DER KAFKA HYPE – SONST NIX?

Thomas Müller | Java Forum Stuttgart 2021

sidion

ABOUT ME…

- Thomas Müller
- Java-Entwickler seit 2003
- Bis ca. 2009 Fullstack-Entwickler
- Danach Fokus auf Backend- und Middleware-Systeme
- Schwerpunkte:
  - Messager-Broker
  - Streaming von Daten
- Seit 2009 in agilen Projekten unterwegs
- In der Freizeit Raspberry PI-Projekte

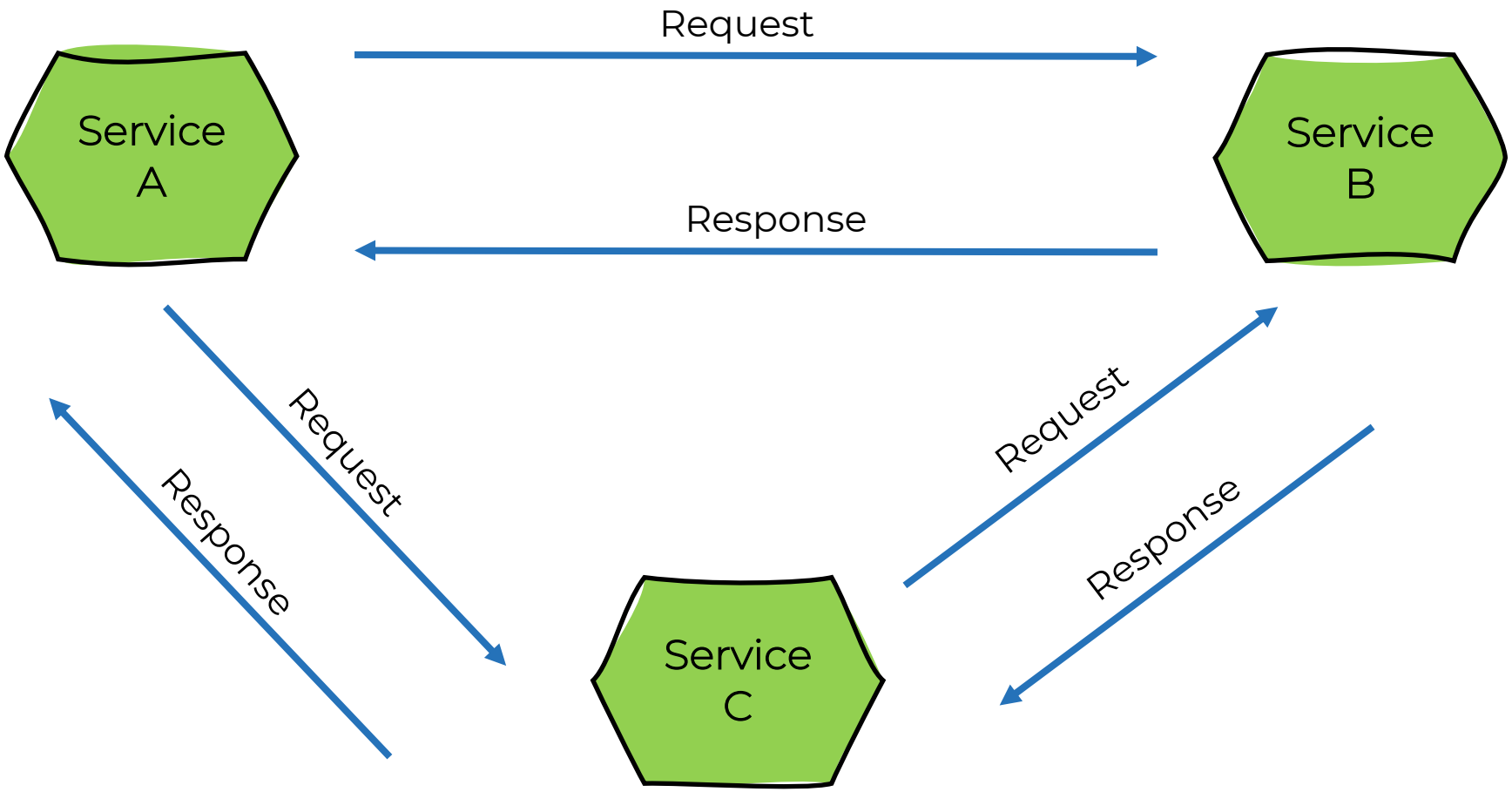
sidion

AGENDA

1. About Me
2. Funktionsweise Message Broker
3. Kafka
4. RabbitMQ
5. RedisMQ und Hazelcast
6. Zusammenfassung

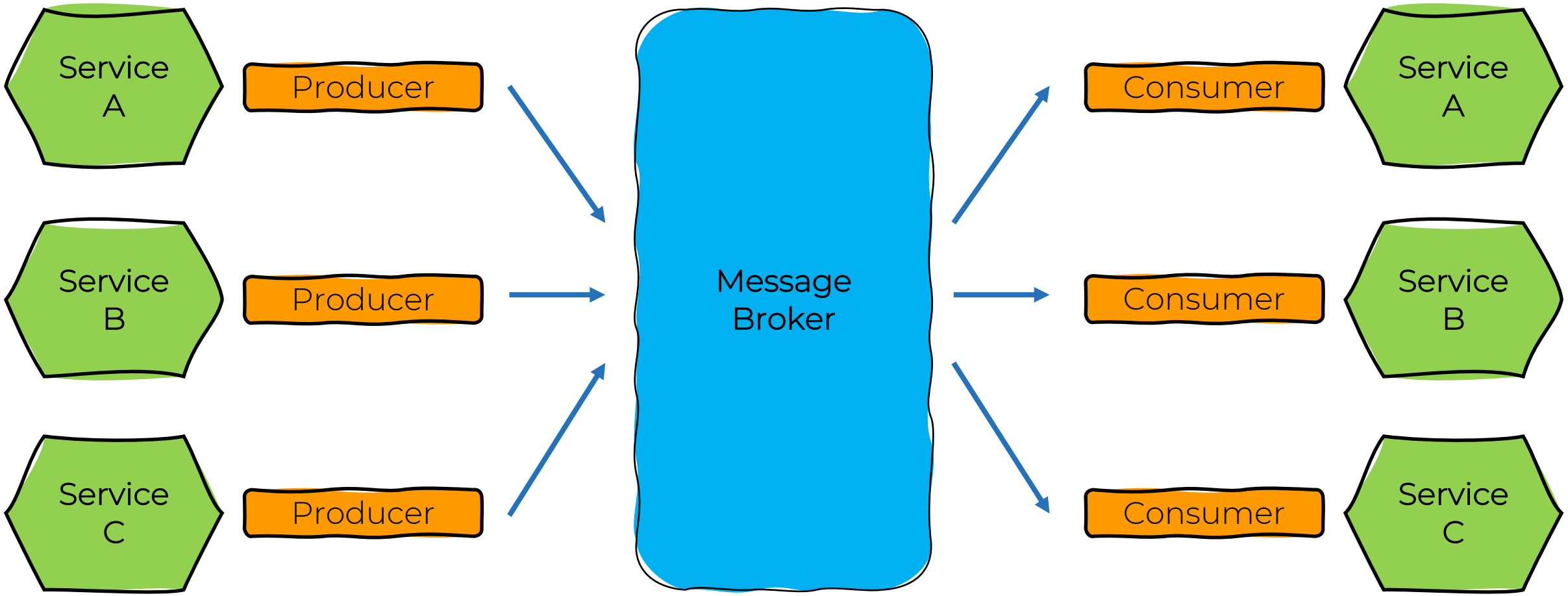
