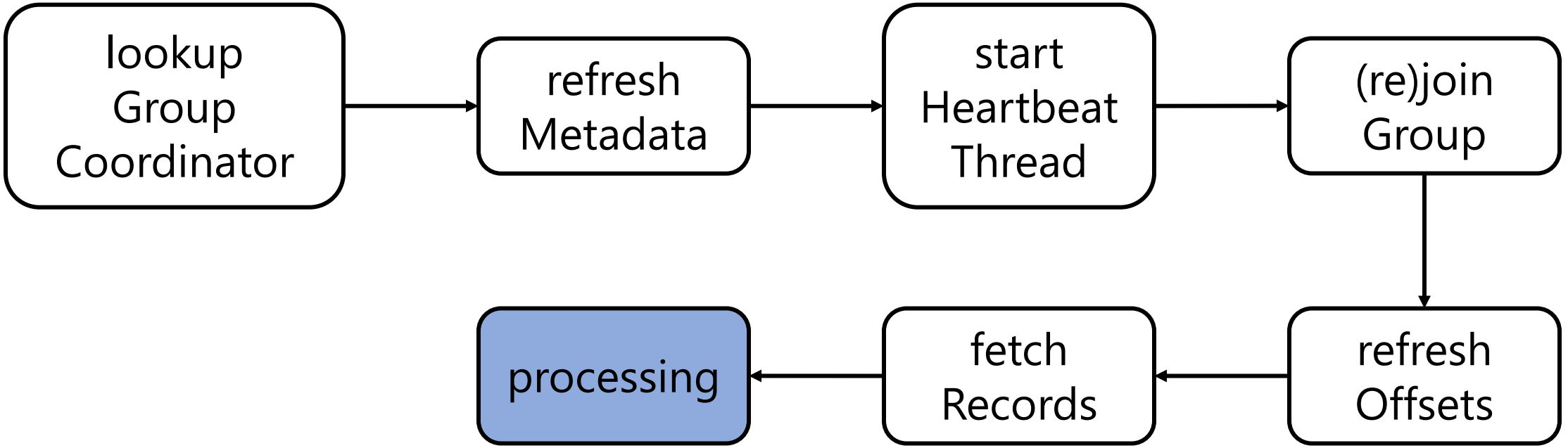
Consumer под капотом — coordinator



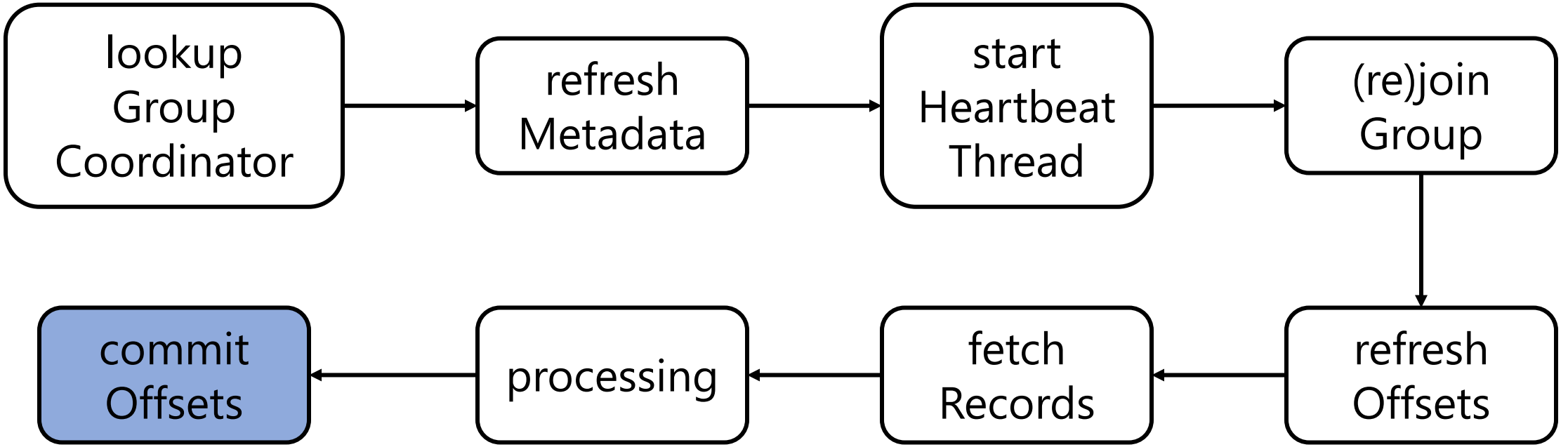
Consumer под капотом — coordinator



