目　　标

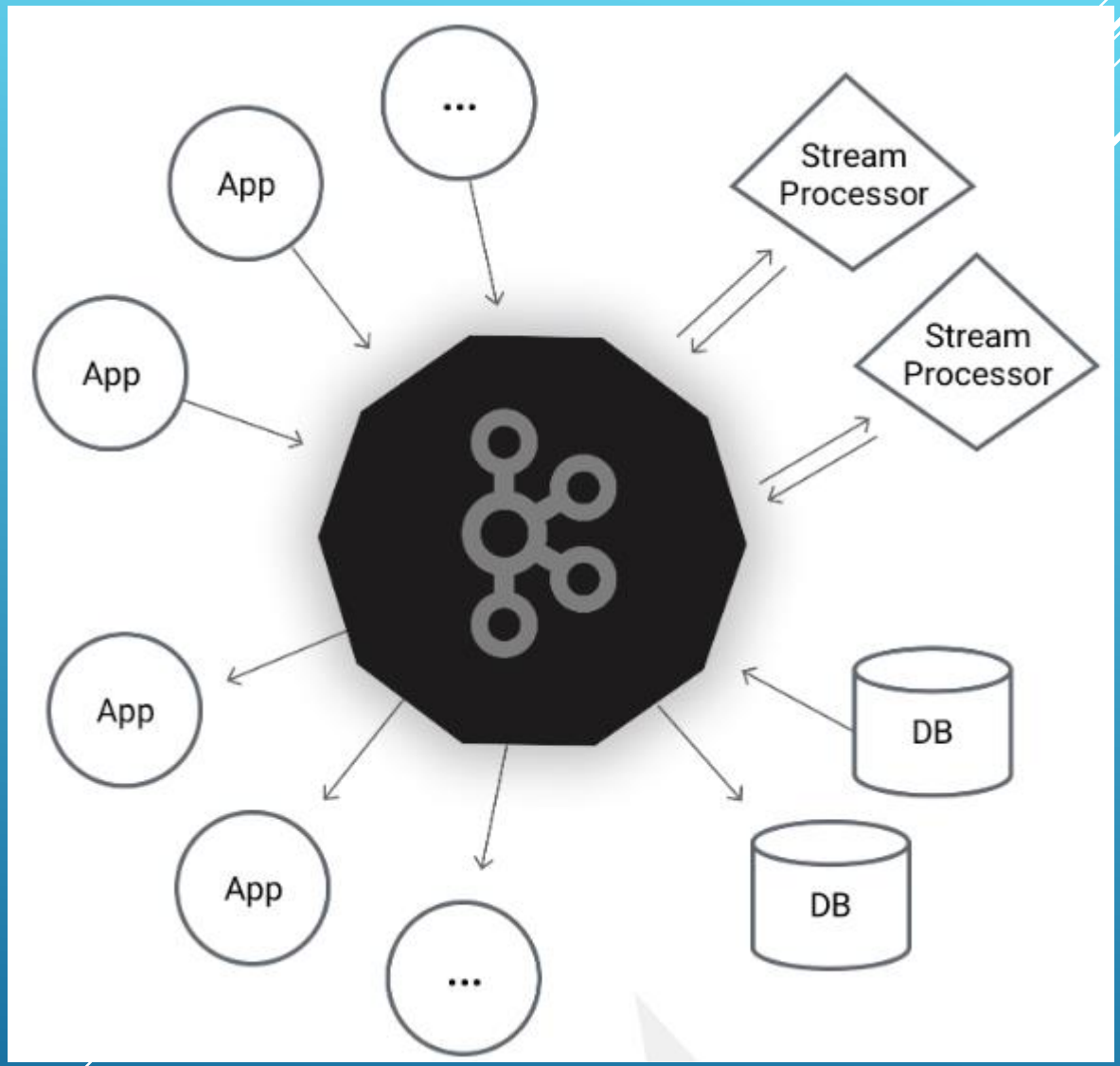
1. 了解Kafka的作用与基本原理
2. 掌握Kafka的经典使用场景及编程方式

一、什么是Kafka

错误：
摘自百度百科：Kafka是由Apache软件基金会开发的一个开源流处理平台，由Scala和Java编写。Kafka是一种高吞吐量的分布式发布订阅消息系统

正确

Linkedin ， 2010年开源
Apache 顶级项目， 2012年



二、它是用来干嘛的？

摘自官网

Kafka® is used for building real-time data pipelines and streaming apps. It is horizontally scalable, fault-tolerant, wicked fast, and runs in production in thousands of companies.

为实时应用、数据管道应用而生
可水平扩展
容错性好，性能高，很多公司在玩。。。。。

Apache Kafka® is *a distributed streaming platform*. What exactly does that mean?

A streaming platform has three key capabilities:

- Publish and subscribe to streams of records, similar to a message queue or enterprise messaging system.
- Store streams of records in a fault-tolerant durable way.
- Process streams of records as they occur.