sidion

REQUEST-RESPONSE



sidion

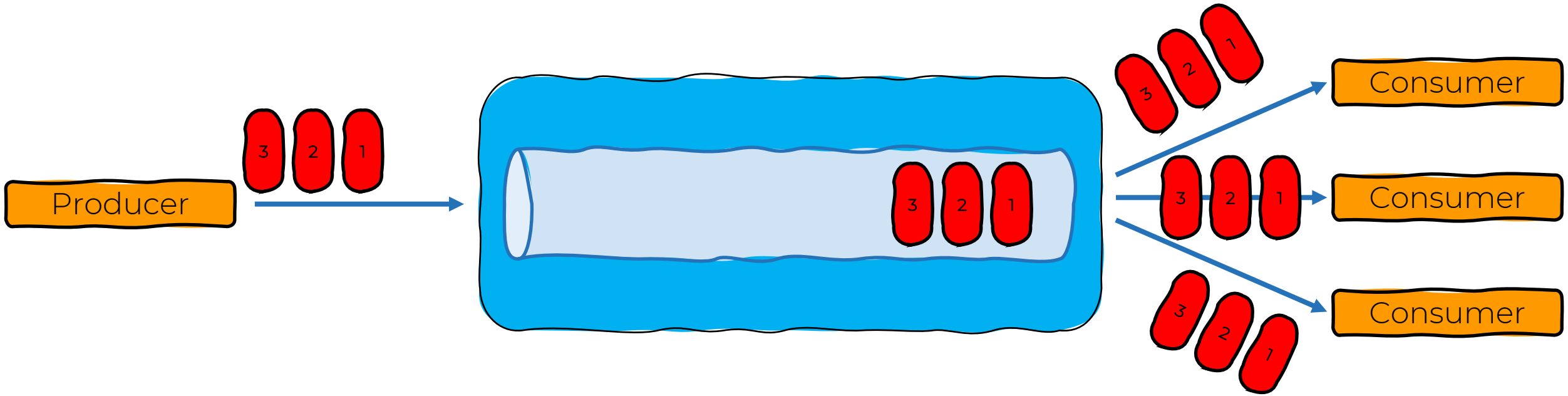
MESSAGE BROKER



sidion

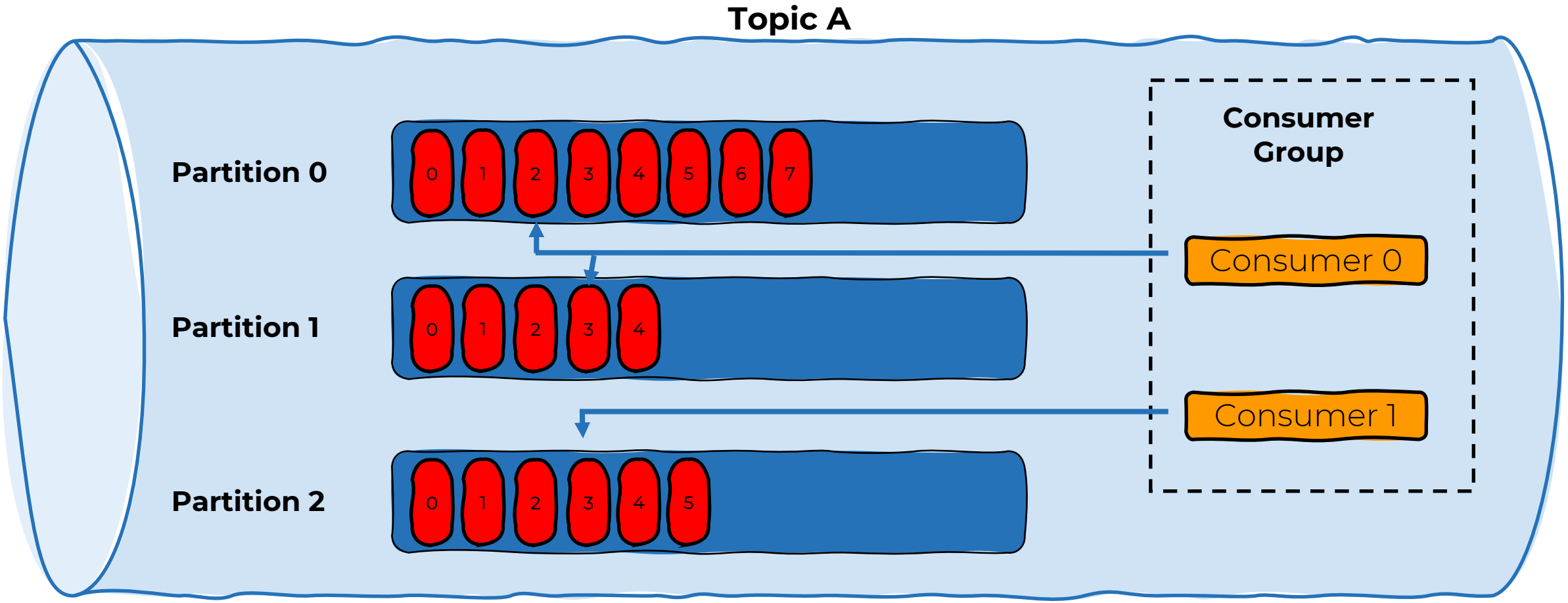
NACHRICHTENVERTEILUNG

KAFKA TOPIC



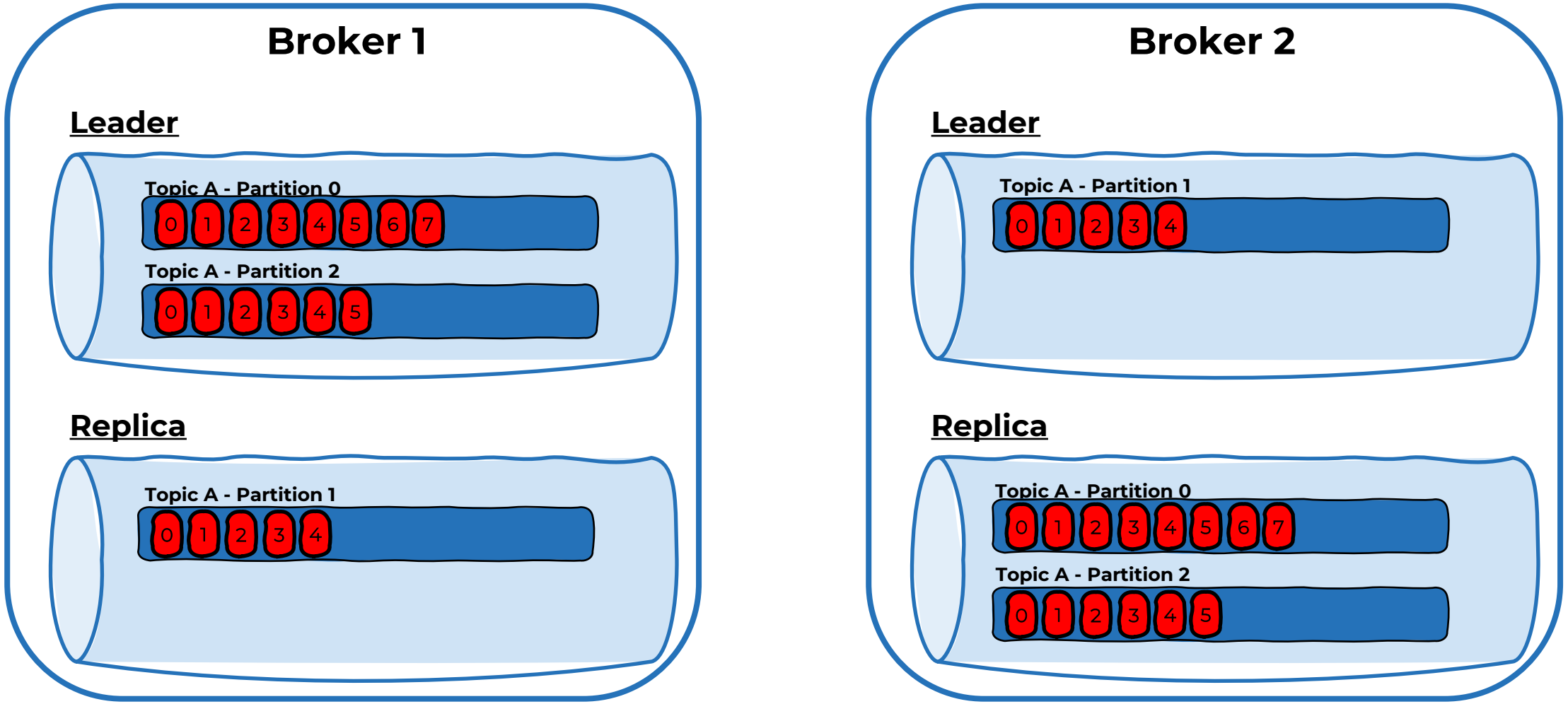
sidion

KAFKA TOPIC - PARTITIONEN



sidion