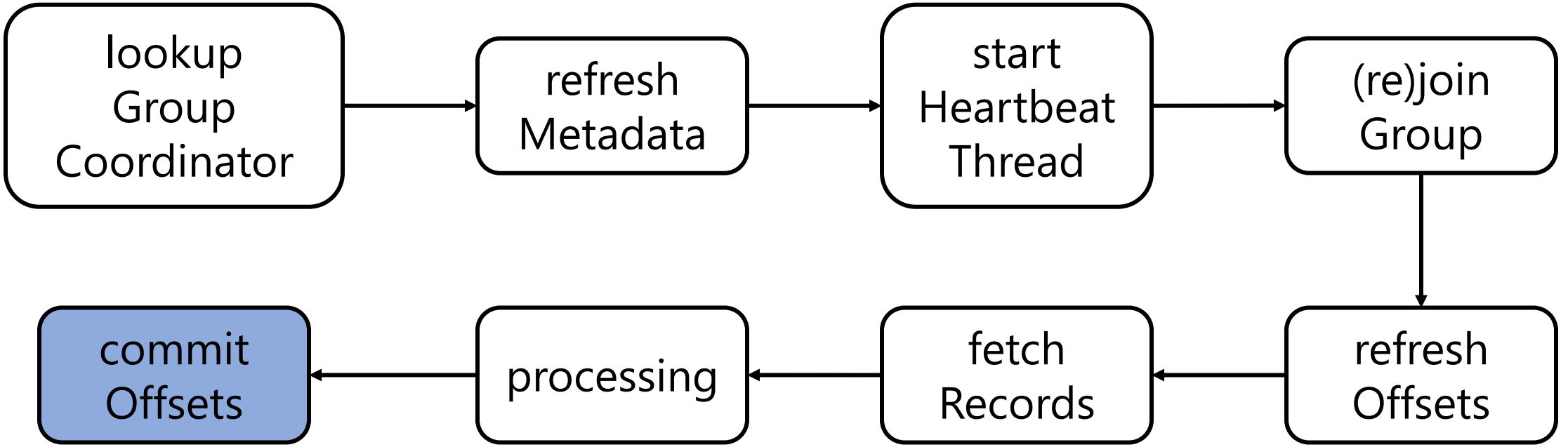
Consumer под капотом — coordinator



Consumer под капотом — coordinator

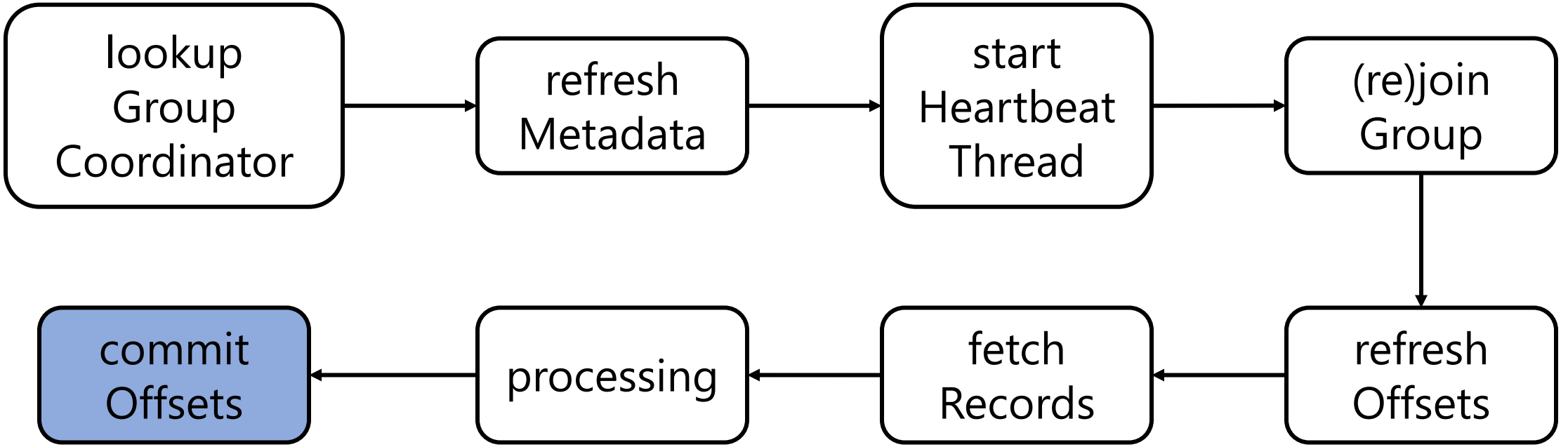


