



APUSIC
固若长城
睿比世界

极简运维手册

金蝶Apusic分布式消息队列V2.0.6_for_kafka

版权声明

本文档所涉及的软件著作权、版权等知识产权已依法进行了注册，由金蝶天燕云计算股份有限公司合法拥有。受《中华人民共和国著作权法》《计算机软件保护条例》《知识产权保护条例》和相关国际版权条约、法律、法规以及其它知识产权法律和条约的保护。未经授权许可，不得非法使用。

免责声明

本文档包含的版权信息由金蝶天燕云计算股份有限公司合法拥有，受法律的保护，金蝶天燕云计算股份有限公司对本文档可能涉及到的非金蝶天燕云计算股份有限公司的信息不承担任何责任。在法律允许的范围内，您可以查阅并仅能够在《中华人民共和国著作权法》规定的合法范围内复制和打印本文档。任何单位和个人未经金蝶天燕云计算股份有限公司书面授权许可，不得使用、修改、再发布本文档的任何部分和内容，否则将被视为侵权，金蝶天燕云计算股份有限公司有依法追究其责任的权利。

本文档如有更新，不另行通知。对本文档中的问题您可向金蝶天燕云计算股份有限公司告知或查询。未经本公司明确授予的任何权利均予保留。

商标声明

 是深圳市金蝶天燕云计算股份有限公司向中华人民共和国国家商标局申请注册的注册商标，注册商标专用权由金蝶天燕合法拥有，受法律保护。未经金蝶天燕的书面许可，任何单位及个人不得以任何方式或理由对该商标的任何部分进行使用、复制、修改、传播、抄录或与其它产品捆绑使用销售。凡侵犯金蝶天燕商标权的，金蝶天燕将依法追究其法律责任。本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

目录

- 1 概述
- 2 产品简介
 - 2.1 什么是 ADMQ for Kafka
 - 2.2 核心特性
 - 2.3 适用场景
 - 2.4 产品组成
- 3 部署架构
 - 3.1 单机部署
 - 3.2 集群部署（推荐生产环境）
- 4 快速安装
 - 4.1 系统要求
 - 4.2 端口规划
 - 4.3 单机安装（5分钟完成）
 - 4.4 集群安装（3节点）
- 5 常用命令速查
 - 5.1 进程管理
 - 5.2 Topic 管理
 - 5.3 消息测试
 - 5.4 消费者组管理
 - 5.5 集群管理
- 6 日常运维
 - 6.1 巡检清单
 - 6.2 核心配置文件
 - 6.3 故障处理
- 7 性能调优
 - 7.1 系统级优化
 - 7.2 Broker 高吞吐配置
 - 7.3 生产者优化
 - 7.4 消费者优化
 - 7.5 JVM 调优
- 8 监控指标
 - 8.1 关键告警指标
 - 8.2 监控工具
- 9 目录结构
- 10 技术支持

1 概述

本文档为 ADMQ for Kafka 极简运维手册，涵盖部署架构、快速安装、常用命令、日常运维及性能调优核心内容。

2 产品简介

2.1 什么是 ADMQ for Kafka

ADMQ for Kafka（Apusic Distributed Message Queue for Kafka）是专为 Kafka 用户打造的信创替代型云原生消息中间件，基于 Apache Kafka 3.9.1 内核构建，完全遵循 Kafka 核心架构设计与通信协议规范，实现协议、客户端、生态工具的全兼容。

2.2 核心特性

特性	说明
协议完全兼容	完整实现 Kafka 生产消费协议、分区分配协议、Offset 管理协议，行为与原生 Kafka 一致
零改造替代	支持 Java、Go、Python、C++ 等 Kafka 官方客户端，无需修改代码即可替换
信创适配	适配麒麟、统信等信创操作系统，兼容 ARM、RISC-V 等信创芯片
高可用架构	支持多副本冗余、故障自动转移、水平扩展，满足生产环境可靠性要求
生态复用	兼容 Kafka Connect、MirrorMaker、Prometheus 等原生生态工具

