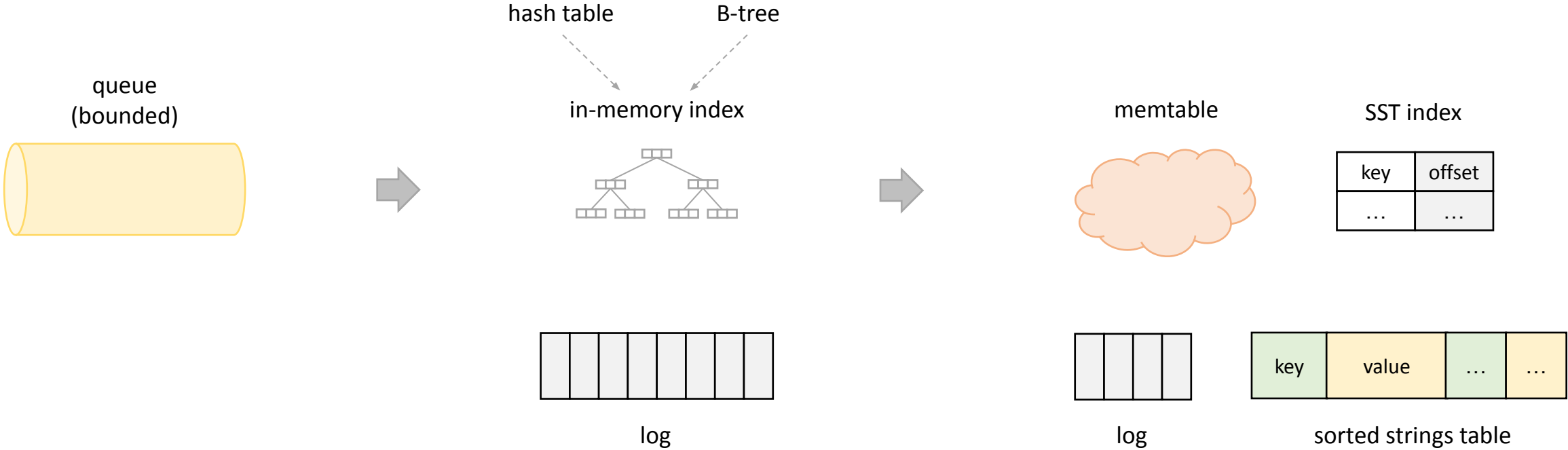
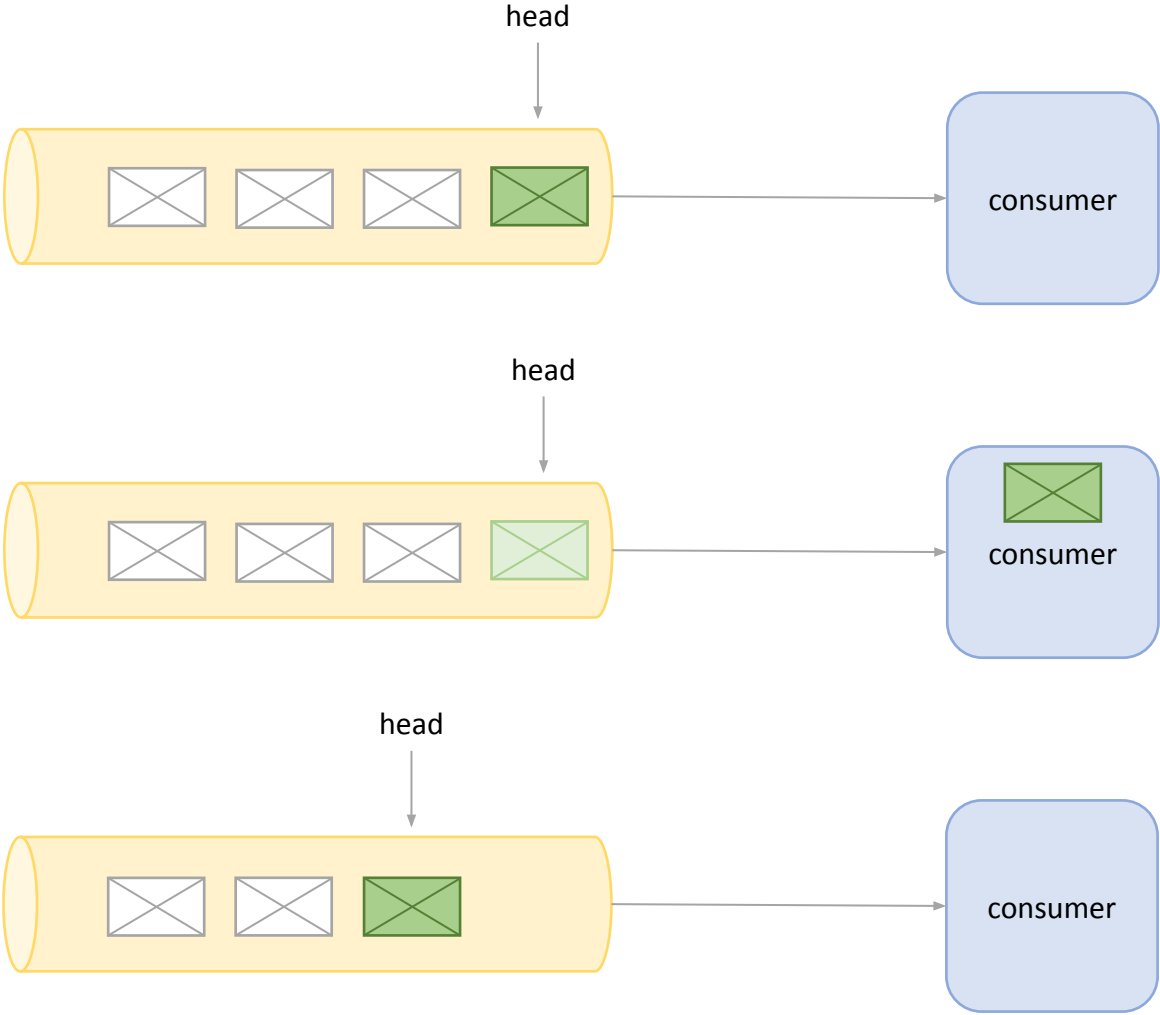