Consumer под капотом — coordinator



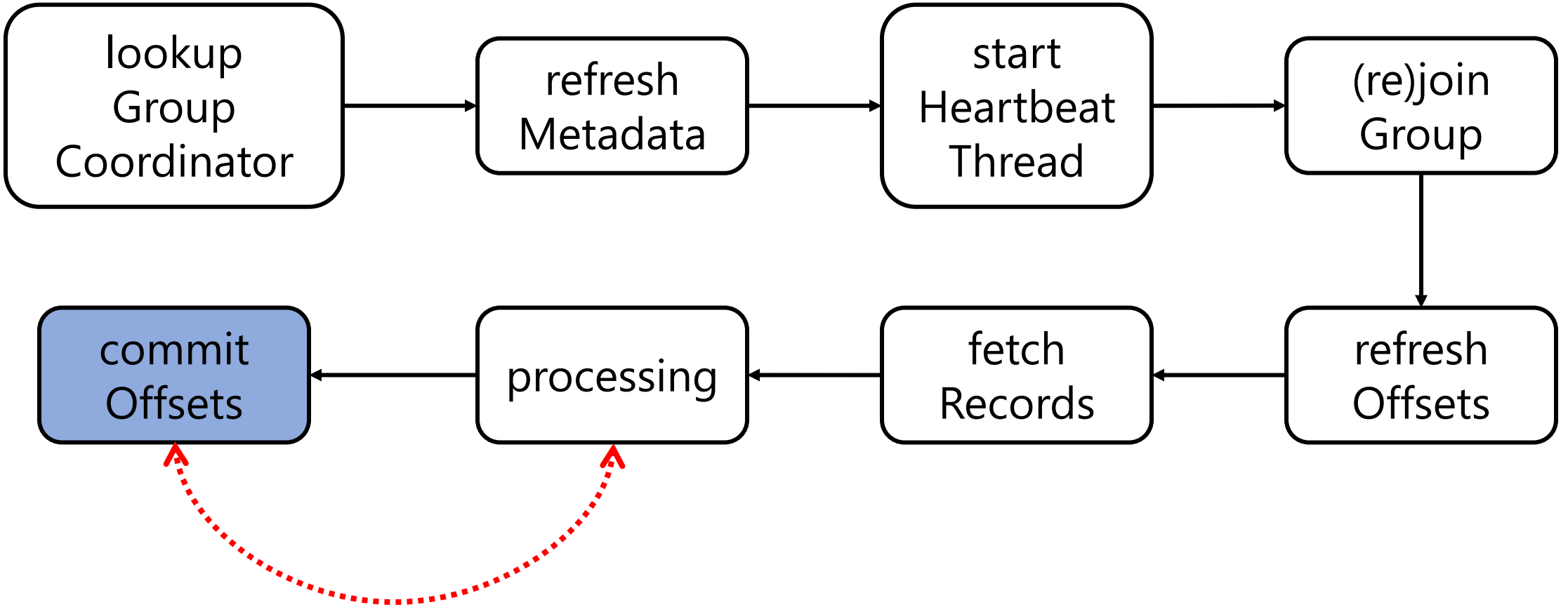

```
enable.auto.commit = true
auto.commit.interval.ms = 5 000
```