2.3 适用场景

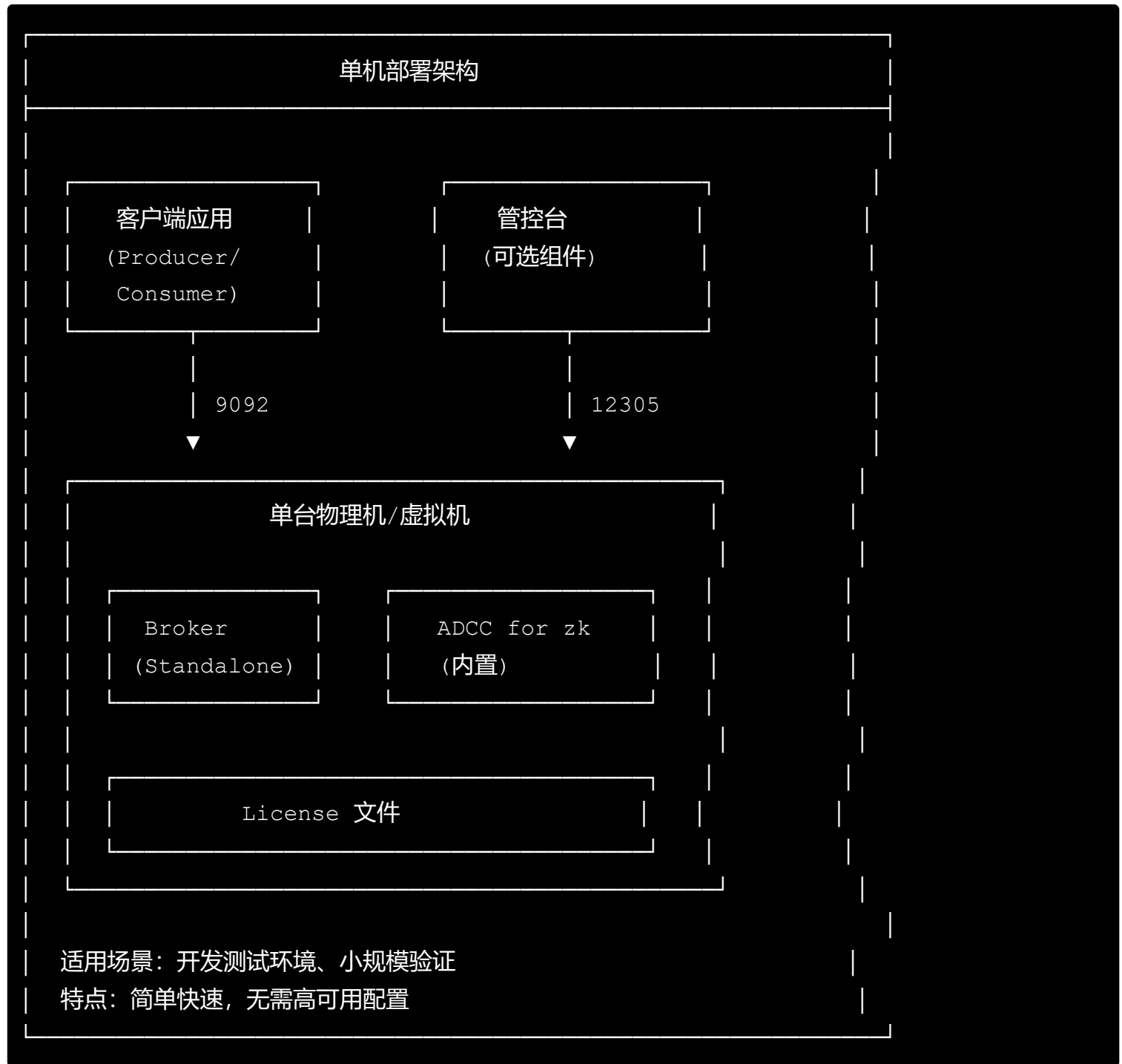
- **信创替代**：政企信创改造中直接替换原有 Kafka 集群
- **高吞吐数据传输**：日志采集、业务数据同步、实时数据管道
- **流式数据处理**：兼容 Flink、Spark Streaming 等流计算引擎
- **系统解耦**：基于发布订阅模式实现分布式系统间异步通信

