



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_rabbitmq

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。

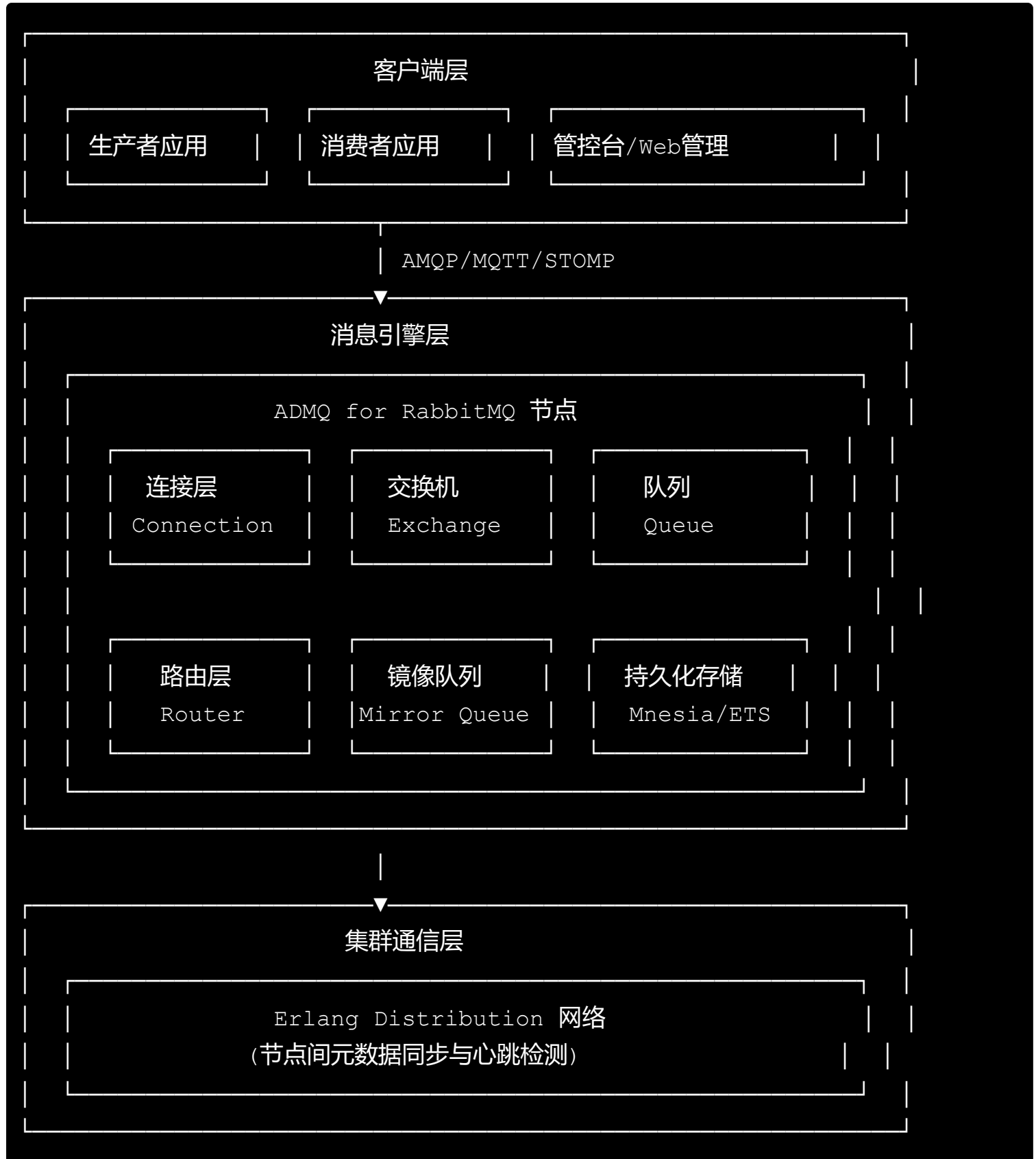
目录

- .1 一、部署架构
 - .1.1 整体架构
 - .1.2 部署模式
- .2 二、快速安装
 - .2.1 1. 环境准备
 - .2.2 2. 单机安装
 - .2.3 3. 集群安装 (3 节点)
- .3 三、核心概念
 - .3.1 消息模型
 - .3.2 交换机类型
 - .3.3 队列类型
- .4 四、常用命令速查
 - .4.1 服务管理
 - .4.2 状态查看
 - .4.3 用户管理
 - .4.4 队列管理
 - .4.5 集群管理
 - .4.6 策略管理
 - .4.7 备份恢复
- .5 五、日常运维
 - .5.1 日志位置
 - .5.2 日常检查清单
 - .5.3 健康检查脚本
- .6 六、配置优化
 - .6.1 rabbitmq.conf 核心配置
 - .6.2 系统优化
- .7 七、故障排查
 - .7.1 服务无法启动
 - .7.2 消息消费异常
 - .7.3 集群异常
 - .7.4 诊断命令
- .8 八、安全加固

- 8.1 修改默认用户
- 8.2 防火墙配置
- 8.3 SSL 配置
- 9 九、性能优化要点
 - 9.1 队列优化
 - 9.2 客户端优化
 - 9.3 监控指标
- 10 十、运维最佳实践

1 一、部署架构

1.1 整体架构



1.2 部署模式

部署模式	节点数	适用场景	高可用性
单机部署	1 节点	开发测试	无
普通集群	2+ 节点	生产环境	元数据高可用
镜像队列集群	3+ 节点	生产环境	元数据+数据高可用

2 二、快速安装

2.1 1. 环境准备

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

端口要求：

端口	用途
5672	AMQP 协议
15672	Web 管理界面
25672	集群节点间通信

2.2 2. 单机安装

```
# 创建目录
mkdir -p /apusic && cd /apusic

# 解压安装包
tar zxvf ADMQ-V2.0.6.xxx.tar.gz
cd admq-rabbitmq

# 放置 license.xml 授权文件到根目录
# 修改主机名（只能包含字母、数字、连字符）
hostname node1

# 启动服务
bin/admq-daemon start rabbitmq server
```

2.3 3. 集群安装 (3 节点)

所有节点执行：

```
mkdir -p /apusic && cd /apusic
tar zxvf ADMQ-V2.0.6.xxx.tar.gz
cd admq-rabbitmq