Kafka is generally used for two broad classes of applications:

- Building real-time streaming data pipelines that reliably get data between systems or applications
- Building real-time streaming applications that transform or react to the streams of data

分布式流平台

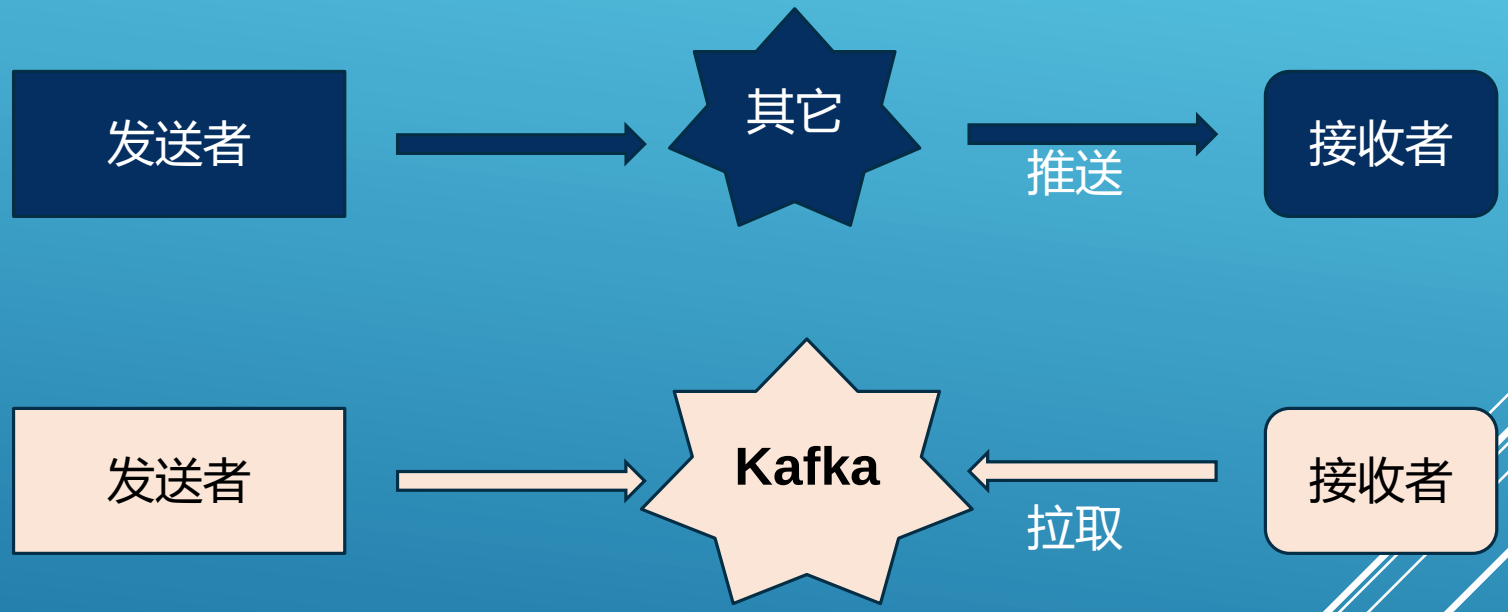
讲故事时间到

话说阿K公司上线的业务板块有很多个，涉及不同的领域，