2.4 产品组成

组件	说明	必需
Broker（核心引擎）	消息处理与存储核心，受 License 控制	是
ADCC for zk（分布式协调）	集群元数据管理与协调	是
管控台	可视化运维管理界面	否（可选）

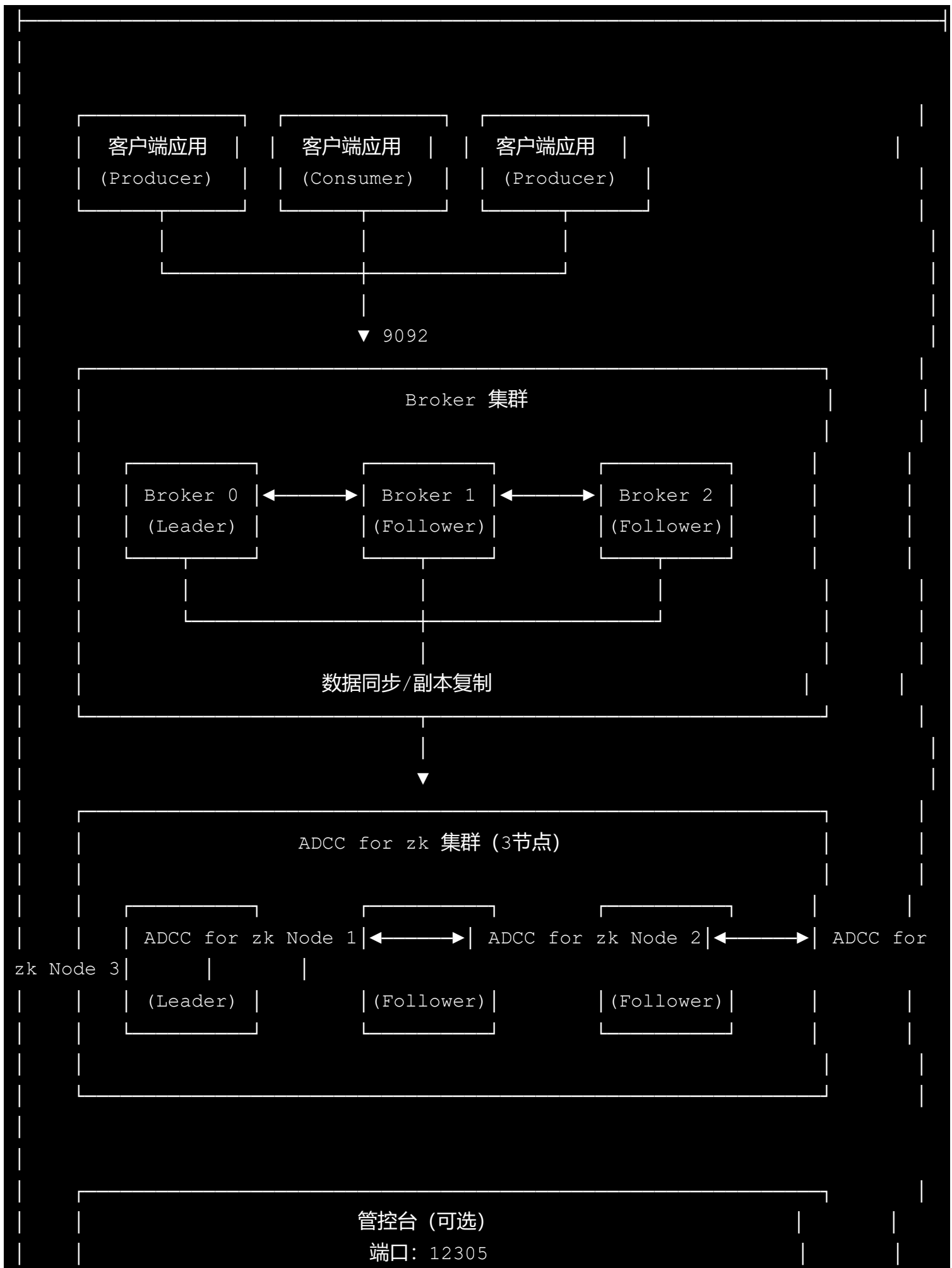
3 部署架构

3.1 单机部署



3.2 集群部署 (推荐生产环境)