Consumer под капотом — coordinator

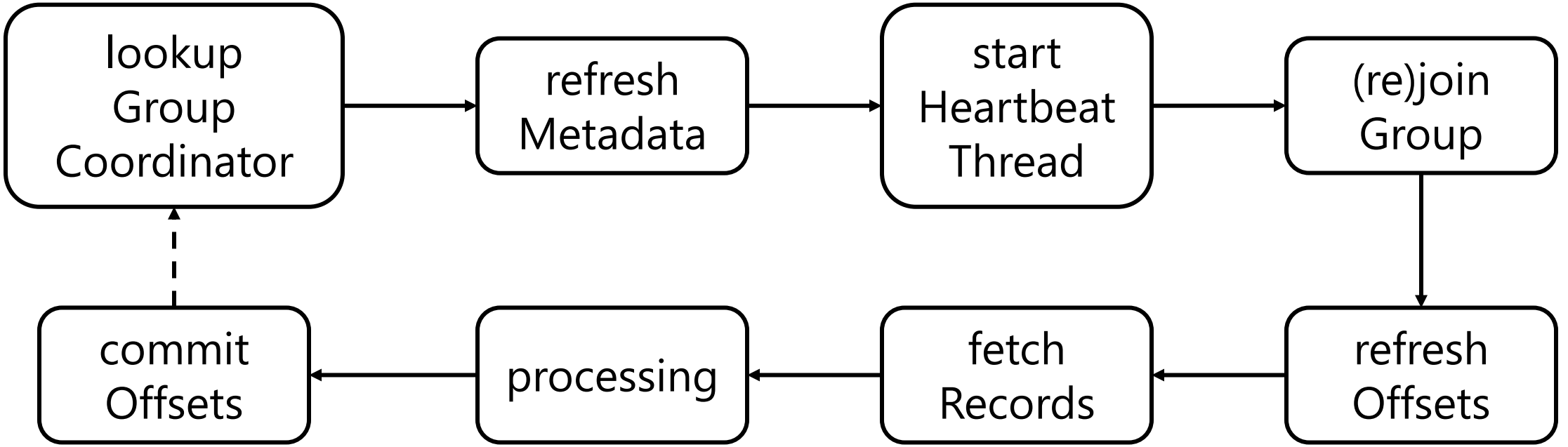



```
KafkaConsumer.commitSync
KafkaConsumer.commitAsync
```

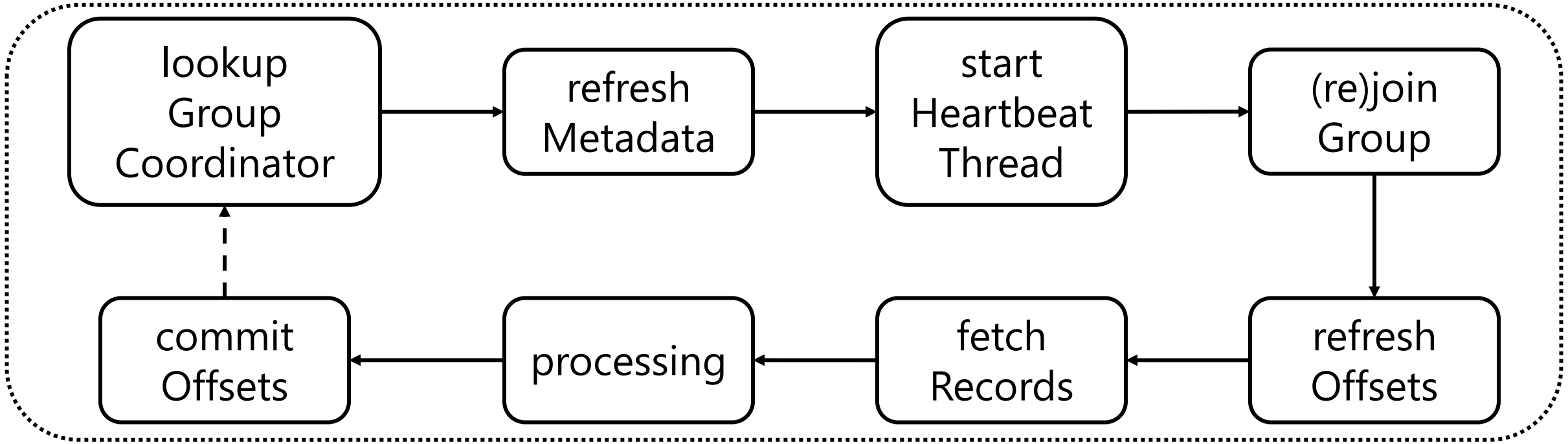
Consumer под капотом — coordinator



Consumer под капотом — coordinator



Consumer под капотом — coordinator



Consumer под капотом — coordinator

```
while (true) {
   /*  */
   ConsumerRecords<UUID, Event> records = consumer.poll(timer.toDuration());
   process(records);
   consumer.commitAsync();
   /*  */
}
```

Consumer под капотом — coordinator

```
while (true) {
   /* */
   ConsumerRecords<UUID, Event> records = consumer.poll(timer.toDuration());
   process(records);
   consumer.commitAsync();
   /* */
}
```

```
time-between-poll (avg, max)
    last-poll-seconds-ago
     poll-idle-ratio-avg
```

KIP-517
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-517%3A+Add+consumer+metrics+to+observe+user+poll+behavior