# 放置 license.xml
# 修改主机名
hostname node171 # node172, node173

# 配置 /etc/hosts
172.20.140.171 node171
172.20.140.172 node172
172.20.140.173 node173

# 启动服务
bin/admq-daemon start rabbitmq server
```

Node 1 执行集群创建：

```
bin/admq rabbitmq cluster add -s 172.20.140.172:58181 -u monitor -p
apusic_123
bin/admq rabbitmq cluster add -s 172.20.140.173:58181 -u monitor -p
apusic_123
```

开启镜像队列（可选）：

```
bin/admq rabbitmq admin ctl set_policy ha-all "^" '{"ha-
mode":"all","ha-sync-mode":"automatic"}'
```


3 三、核心概念

3.1 消息模型

```
Producer -> Exchange -> [Binding/Routing Key] -> Queue -> Consumer
```

3.2 交换机类型

类型	路由方式	适用场景
Direct	精确匹配 Routing Key	点对点消息
Fanout	广播到所有队列	广播通知
Topic	模式匹配 * #	日志分类、事件路由
Headers	基于消息头匹配	复杂条件路由

3.3 队列类型

类型	特点	适用场景
Classic Queue	传统队列，支持镜像	一般业务场景
Quorum Queue	基于 Raft，自动多副本	高可用场景
Stream Queue	只追加日志，支持多次读取	高吞吐、日志场景

4 四、常用命令速查

4.1 服务管理

```
# 启动/停止/重启
bin/admq-daemon start rabbitmq server
bin/admq-daemon stop rabbitmq server
bin/admq-daemon restart rabbitmq server

# 应用启停
bin/admq rabbitmq admin ctl start_app
bin/admq rabbitmq admin ctl stop_app
```

4.2 状态查看

```
# 节点状态
bin/admq rabbitmq admin ctl status

# 集群状态
bin/admq rabbitmq admin ctl cluster_status

# 健康检查
bin/admq rabbitmq admin diagnostics health_check

# 内存使用
bin/admq rabbitmq admin diagnostics memory_breakdown
```