Consumer offsets



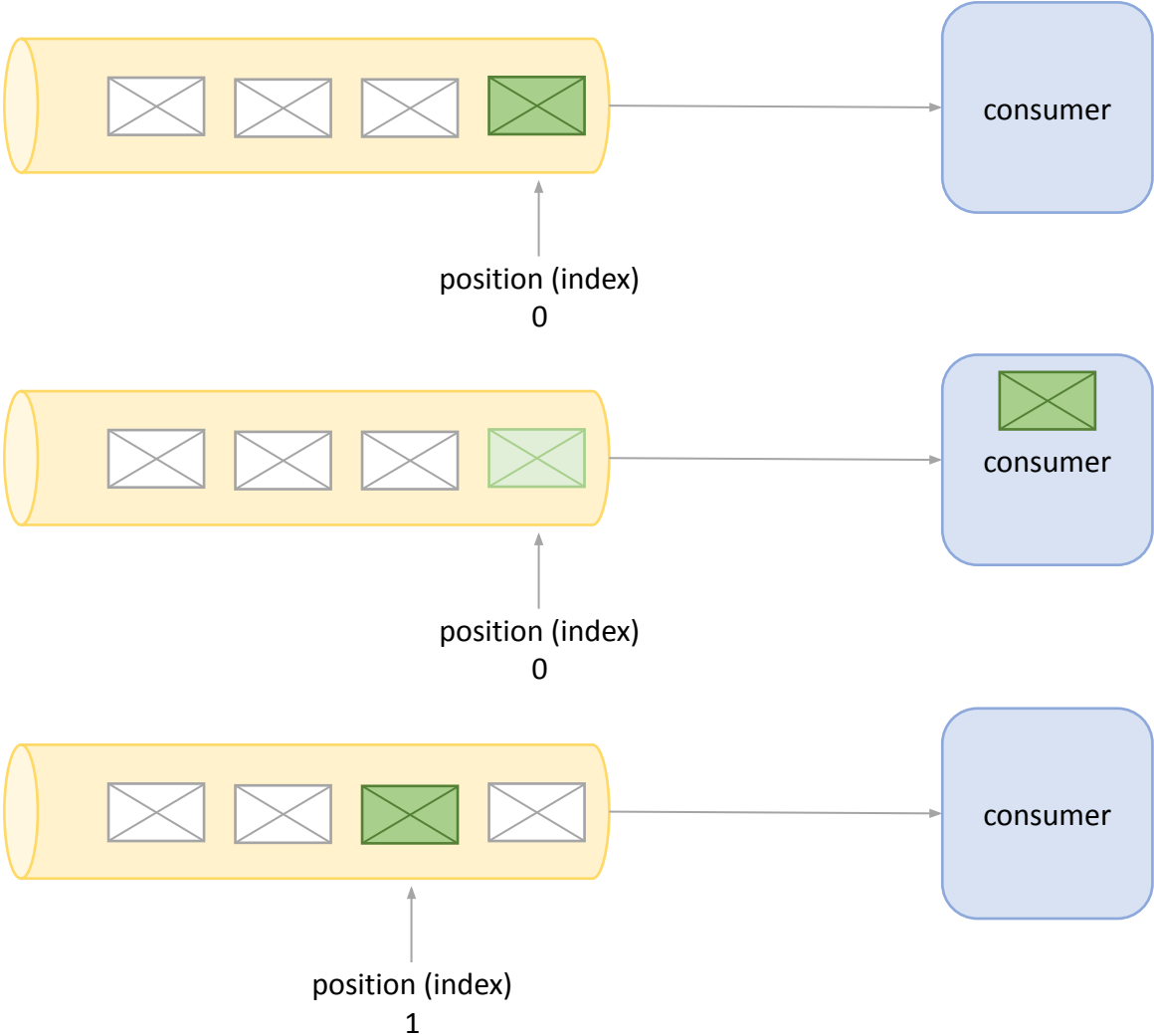
Consumer offsets

option 1
delete message when processed



Consumer offsets

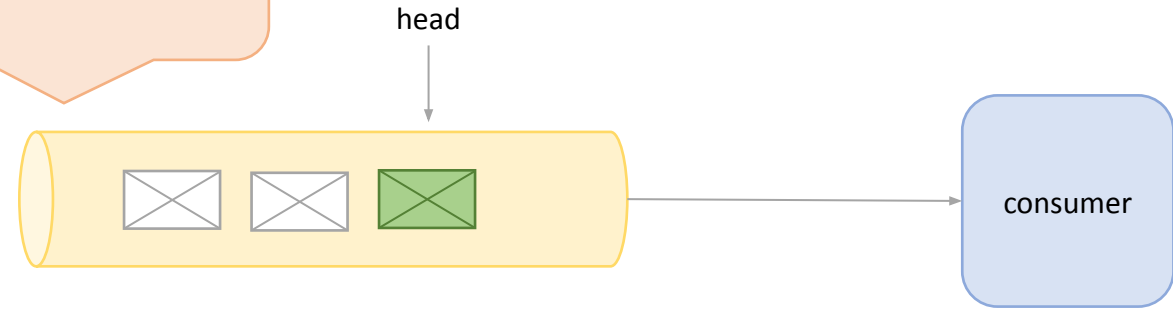
option 2
increment message position



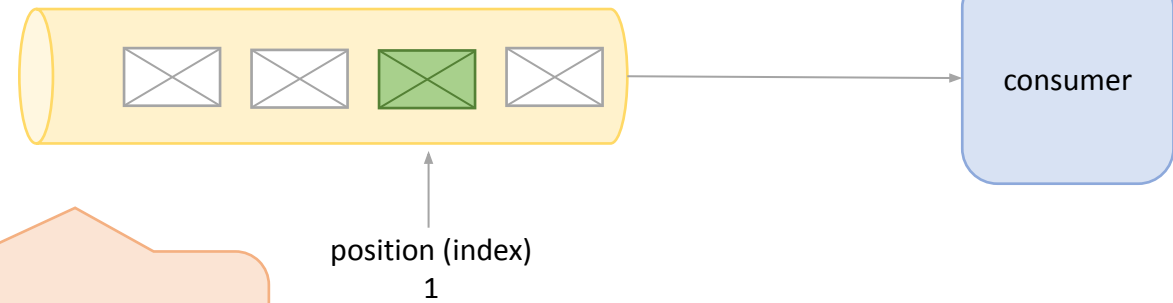
Consumer offsets

I need to do a lot of heavy lifting: track message status, process acks, delete message, ...