Consumer под капотом — coordinator

```
while (true) {
   /* */
   ConsumerRecords<UUID, Event> records = consumer.poll(timer.toDuration());
   process(records);
   consumer.commitAsync();
   /* */
}
```

```
time-between-poll (avg, max)
   last-poll-seconds-ago
    poll-idle-ratio-avg
```

KIP-517
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-517%3A+Add+consumer+metrics+to+observe+user+poll+behavior

Consumer под капотом — coordinator

```
while (true) {
   /* */
   ConsumerRecords<UUID, Event> records = consumer.poll(timer.toDuration());
   process(records);
   consumer.commitAsync();
   /* */
}
```

$$0.0 < \frac{\text{время внутри } poll}{\text{общее время}} < 1.0$$

```
time-between-poll (avg, max)
   last-poll-seconds-ago
    poll-idle-ratio-avg
```

KIP-517
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-517%3A+Add+consumer+metrics+to+observe+user+poll+behavior

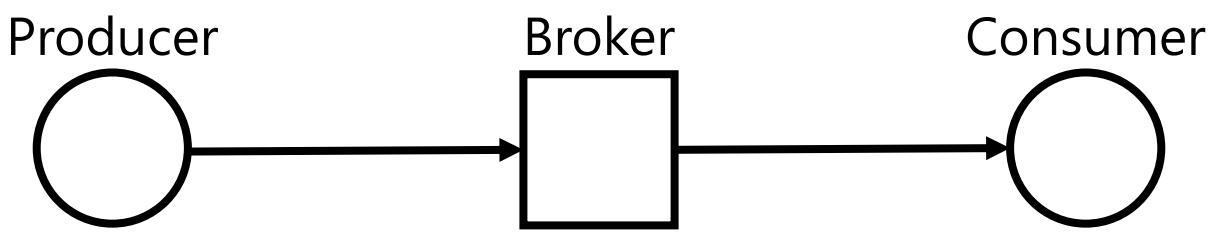
Про перформанс на посошок

Про перформанс на посошок

— down conversion

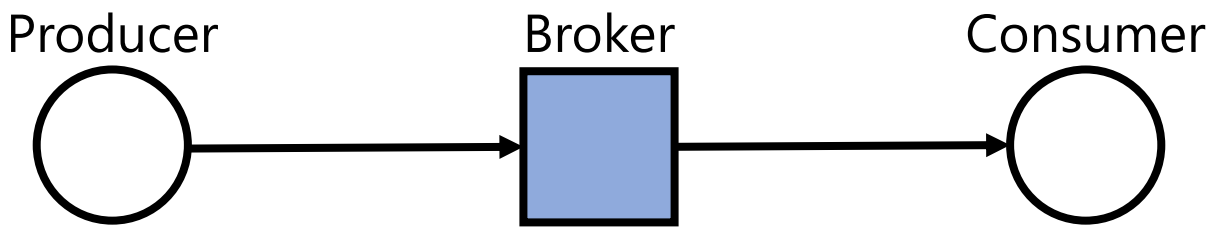
Про перформанс на посошок

— down conversion



Про перформанс на посошок

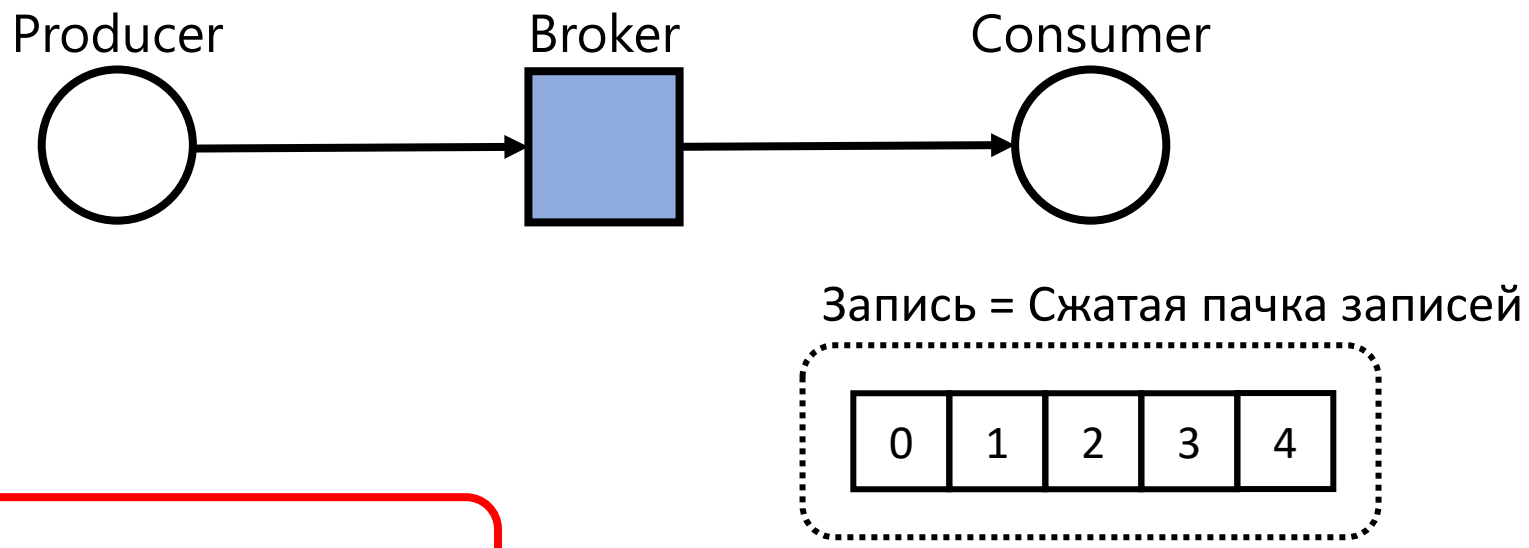
— down conversion




```
log.message.format.version
```