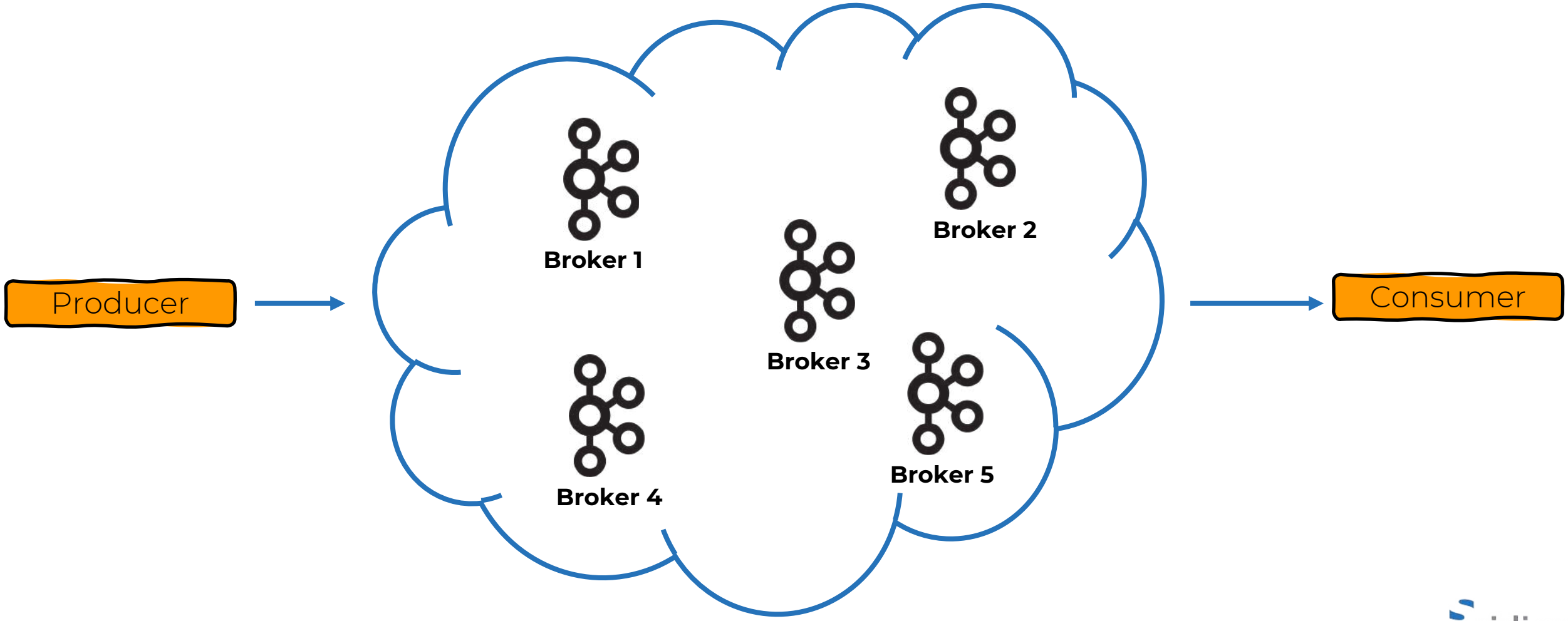
KAFKA REPLIKATION



sidion

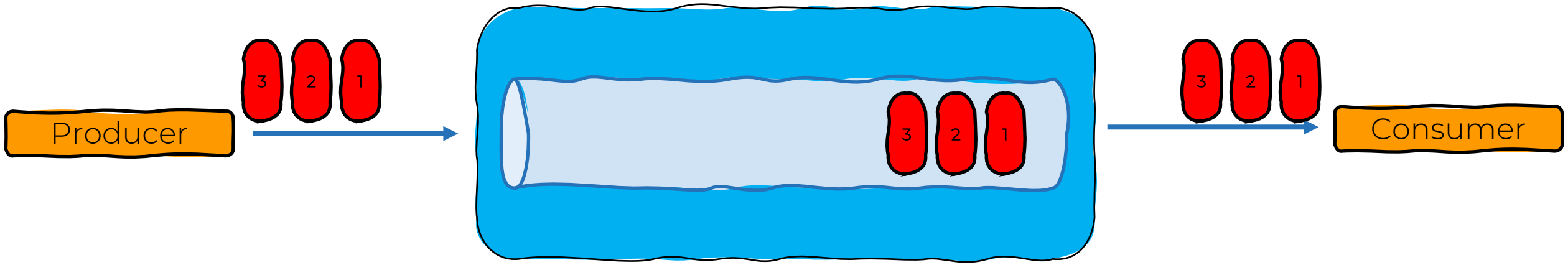
CLUSTERING

KAFKA CLUSTER



sidion

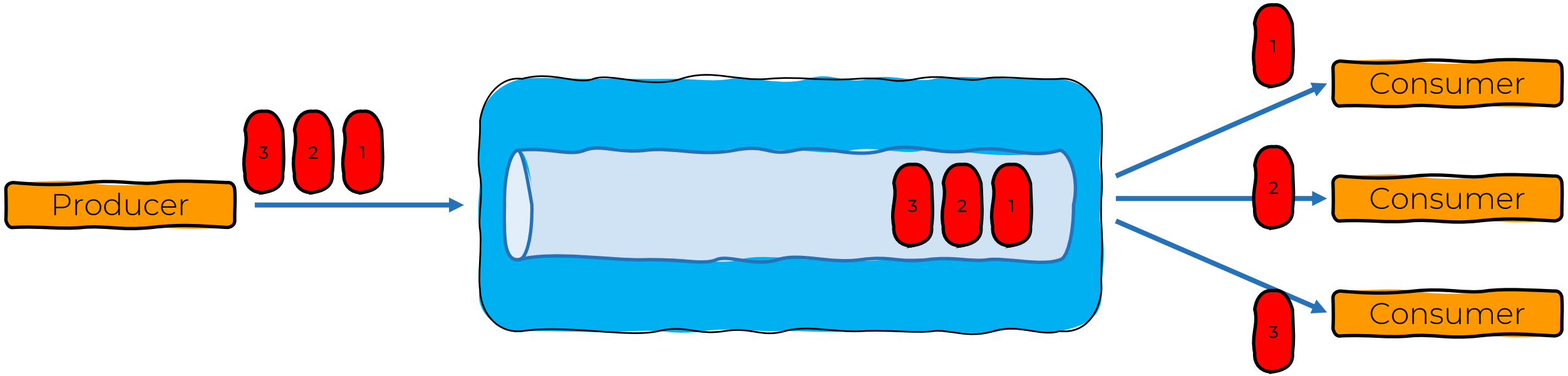
MESSAGE QUEUE



sidion

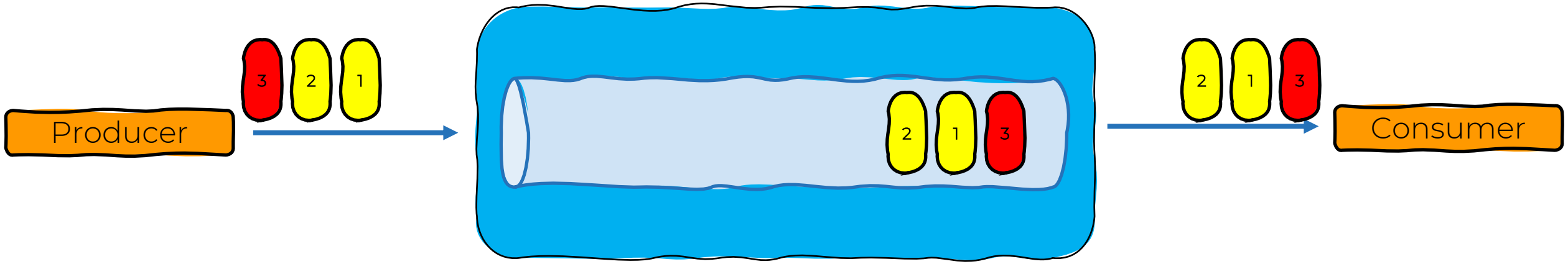
NACHRICHTENVERTEILUNG

MESSAGE QUEUE