4.3 用户管理

```
# 创建用户
bin/admq rabbitmq admin ctl add_user username password
bin/admq rabbitmq admin ctl set_user_tags username administrator
```

```
bin/admq rabbitmq admin ctl set_permissions -p /vhost username ".*"
".*" ".*"

# 修改密码
bin/admq rabbitmq admin ctl change_password username newpassword

# 删除用户
bin/admq rabbitmq admin ctl delete_user username

# 查看用户
bin/admq rabbitmq admin ctl list_users
```

4.4 队列管理

```
# 查看队列
bin/admq rabbitmq admin ctl list_queues
bin/admq rabbitmq admin ctl list_queues name messages consumers

# 查看交换机
bin/admq rabbitmq admin ctl list_exchanges name type

# 查看连接
bin/admq rabbitmq admin ctl list_connections peer_host state

# 查看通道
bin/admq rabbitmq admin ctl list_channels connection consumer_count
```

4.5 集群管理

```
# 查看集群状态
bin/admq rabbitmq admin ctl cluster_status

# 查看队列详情 (含镜像)
bin/admq rabbitmq admin ctl list_queues name pid slave_pids
```

```
synchronised_slave_pids
```

```
# 移除故障节点
```

```
bin/admq rabbitmq admin ctl forget_cluster_node rabbit@node2
```

4.6 策略管理

```
# 查看策略
```

```
bin/admq rabbitmq admin ctl list_policies
```

```
# 设置镜像策略
```

```
bin/admq rabbitmq admin ctl set_policy ha-all "^" '{"ha-mode":"all"}' --priority 0 --apply-to queues
```

```
# 清除策略
```

```
bin/admq rabbitmq admin ctl clear_policy ha-all
```

4.7 备份恢复

```
# 导出配置
```

```
bin/admq rabbitmq admin ctl export_definitions  
/backup/definitions.json
```

```
# 导入配置
```

```
bin/admq rabbitmq admin ctl import_definitions  
/backup/definitions.json
```

```
# 备份 Mnesia
```

```
bin/admq rabbitmq admin ctl eval  
'rabbit_mnesia:backup("/backup/mnesia.backup").'
```

5 五、日常运维

5.1 日志位置

日志类型	路径
服务日志	./log/rabbitmq/ rabbit@hostname.log
启动日志	./log/rabbitmq/ rabbit@hostname-sasl.log
管控台日志	./logs/

5.2 日常检查清单

检查项	命令	正常标准
节点状态	bin/admq rabbitmq admin ctl status	运行正常
集群状态	bin/admq rabbitmq admin ctl cluster_status	所有节点在线
队列堆积	bin/admq rabbitmq admin ctl list_queues messages	无异常堆积
内存使用	bin/admq rabbitmq admin ctl status	低于水位线
磁盘空间	df -h	高于 disk_free_limit

5.3 健康检查脚本

```
#!/bin/bash
# 节点健康检查
bin/admq rabbitmq admin diagnostics health_check

# 内存检查
bin/admq rabbitmq admin diagnostics memory_breakdown

# 告警队列检查 (消息数 > 10000)
bin/admq rabbitmq admin ctl list_queues name messages consumers |
awk '$2 > 10000 {print "WARNING: Queue \"$1\" has \"$2\" messages"}'
```

连接数检查

```
CONN_COUNT=$(bin/admq rabbitmq admin ctl list_connections | wc -l)
if [ $CONN_COUNT -gt 1000 ]; then
    echo "WARNING: Connection count $CONN_COUNT exceeds threshold"
fi
```

6 六、配置优化

6.1 rabbitmq.conf 核心配置

```
# 网络
listeners.tcp.default = 5672
listeners.ssl.default = 5671

# 内存 (推荐 0.6-0.7)
vm_memory_high_watermark.relative = 0.6
vm_memory_high_watermark.paging_ratio = 0.5

# 磁盘 (推荐绝对值 1GB)
disk_free_limit.relative = 1.5
disk_free_limit.absolute = 1GB

# 连接
channel_max = 2047
heartbeat = 30

# 队列
queue_master_locator = min-masters
```