Про перформанс на посошок

— down conversion



`log.message.format.version`

Про перформанс на посошок

— down conversion

KIP-283
v.2.0.0

Про перформанс на посошок

— down conversion
— compression

Gzip, LZ4, Snappy

Про перформанс на посошок

— down conversion
— compression

Gzip, LZ4, Snappy, Zstd

KIP-110
v.2.1.0

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group

KIP-389
v.2.2.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-389%3A+Introduce+a+configurable+consumer+group+size+limit

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group
— static membership

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group
— static membership

```
group.instance.id = null
```

KIP-345
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-345%3A+Introduce+static+membership+protocol+to+reduce+consumer+rebalances

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group
— static membership

```
group.instance.id = "id"
session.timeout.ms
```

KIP-345
v.2.4.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-345%3A+Introduce+static+membership+protocol+to+reduce+consumer+rebalances

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group
— static membership
— rack-aware partition assignment

Про перформанс на посошок

— down conversion
— compression
— размер Consumer Group
— static membership
— rack-aware partition assignment

KIP-881
3.5.0

https://cwiki.apache.org/confluence/display/KAFKA/KIP-848%3A+The+Next+Generation+of+the+Consumer+Rebalance+Protocol

Про перформанс на посошок

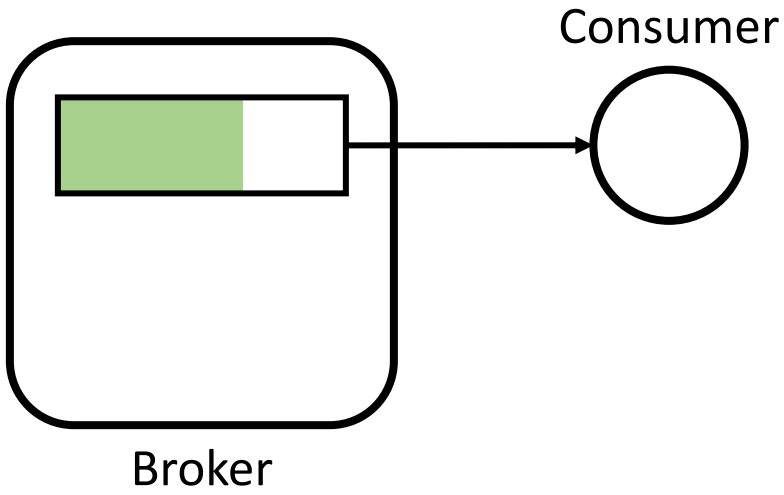
— down conversion
— compression
— размер Consumer Group
— static membership
— rack-aware partition assignment
— Next Gen consumer rebalance protocol