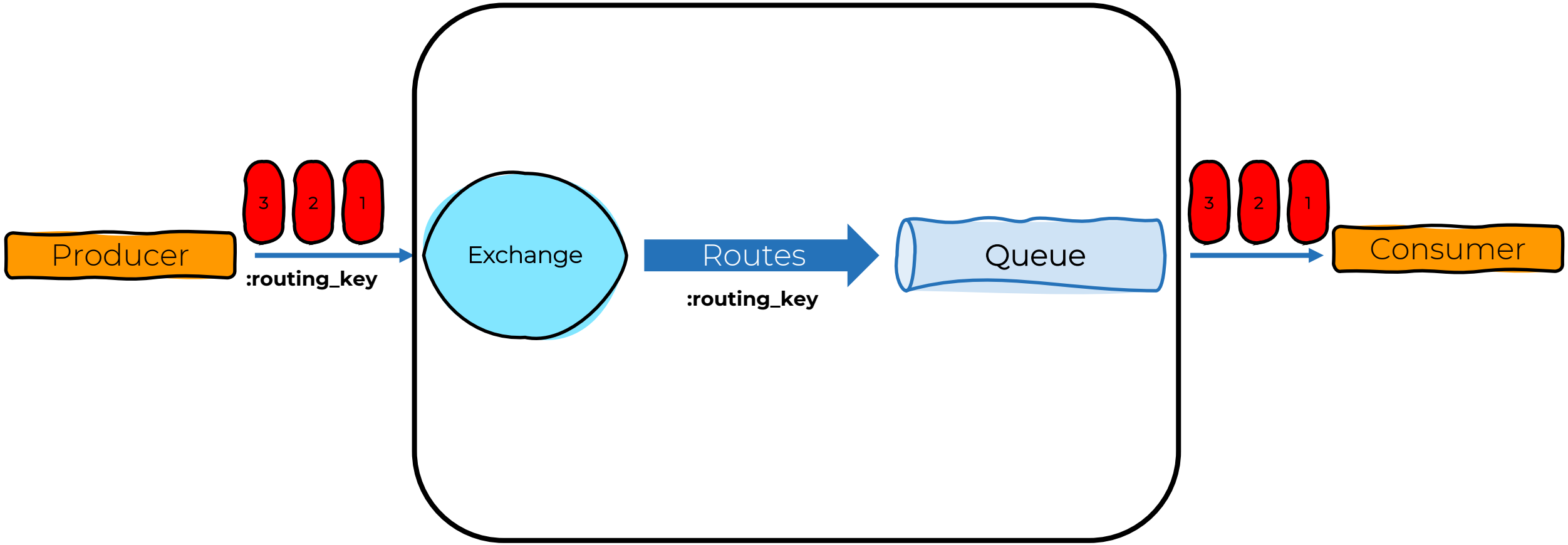
sidion

PRIQRITY QUEUE



sidion

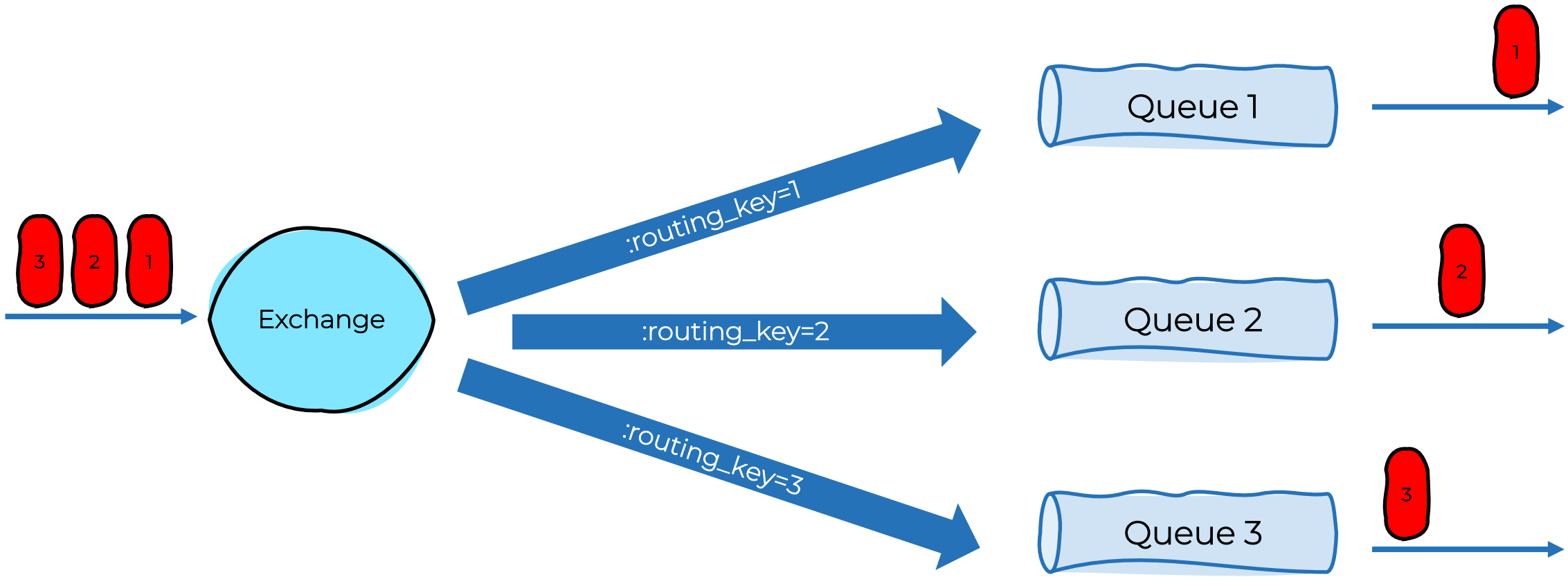
RABBITMQ EXCHANGE



sidion

NACHRICHTENVERTEILUNG

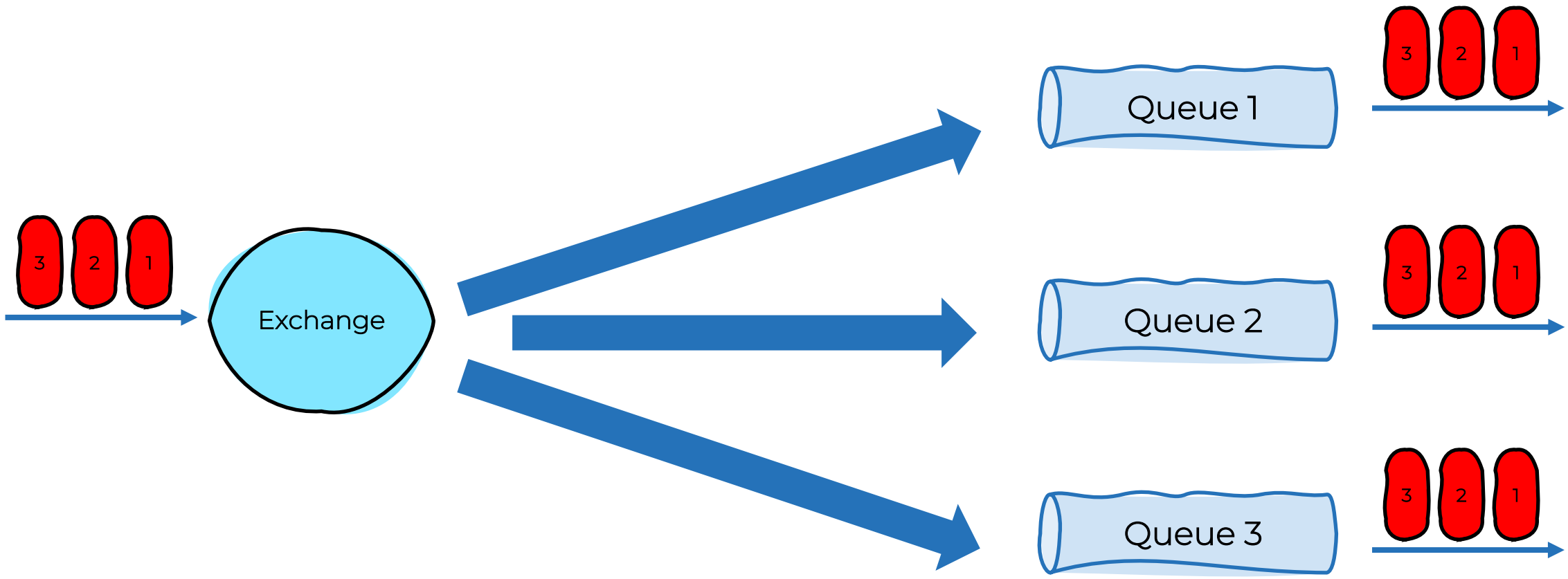
DIRECT EXCHANGE



sidion

NACHRICHTENVERTEILUNG

FANOUT EXCHANGE