其中不同的业务板块使用各自的IT系统，某天老大要求：

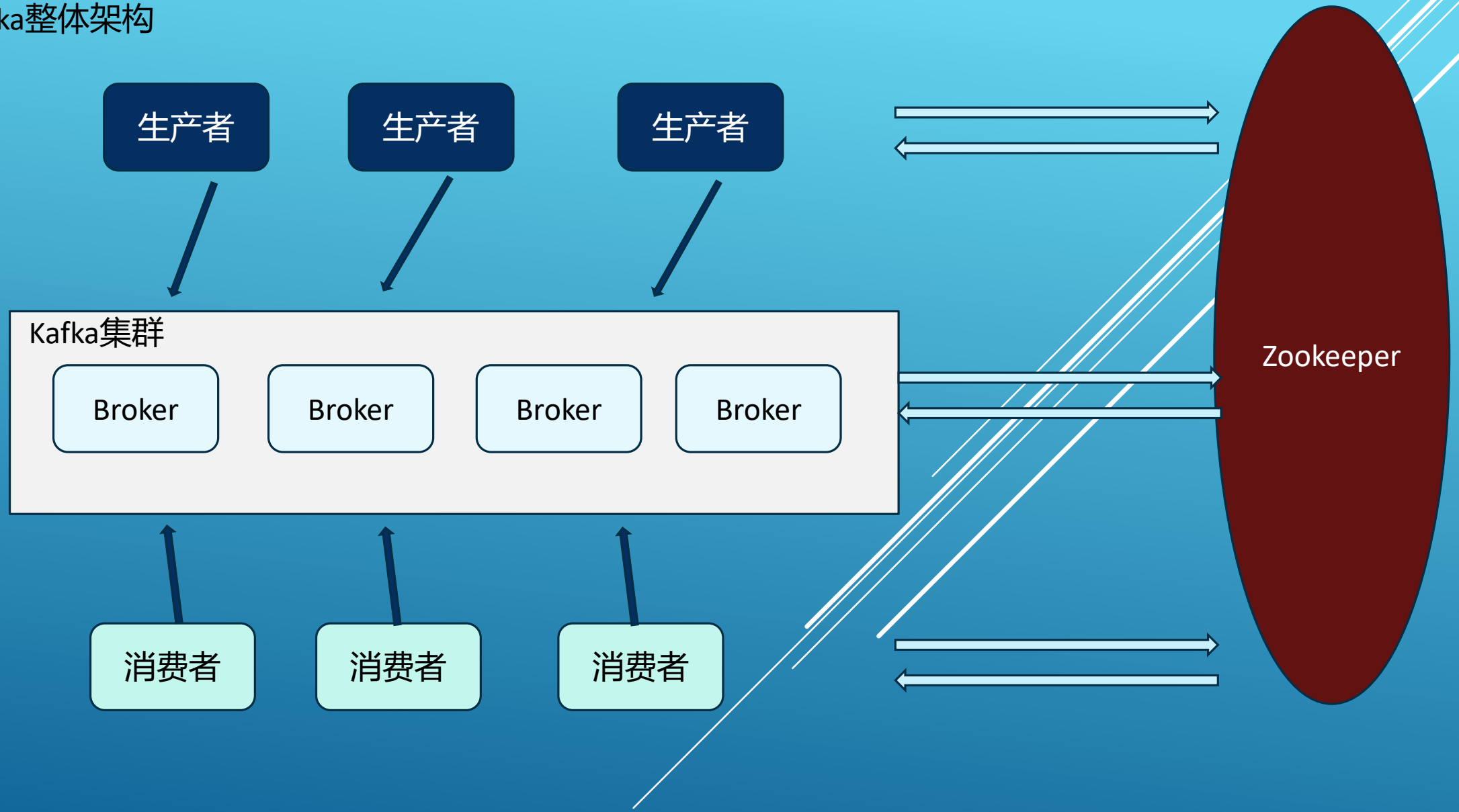
1、收集有用数据进行统一处理（分析、挖掘、监测）
2、各个业务系统得有交流

那么问题来了，针对1的场景 各业务系统产生数据的速度不同，规模不同，针对2 的场景 业务系统如何通信、同步

Kafka可以替代消息中间件，消息传送方式如下



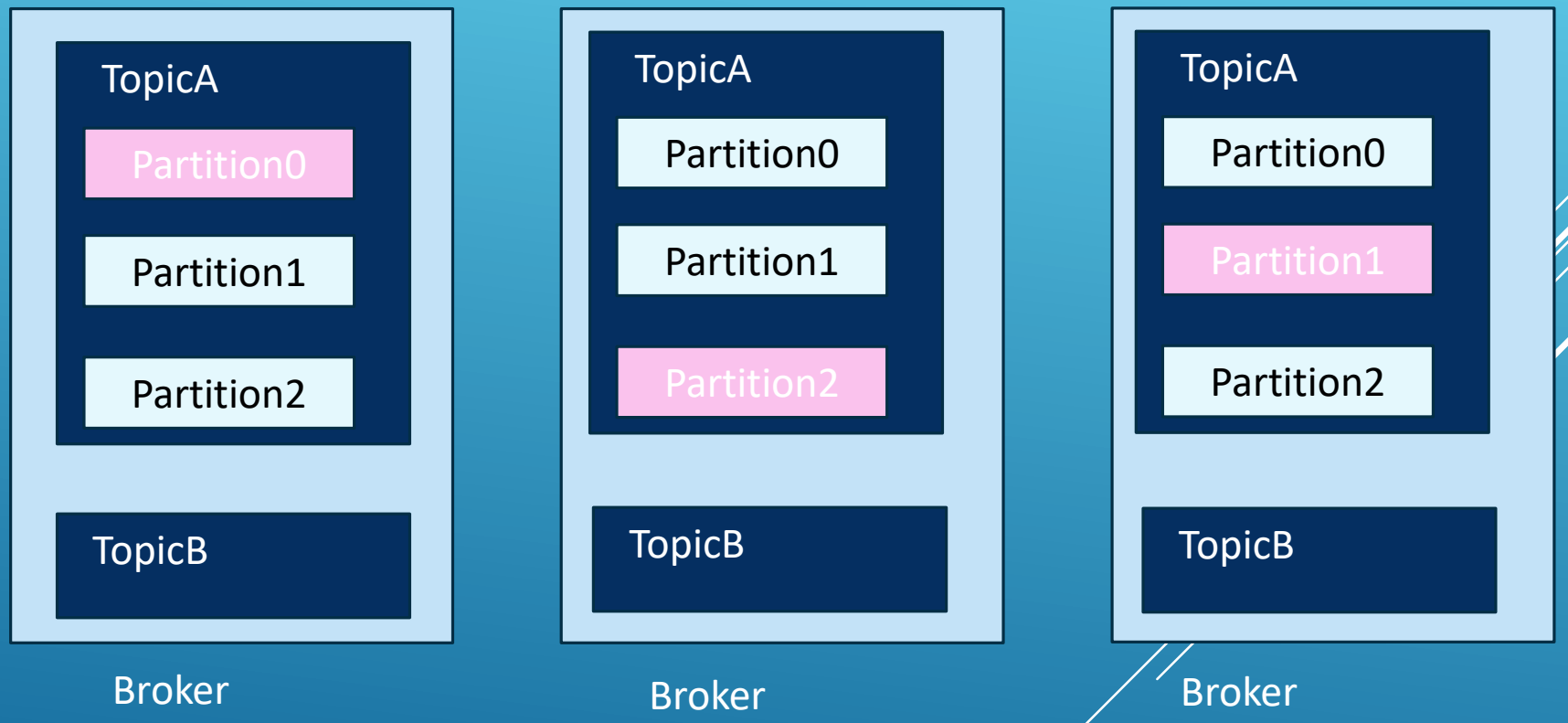
三、Kafka整体架构


生产者
生产者
生产者
Kafka集群
Broker
Broker
Broker
Broker
消费者
消费者
消费者
Zookeeper

SparkStreaming 的角色扮演是什么?