6.2 系统优化

```
# 文件描述符
ulimit -n 65535

# 内核参数 (/etc/sysctl.conf)
net.ipv4.ip_local_port_range = 1024 65535
net.ipv4.tcp_tw_reuse = 1
net.core.somaxconn = 65535
vm.swappiness = 10
```

磁盘调度

```
echo deadline > /sys/block/sda/queue/scheduler
```


7 七、故障排查

7.1 服务无法启动

现象	可能原因	解决方案
端口被占用	5672/15672 被占用	停止占用进程或修改端口
Cookie 不一致	集群节点 Cookie 不匹配	同步 .erlang.cookie
内存不足	系统内存不足	增加内存或降低水位线
磁盘不足	低于 disk_free_limit	清理磁盘或调整限制

7.2 消息消费异常

现象	可能原因	解决方案
消息堆积	消费者过慢或离线	增加消费者或优化逻辑
消息丢失	未启用持久化	启用 delivery_mode=2
重复消费	确认超时	优化处理逻辑，及时确认

7.3 集群异常

现象	可能原因	解决方案
节点离线	网络故障	检查网络，重启服务
脑裂	网络分区	使用 forget_cluster_node 清理
队列不可用	节点离线	使用镜像队列或 Quorum Queue

7.4 诊断命令

```
# 查看详细信息
bin/admq rabbitmq admin ctl status
bin/admq rabbitmq admin ctl cluster_status
bin/admq rabbitmq admin ctl list_connections peer_host peer_port
state
```

```
bin/admq rabbitmq admin ctl list_queues name messages consumers  
memory
```

查看内存/进程

```
bin/admq rabbitmq admin diagnostics memory_breakdown  
bin/admq rabbitmq admin diagnostics processes  
bin/admq rabbitmq admin diagnostics connections
```

8 八、安全加固

8.1 修改默认用户

```
# 删除默认 guest 用户
bin/admq rabbitmq admin ctl delete_user guest

# 创建新管理员
bin/admq rabbitmq admin ctl add_user admin strongpassword
bin/admq rabbitmq admin ctl set_user_tags admin administrator
bin/admq rabbitmq admin ctl set_permissions -p / admin ".*" ".*"
".*"
```

8.2 防火墙配置

```
firewall-cmd --permanent --add-port=5672/tcp
firewall-cmd --permanent --add-port=15672/tcp
firewall-cmd --permanent --add-port=25672/tcp
firewall-cmd --reload
```

8.3 SSL 配置

```
listeners.ssl.default = 5671
ssl_options.cacertfile = /path/to/ca_certificate.pem
ssl_options.certfile = /path/to/server_certificate.pem
ssl_options.keyfile = /path/to/server_key.pem
ssl_options.verify = verify_peer
ssl_options.fail_if_no_peer_cert = false
```

9 九、性能优化要点

9.1 队列优化

```
Map<String, Object> args = new HashMap<>();
// 队列长度限制
args.put("x-max-length", 100000);
args.put("x-max-length-bytes", 1073741824); // 1GB

// 消息 TTL
args.put("x-message-ttl", 3600000); // 1小时

// 惰性队列（减少内存）
args.put("x-queue-mode", "lazy");
```

9.2 客户端优化

```
// 连接池 + 自动恢复
factory.setAutomaticRecoveryEnabled(true);
factory.setNetworkRecoveryInterval(5000);

// 单连接多通道
Connection connection = factory.newConnection();
for (int i = 0; i < 10; i++) {
    Channel channel = connection.createChannel();

// QoS 预取
channel.basicQos(100);

// 批量确认
channel.basicAck(deliveryTag, true);
}
```

9.3 监控指标

指标	说明
rabbitmq_connections	连接数
rabbitmq_queue_messages	总消息数
rabbitmq_node_mem_used	内存使用量
rabbitmq_node_disk_free	磁盘剩余空间

10 十、运维最佳实践

1. **定期检查**：建立日常检查机制
 2. **监控告警**：配置完善的监控告警体系
 3. **备份策略**：定期备份配置和数据
 4. **容量规划**：提前评估并扩容
 5. **安全更新**：及时更新安全补丁
 6. **最小权限**：为业务用户分配最小必要权限
 7. **灾难演练**：定期进行故障恢复演练
-

技术支持：

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