Про перформанс на посошок

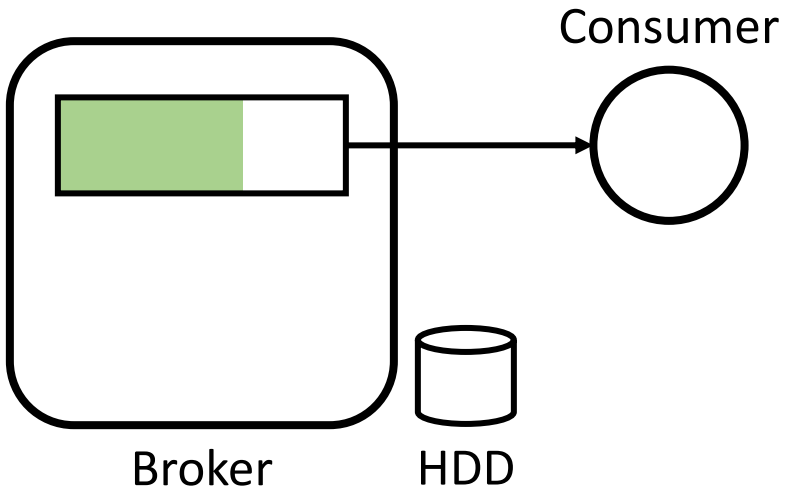
— down conversion
— compression
— размер Consumer Group
— static membership
— rack-aware partition assignment
— Next Gen consumer rebalance protocol