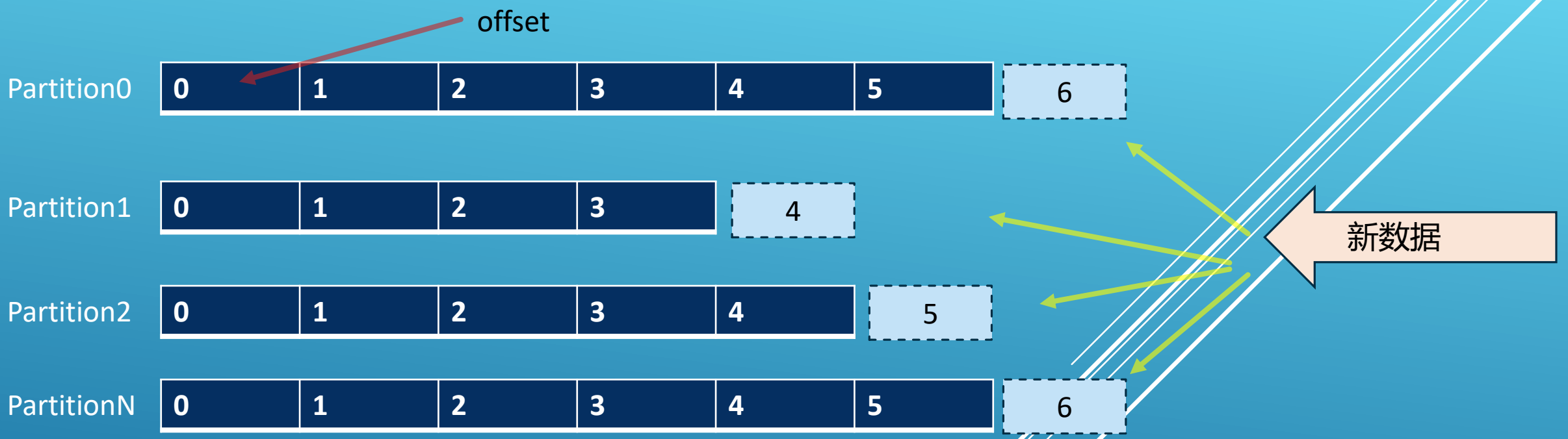
五、Kafka是如何储存消息的

消息按Topic分类， 每一个Topic可能划分为多个Patition，
并选一个Partiton为Leader



每一个Topic可以设定副本数，提高Topic可用性

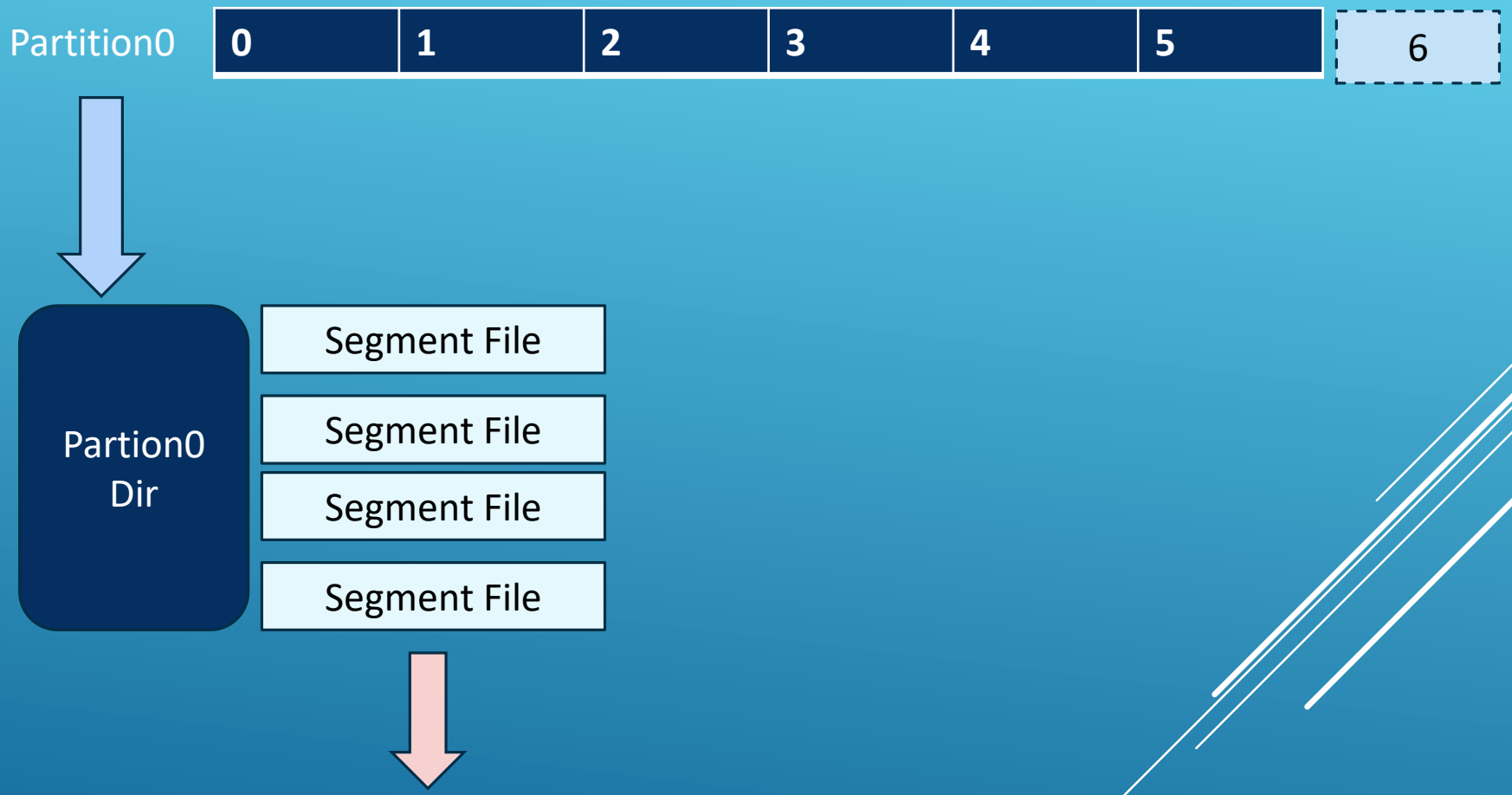
Partition如何储存数据



Partion内部数据有序，**但并非严格连续有序，**且以追加的形式写入

- 对比概念，最长公共子序列

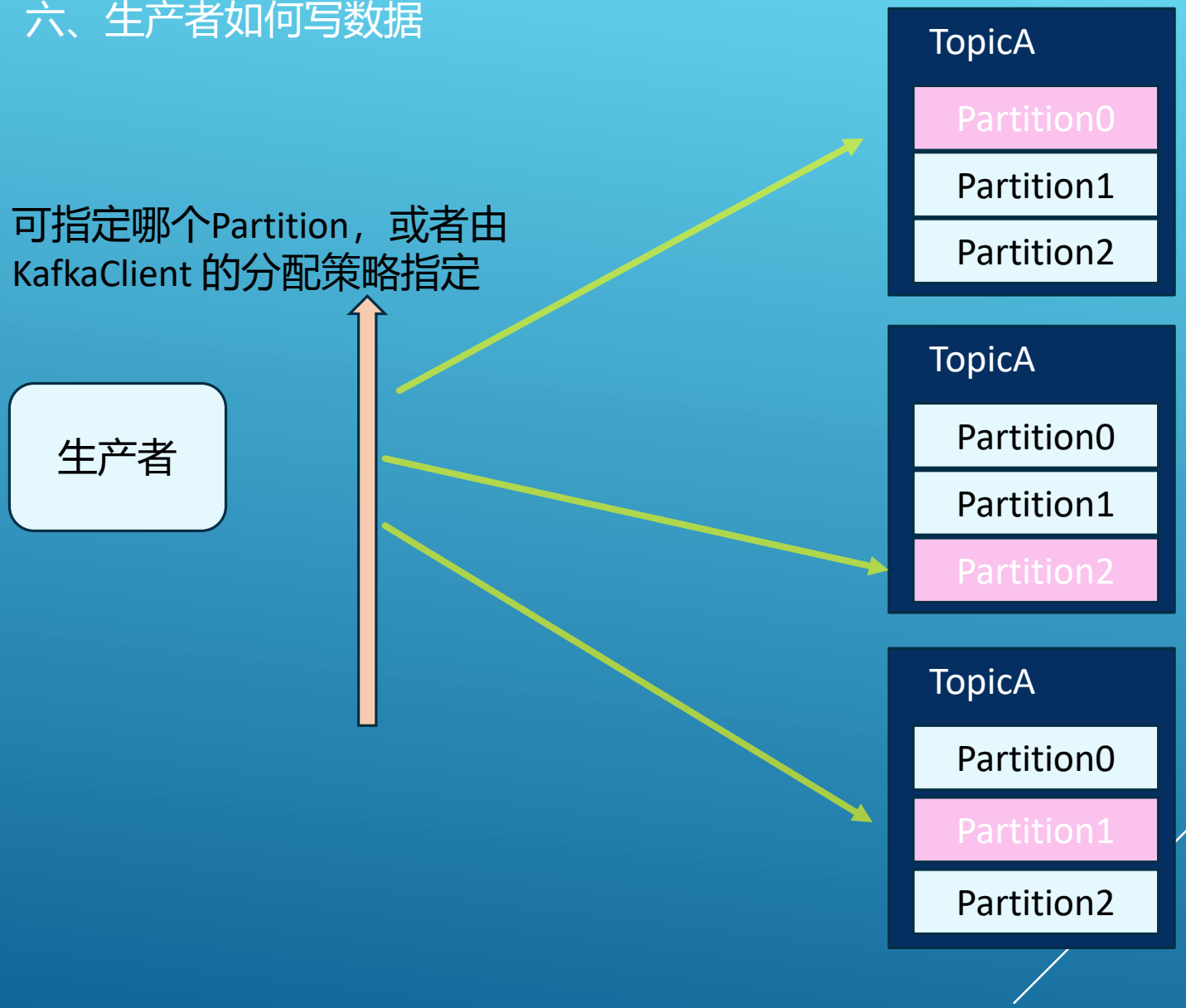
Partition如何储存数据