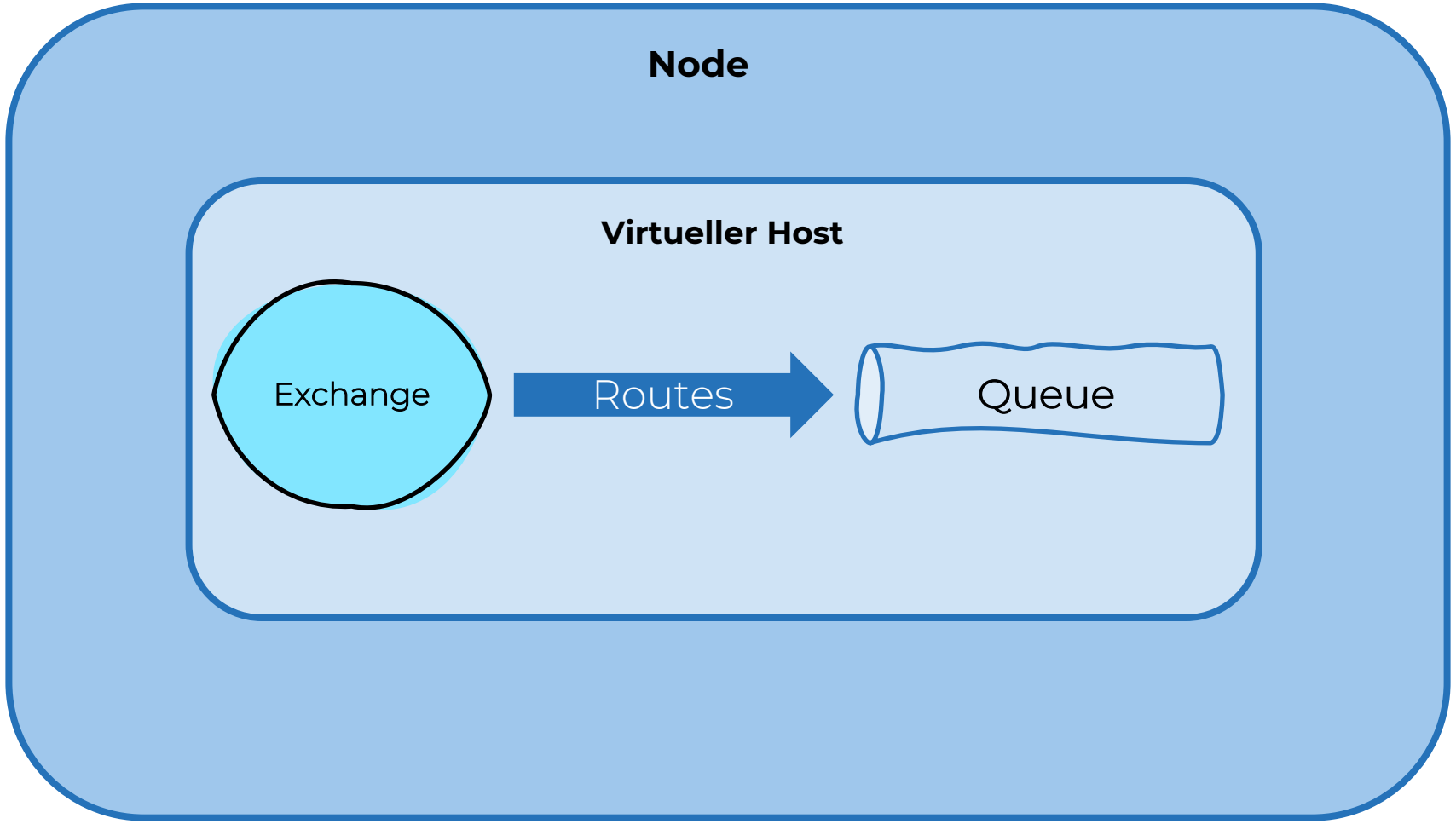
sidion

WEITERE EXCHANGES

- Topic Exchange
  - Mischform aus Direct Exchange und Fan Exchange
  - Anhand von Routing Patterns werden Messages ggf. mehrfach in Queues verteilt
- Headers Exchange
  - Ähnlich wie Topic Exchange
  - Anhand von Message Headern werden Messages ggf. mehrfach in Queues verteilt
- Custom Exchanges
  - Mittels selbstgeschriebener Plugins können eigene Routing Möglichkeiten implementiert werden
- Dead Letter Exchanges
  - Messages die nicht zugestellt werden können werden hier abgefangen und umgeleitet