KIP-848
WIP

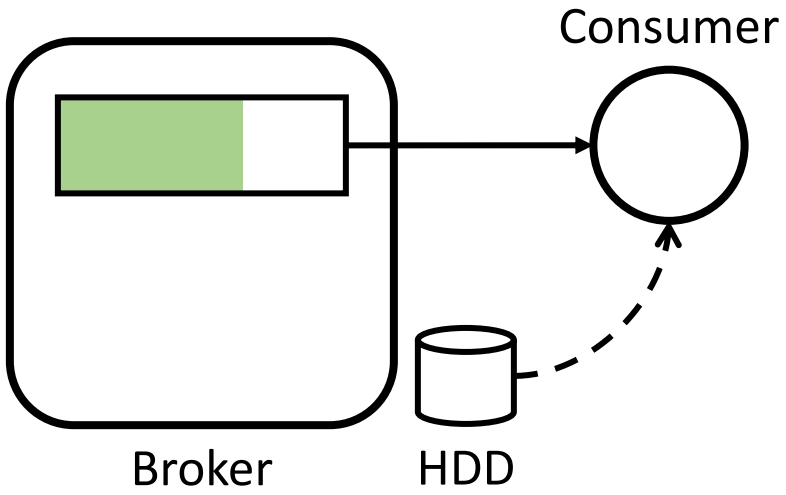
Про перформанс на посошок



Про перформанс на посошок

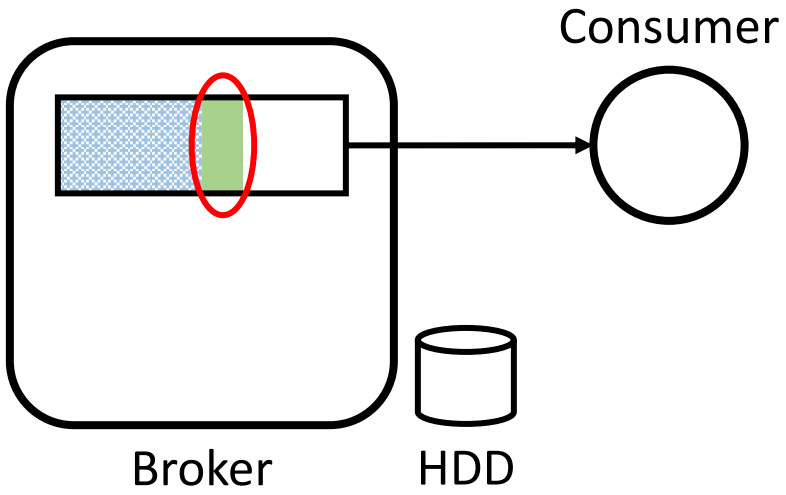


Про перформанс на посошок

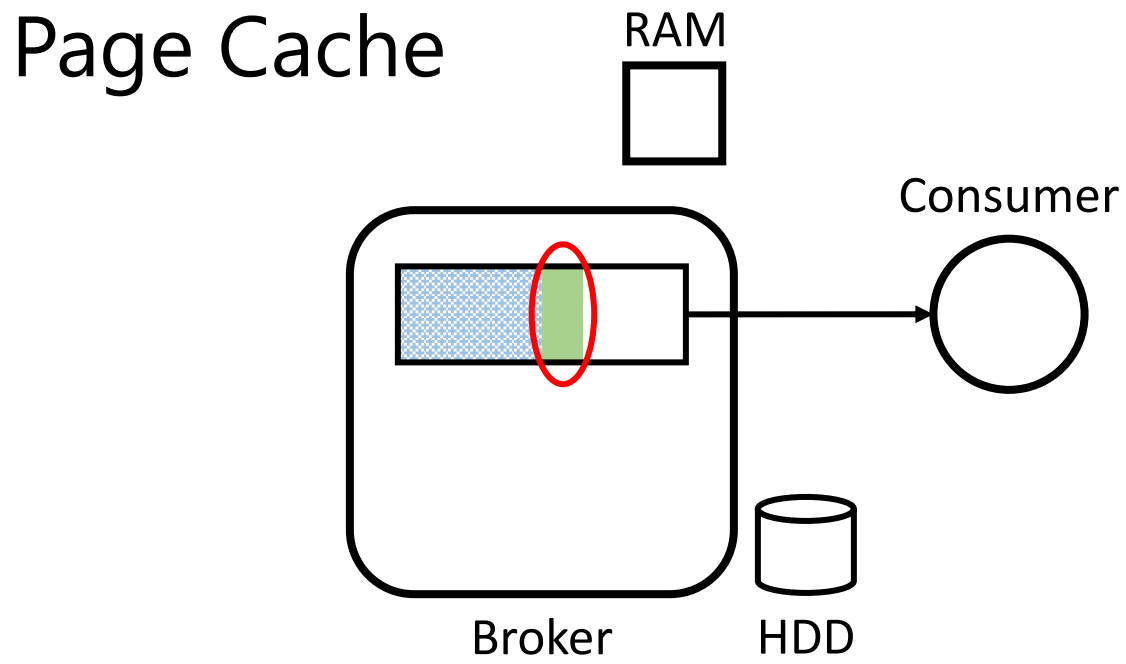


Про перформанс на посошок

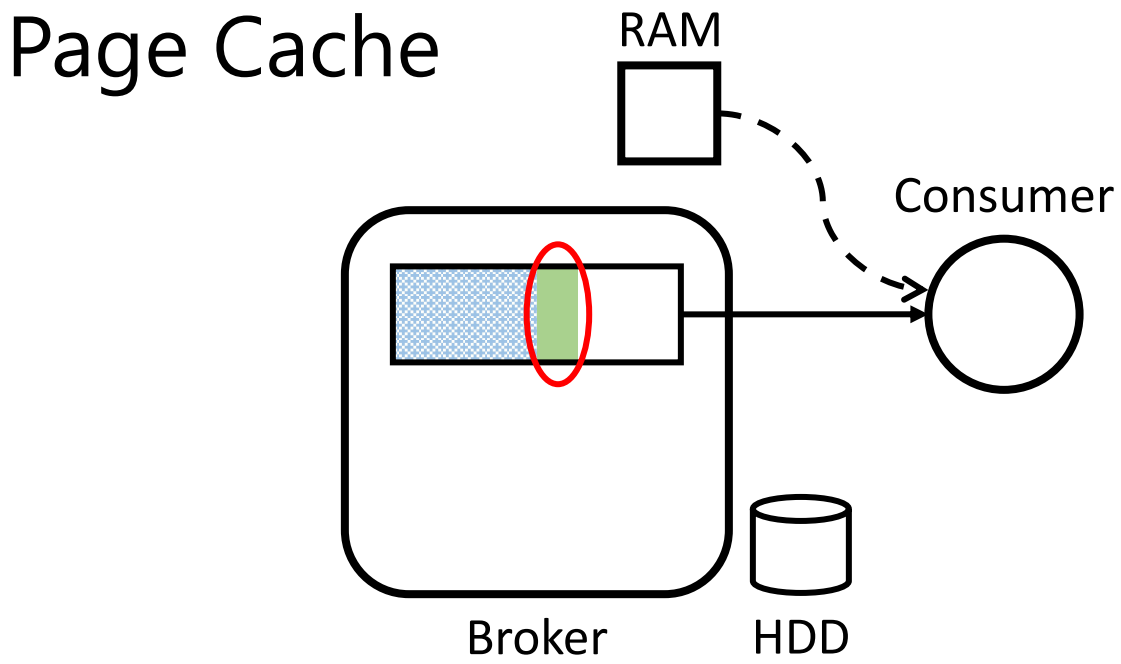
Page Cache



Про перформанс на посошок



Про перформанс на посошок



Q/A

Другие доклады и материалы:
https://t.me/chnl_GregoryKoshelev