适用场景：生产环境、高可用业务场景

特点：多副本冗余、故障自动转移、水平扩展

4 快速安装

4.1 系统要求

组件	最低要求	推荐配置
操作系统	Linux (CentOS 7+, Ubuntu 18.04+, 麒麟 V10, 统信 UOS)	-
CPU	4 核	8 核及以上
内存	8 GB	16 GB 及以上
磁盘空间	100 GB	500 GB 及以上 SSD

4.2 端口规划

端口	服务	说明
9092	Broker	生产者和消费者连接
2181	ADCC for zk	客户端连接
2888	ADCC for zk	Leader 和 Follower 通信
3888	ADCC for zk	选举通信
12305	管控台	Web 界面 (HTTP)

4.3 单机安装（5分钟完成）

```
# 1. 解压
mkdir -p /apusic && cd /apusic
tar zxvf ADMQ-V2.0.6.391-Kafka-20260319.tar.gz
cd admq-kafka

# 2. 配置 (修改为本机IP)
vi config/kafka-standalone.conf
# listeners=PLAINTEXT://192.168.1.10:9092

# 3. 放置 License
cp /path/to/license.xml ./

# 4. 启动 (先zk, 后Broker)
bin/admq-daemon start kafka zk
```

```
sleep 3
bin/admq-daemon start kafka standalone

# 5. 验证
netstat -nltup | grep 9092
kafka/bin/kafka-topics.sh --list --bootstrap-server 192.168.1.10:9092
```

4.4 集群安装 (3节点)

节点规划:

节点	IP	ADCC for zk MyID	Broker ID
Node 1	192.168.1.10	1	0
Node 2	192.168.1.11	2	1
Node 3	192.168.1.12	3	2

各节点执行:

```
# 1. 解压并配置
mkdir -p /apusic && cd /apusic
tar zxvf ADMQ-V2.0.6.391-Kafka-20260319.tar.gz
cd admq-kafka

# 2. 配置 ADCC for zk (所有节点相同)
cat > config/kafka-zk.conf << 'EOF'
server.1=192.168.1.10:2888:3888:participant;2181
server.2=192.168.1.11:2888:3888:participant;2181
server.3=192.168.1.12:2888:3888:participant;2181
EOF

# 3. 创建 myid (各节点不同)
mkdir -p data/zk/data
echo "1" > data/zk/data/myid # Node 1 写 1, Node 2 写 2, Node 3 写 3

# 4. 配置 Broker (各节点修改 broker.id 和 listeners)
cat > config/kafka-broker.conf << 'EOF'
broker.id=0
listeners=PLAINTEXT://192.168.1.10:9092
```

```
zookeeper.connect=192.168.1.10:2181,192.168.1.11:2181,192.168.1.12:2181
EOF
```

启动集群：

```
# 所有节点启动 ADCC for zk
bin/admq-daemon start kafka zk

# 所有节点启动 Broker
bin/admq-daemon start kafka broker

# 验证集群
kafka/bin/kafka-broker-api-versions.sh --bootstrap-server 192.168.1.10:9092
```

5 常用命令速查

5.1 进程管理

```
# 启动/停止 ADCC for zk
bin/admq-daemon start kafka zk
bin/admq-daemon stop kafka zk

# 启动/停止 Broker (单机)
bin/admq-daemon start kafka standalone
bin/admq-daemon stop kafka standalone

# 启动/停止 Broker (集群)
bin/admq-daemon start kafka broker
bin/admq-daemon stop kafka broker
```

5.2 Topic 管理

```
# 创建 Topic
kafka/bin/kafka-topics.sh --create \
  --bootstrap-server localhost:9092 \
  --topic my-topic --partitions 6 --replication-factor 3

# 查看 Topic 列表
kafka/bin/kafka-topics.sh --list --bootstrap-server localhost:9092

# 查看 Topic 详情 (含 ISR)
kafka/bin/kafka-topics.sh --describe \
  --bootstrap-server localhost:9092 --topic my-topic

# 删除 Topic
kafka/bin/kafka-topics.sh --delete \
  --bootstrap-server localhost:9092 --topic my-topic

# 增加分区 (只能增加)
kafka/bin/kafka-topics.sh --alter \
```

```
--bootstrap-server localhost:9092 \  
--topic my-topic --partitions 12
```