周期性保留， 设置固定大小， 写满了重新写新文件

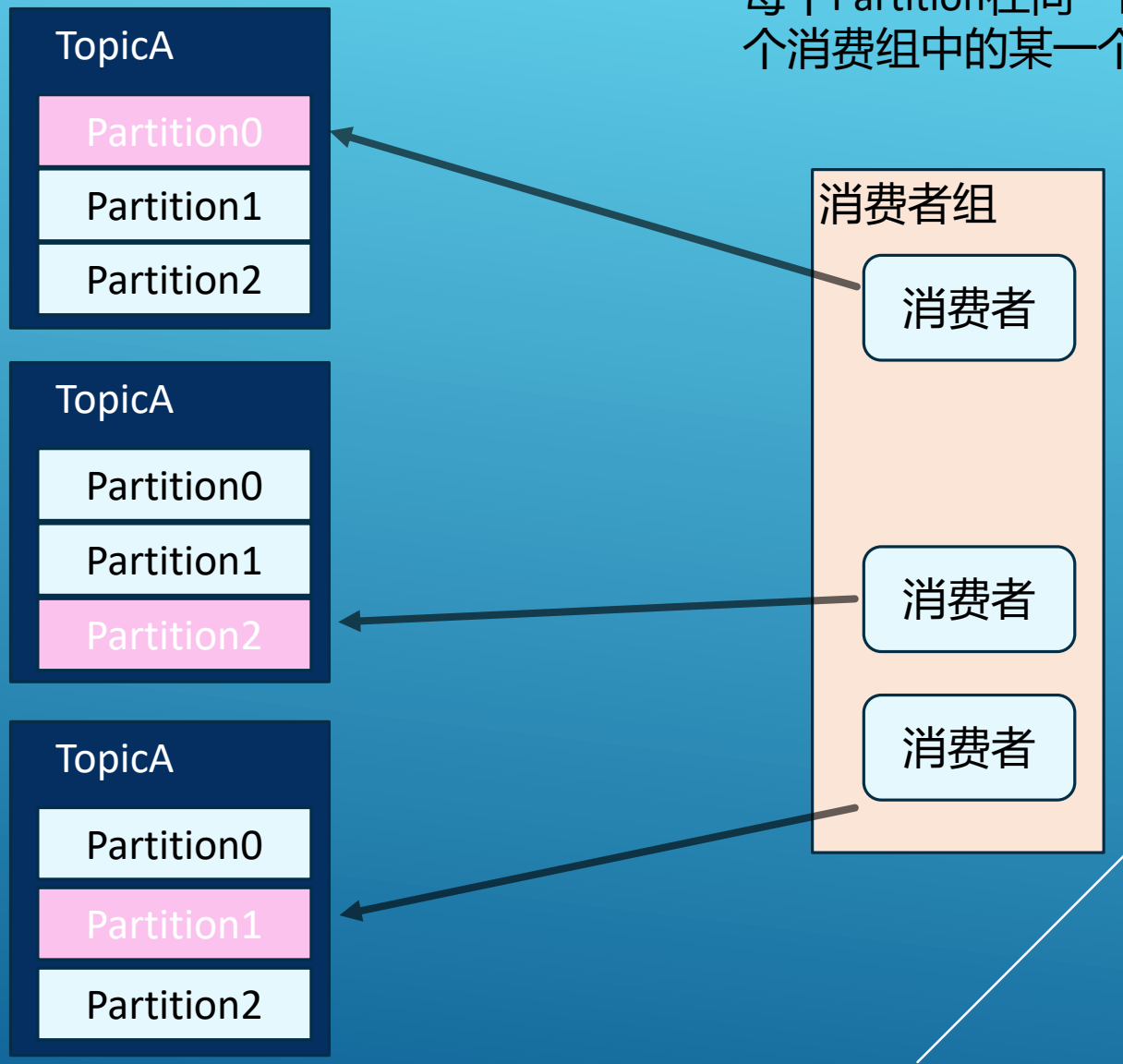
六、生产者如何写数据

可指定哪个Partition，或者由KafkaClient 的分配策略指定



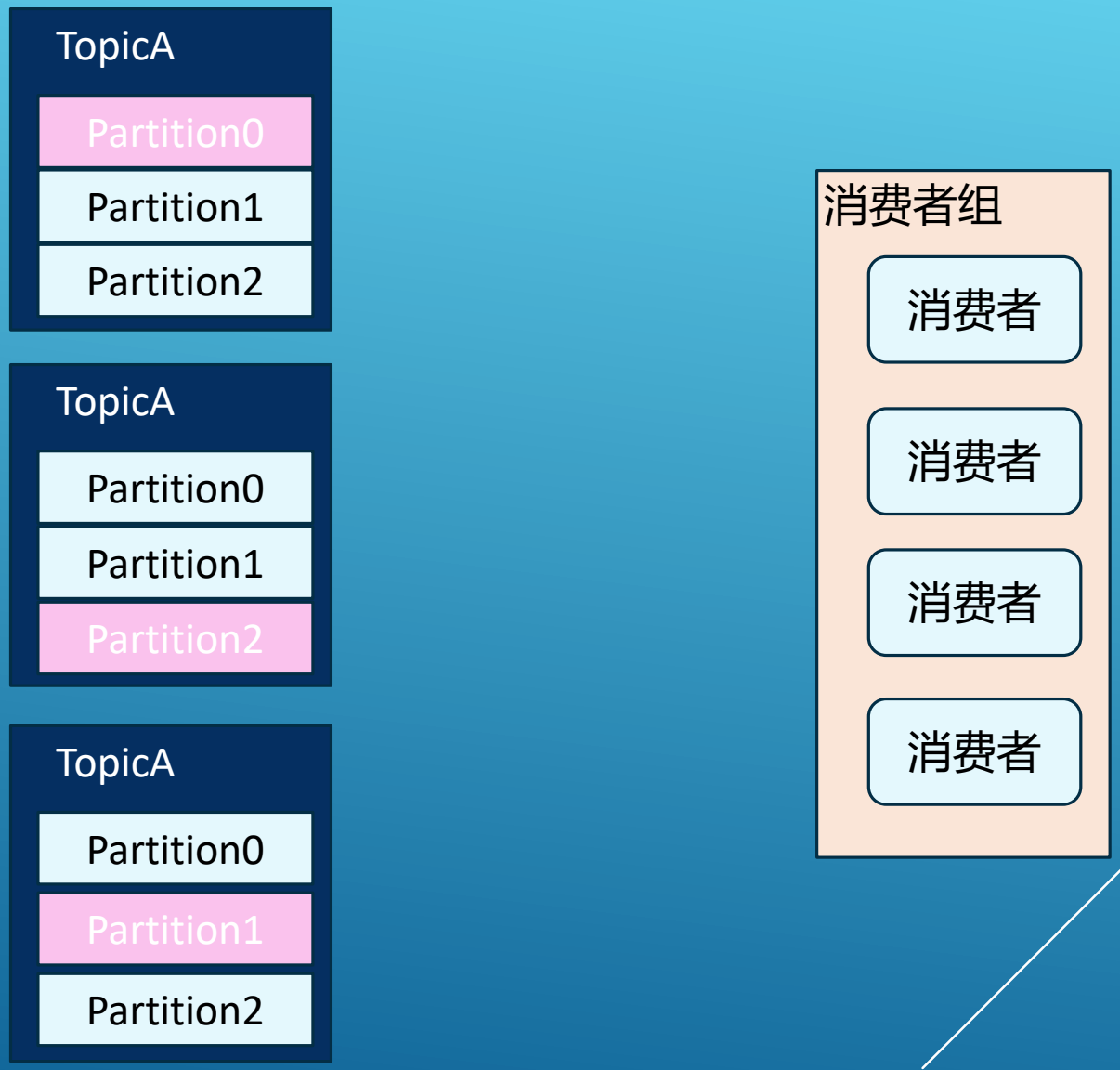
七、消费者如何写数据

每个Partition在同一时间只能被同一个消费组中的某一个消费者读取

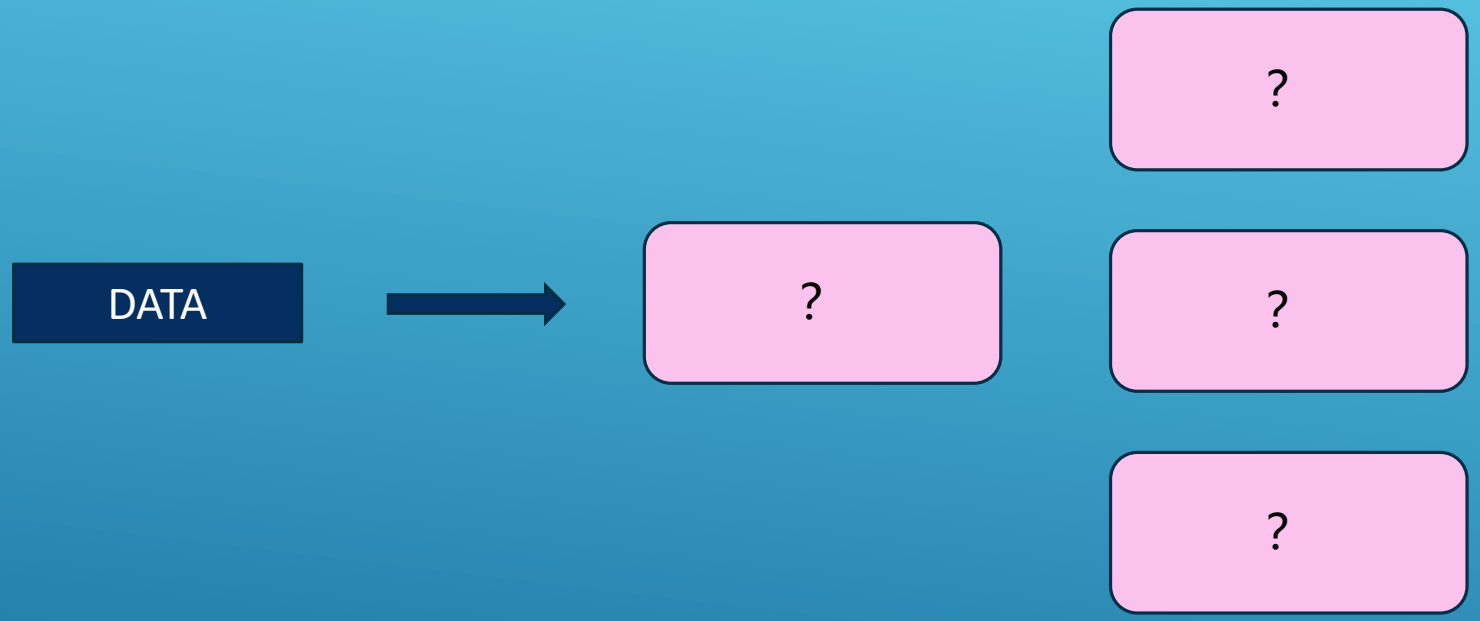


可指定哪个Partition，或者由KafkaClient 的分配策略指定

消费者数多于Partition个数，会出现什么情况?



一起总结下Kafka的数据储存是怎么组织起来的