sidion

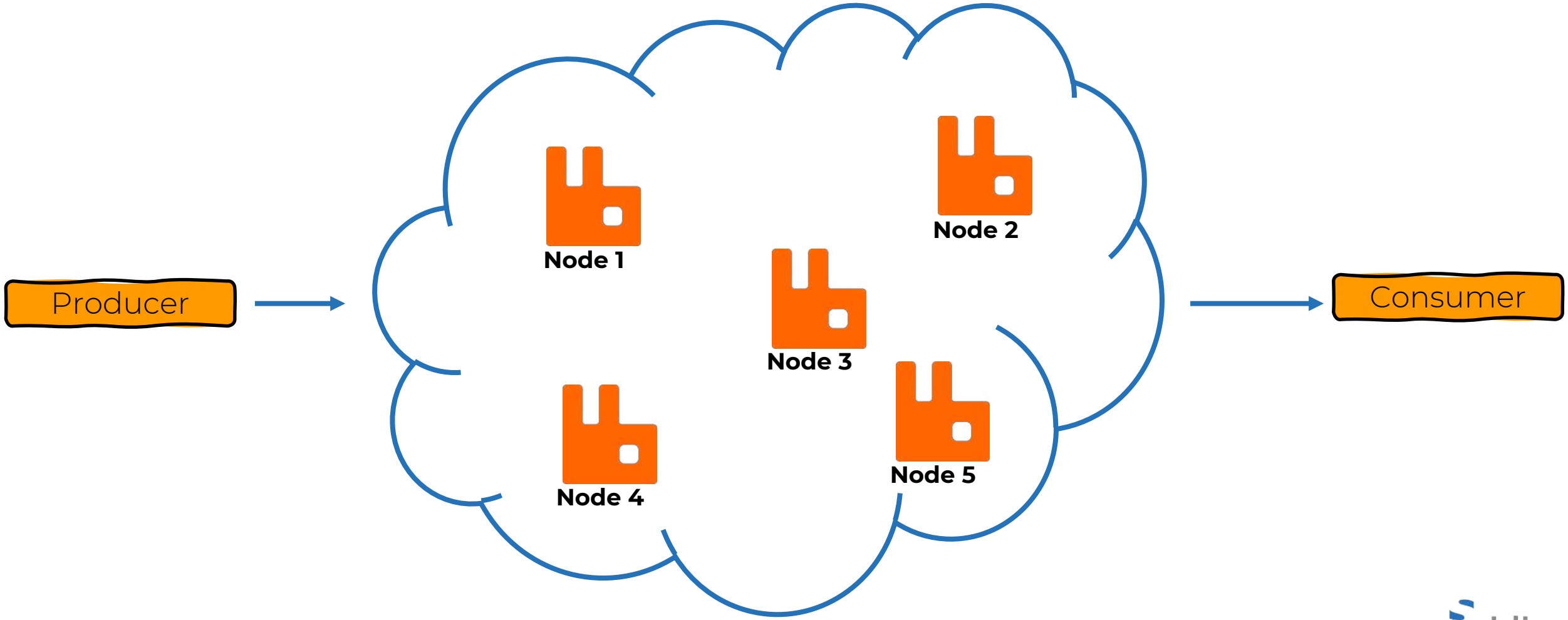
RABBITMQ NODES



sidion

CLUSTERING

RABBITMQ CLUSTER



sidion

KAFKA VS. RABBITMQ

	Apache Kafka	RabbitMQ
Einsatzbereich	Message-Bus, optimiert für hohen Datendurchsatz, Streaming und Replay	Klassischer Message Broker der das AMQP-Protokoll unterstützt. Zusätzlich sehr stark um langlaufende Jobs zu handlen
Persistence	Ja, mittels Log-File	Bedingt, bis zum Ack vom Consumer
Wiederabspielbar	Ja	Nein
Skalierbarkeit	Horizontale Skalierung wird empfohlen	Kombination aus Horizontaler und Vertikale Skalierung wird empfohlen

sidion

KAFKA VS. RABBITMQ

	Apache Kafka	RabbitMQ
Lizenz	Open Source: Apache License 2.0	Open Source: Mozilla Public License
Unterstützte Sprachen	Alle gängigen Programmiersprachen werden unterstützt	Alle gängigen Programmiersprachen werden unterstützt
Security	Kerberos, OAuth2, Standard Authentication	OAuth2, Standard Authentication
Monitoring	Mitgelieferte UI zur Überwachung vorhanden	Verfügbar über Third-Party Tools

sidion

REDIS

- Open Source In-Memory Data-Structure Store
- Aufgebaut als Key-Value-Store
- Persistenz durch regelmäßiges Abspeichern
- Sehr performant durch einfaches Clustering
- Enthält Datenstrukturen wie z.B. Strings, Hashes, Lists, Sets, Sorted Sets, etc
- Implementiert Pub/Sub als Message Broker

sidion

MESSAGE BROKER

HAZELCAST

- Open Source Distributed In-Memory Data-Grid
- Diverse Möglichkeiten der Datenverteilung unter Nutzung von z.B.
  - Maps
  - Sets
  - Lists
  - Queue
  - Topics
  - etc.
- Nutzung als Embedded-Version oder im Client-Server-Betrieb
- Deploybar mittlerweile in alle gängigen Cloud-Systeme

sidion

ZUSAMMENFASSUNG

- Apache Kafka
  - große Datenmengen die schnell verarbeitet werden müssen inkl. Realtime-Verarbeitung durch Streaming
  - Speicherung der Daten und somit die Wiederverwendung der Nachrichten
  - größerer Administrations-Aufwand
- RabbitMQ
  - Klassischer Message-Broker zur Vernetzung meiner Microservices
  - langlaufende Jobs innerhalb der Queues
  - Nachrichten mit Prioritäten
- RedisMQ
  - Bereits vorhandes Redis-Cluster als Pub-Sub-Mechnismus für meine Applikationen
- Hazelcast
  - Messaging ohne eigens deployten Service nutzbar

sidion

VIELEN DANK