5.3 消息测试

生产消息

```
kafka/bin/kafka-console-producer.sh \  
--bootstrap-server localhost:9092 --topic my-topic
```

消费消息 (从头开始)

```
kafka/bin/kafka-console-consumer.sh \  
--bootstrap-server localhost:9092 \  
--topic my-topic --from-beginning
```

消费消息 (最新)

```
kafka/bin/kafka-console-consumer.sh \  
--bootstrap-server localhost:9092 --topic my-topic
```

5.4 消费者组管理

查看消费者组列表

```
kafka/bin/kafka-consumer-groups.sh \  
--bootstrap-server localhost:9092 --list
```

查看消费进度 (关键: 关注 LAG)

```
kafka/bin/kafka-consumer-groups.sh \  
--bootstrap-server localhost:9092 \  
--group my-group --describe
```

重置 Offset 到最早

```
kafka/bin/kafka-consumer-groups.sh \  
--bootstrap-server localhost:9092 \  
--group my-group --topic my-topic \  
--reset-offsets --to-earliest --execute
```

重置 Offset 到最新

```
kafka/bin/kafka-consumer-groups.sh \  
--bootstrap-server localhost:9092 --topic my-topic
```

```
--bootstrap-server localhost:9092 \  
--group my-group --topic my-topic \  
--reset-offsets --to-latest --execute
```

5.5 集群管理

```
# 查看 Broker 信息  
kafka/bin/kafka-broker-api-versions.sh --bootstrap-server localhost:9092  
  
# 首选副本选举 (Leader 不均衡时)  
kafka/bin/kafka-leader-election.sh \  
  --bootstrap-server localhost:9092 \  
  --election-type preferred --all-topic-partitions  
  
# 副本重分配 (扩容后使用)  
kafka/bin/kafka-reassign-partitions.sh \  
  --bootstrap-server localhost:9092 --generate
```

6 日常运维

6.1 巡检清单

检查项	命令	正常指标
Broker 进程	<code>ps -ef grep admq-kafka</code>	进程存在
端口监听	<code>netstat -nltup grep 9092</code>	LISTEN 状态
磁盘使用率	<code>df -h</code>	< 70%
IO 使用率	<code>iostat -x</code>	< 80%
ISR 状态	<code>kafka-topics.sh --describe</code>	ISR = Replicas
消费积压	<code>kafka-consumer-groups.sh --describe</code>	LAG 不持续增长

6.2 核心配置文件

Broker 关键配置 (`config/kafka-broker.conf`):

```
# 身份与网络
broker.id=0
listeners=PLAINTEXT://192.168.1.10:9092
advertised.listeners=PLAINTEXT://192.168.1.10:9092

# 存储
log.dirs=/data/kafka
num.partitions=6

# 可靠性 (生产环境必配)
default.replication.factor=3
min.insync.replicas=2
unclean.leader.election.enable=false

# 性能
num.network.threads=16
num.io.threads=32
compression.type=zstd

# 日志保留
log.retention.hours=168
```

```
log.segment.bytes=1073741824

# ADCC for zk
zookeeper.connect=192.168.1.10:2181,192.168.1.11:2181,192.168.1.12:2181
```

ADCC for zk 关键配置 (config/kafka-zk.conf):

```
tickTime=2000
initLimit=10
syncLimit=5

dataDir=/apusic/admq-kafka/data/zk/data
dataLogDir=/apusic/admq-kafka/data/zk/logs

# 集群节点
server.1=192.168.1.10:2888:3888:participant;2181
server.2=192.168.1.11:2888:3888:participant;2181
server.3=192.168.1.12:2888:3888:participant;2181

# 自动清理
autopurge.snapRetainCount=3
autopurge.purgeInterval=6
```

6.3 故障处理

故障现象	排查命令	解决方案
Broker 宕机	<code>ps -ef grep admq-kafka</code>	<code>bin/admq-daemon start kafka broker</code>
Leader 不可用	<code>kafka-topics.sh --describe</code>	执行首选副本选举
ISR 不同步	<code>kafka-topics.sh --describe</code>	检查磁盘 IO、网络延迟
消费积压(LAG)	<code>kafka-consumer-groups.sh --describe</code>	增加 Consumer/Partition
磁盘满	<code>df -h</code>	扩容磁盘或缩短日志保留时间
ADCC for zk 连接失败	检查 myid、端口、防火墙	修复配置后重启