option 1
delete message when processed
(ActiveMQ, RabbitMQ, SQS)



option 2
increment message position
(Kafka, Kinesis)

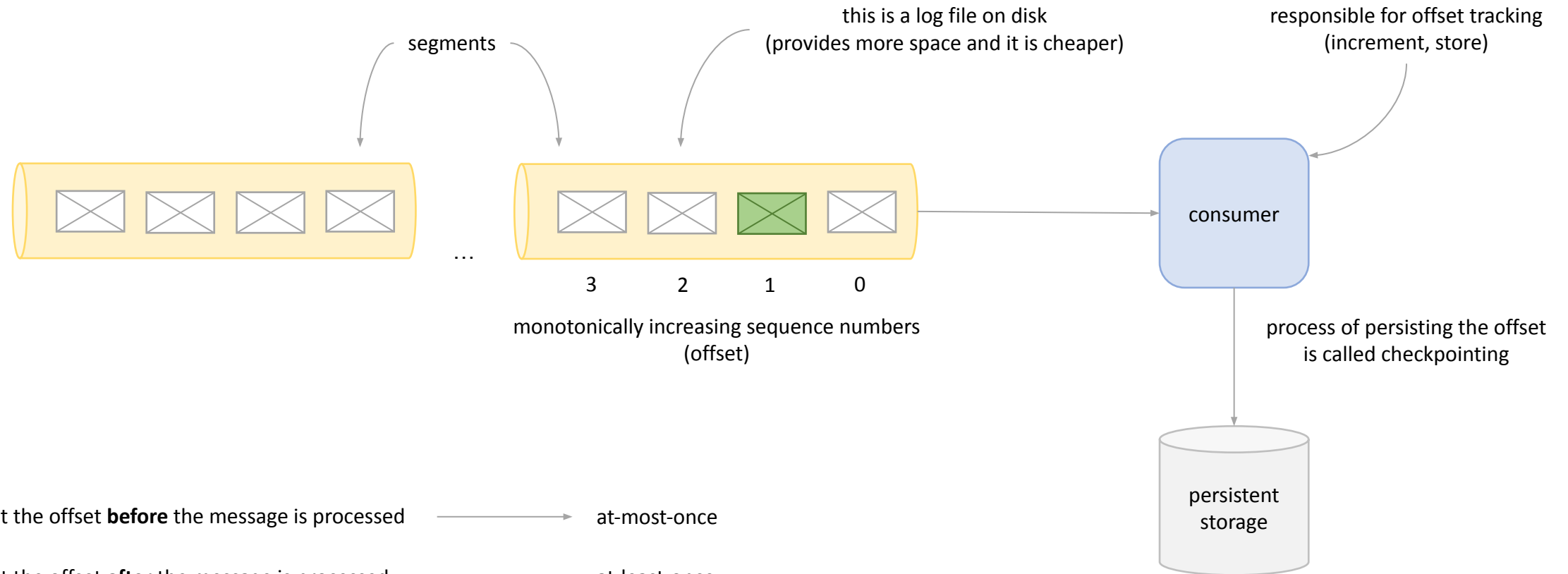


My job is easy: retrieve the message at the requested position and return it.

Consumer offsets

messaging systems that follow this approach are often referred to as **log-based messaging systems**

option 2 details



- checkpoint the offset **before** the message is processed → at-most-once
- checkpoint the offset **after** the message is processed → at-least-once
exactly-once
(if consumer is idempotent)

Consumer offsets

option 1 is reliable, but option 2 is even more reliable