八、Kafka基本使用

创建 Topic

```
[root@c1 zookeeper-3.4.14]# kafka-topics.sh --create --zookeeper c1:2181 --partitions 2 --replication-factor 2 --topic
 gamelog
```

查看已创建成功的 Topic

```
[root@c1 zookeeper-3.4.14]# kafka-topics.sh --describe --zookeeper c1:2181 --topic gamelog
Topic:gamelog   PartitionCount:2        ReplicationFactor:2     Configs:
        Topic: gamelog  Partition: 0    Leader: 0       Replicas: 0,1   Isr: 0,1
        Topic: gamelog  Partition: 1    Leader: 1       Replicas: 1,0   Isr: 1,0
```

每个字段是什么意思?

Kafka基本使用

查看已创建的Topics

```
[root@c1 zookeeper-3.4.14]# kafka-topics.sh --list --zookeeper c1
BookOrder
GameLog
```

往Topic写入数据

```
[root@c1 zookeeper-3.4.14]# kafka-console-producer.sh --broker-list c1:9092,c2:9092 --topic BookOrder
>Java 从入门到精通
>Scala编程实战
>Spark大数据开发实战
```

读取Topic 数据

```
[root@c1 zookeeper-3.4.14]# kafka-console-consumer.sh --bootstrap-server c1:9092 --from-beginning --topic BookOrder
Scala编程实战
Java 从入门到精通
Spark大数据开发实战
```