7 性能调优

7.1 系统级优化

```
# 文件描述符
ulimit -n 1000000

# 网络参数
sysctl -w net.core.rmem_max=67108864
sysctl -w net.core.wmem_max=67108864
sysctl -w net.ipv4.tcp_rmem="4096 87380 67108864"
sysctl -w net.ipv4.tcp_wmem="4096 16384 67108864"

# 禁用 swap
sysctl -w vm.swappiness=0

# 文件系统 (挂载时)
mount -o noatime,nodiratime /dev/sda1 /data/kafka
```

7.2 Broker 高吞吐配置

```
# 网络与线程
num.network.threads=16
num.io.threads=32
socket.send.buffer.bytes=307200
socket.receive.buffer.bytes=307200
queued.max.requests=2000

# 日志
log.segment.bytes=524288000
log.retention.check.interval.ms=300000

# 压缩与副本
compression.type=zstd
replica.socket.timeout.ms=30000
replica.fetch.max.bytes=1048576
```

7.3 生产者优化

```
Properties props = new Properties();
props.put("bootstrap.servers", "192.168.1.10:9092");
props.put("acks", "all");
props.put("compression.type", "zstd");
props.put("batch.size", 32768);
props.put("linger.ms", 10);
props.put("buffer.memory", 67108864);
props.put("max.in.flight.requests.per.connection", 5);
```

7.4 消费者优化

```
Properties props = new Properties();
props.put("bootstrap.servers", "192.168.1.10:9092");
props.put("group.id", "my-group");
props.put("max.poll.records", 1000);
props.put("fetch.min.bytes", 1);
props.put("fetch.max.wait.ms", 500);
```

7.5 JVM 调优

```
# 大堆内存 + G1 GC
KAFKA_HEAP_OPTS="-Xms16g -Xmx16g -XX:+UseG1GC \
  -XX:MaxGCPauseMillis=20 \
  -XX:InitiatingHeapOccupancyPercent=35 \
  -XX:G1HeapRegionSize=16m"

# JMX 监控
JMX_PORT=9999
```

8 监控指标

8.1 关键告警指标

指标	告警阈值	说明
UnderReplicatedPartitions	> 0	未充分同步的分区
OfflinePartitionsCount	> 0	离线分区数
RequestLatencyAvg	> 100ms	平均请求延迟
MessagesInPerSec	根据业务	每秒消息数
LAG	持续增长	消费积压

8.2 监控工具

```
# JMX 连接
jconsole localhost:9999

# Prometheus 配置
scrape_configs:
  - job_name: 'kafka'
    static_configs:
      - targets: ['localhost:12305']
```

9 目录结构

```
admq-kafka/
├─ bin/                # 启动、停止脚本
├─ config/             # 配置文件
│   ├── kafka-standalone.conf    # 单机配置
│   ├── kafka-broker.conf        # Broker 配置
│   └─ kafka-zk.conf            # ADCC for zk 配置
├─ data/               # 数据目录
│   ├── kafka/          # 消息数据
│   └─ zk/              # ADCC for zk 数据
├─ jdk/                # 内置 JDK
├─ logs/               # 运行日志
└─ service/            # systemd 服务文件
```

10 技术支持

- 电话：400-855-5800
- 邮箱：support@apusic.com
- 网址：www.apusic.com

全国统一服务热线
4008-555-800



金蝶天燕云计算股份有限公司(简称“金蝶天燕云”)成立于2000年,前身为“金蝶中间件公司”,是金蝶集团旗下新一代软件基础云平台服务商,云计算国家标准制定企业,国家信创产业核心软件企业。金蝶天燕是国家863重点研发计划与核高基重大专项承接企业,也是“两网一站四库十二金”国家重点工程的基础平台提供商,产品广泛应用于政府、军工、金融、能源等关键行业,累计服务客户总数超过10万家。



云计算国家标准制定企业
金蝶集团旗下基础软件企业
信息技术应用创新核心企业
官网: www.apusic.com

