

## 架构实战营模块8 - 第3课

# 集群架构设计技巧

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

## 李运华

前阿里资深技术专家（P9）

# 教学目标

1. 学习 Redis sentinel 集群的设计技巧
2. 学习 MongoDB replication 集群的设计技巧



它山之石可以攻玉!

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 目录

1. Redis Sentinel 设计技巧

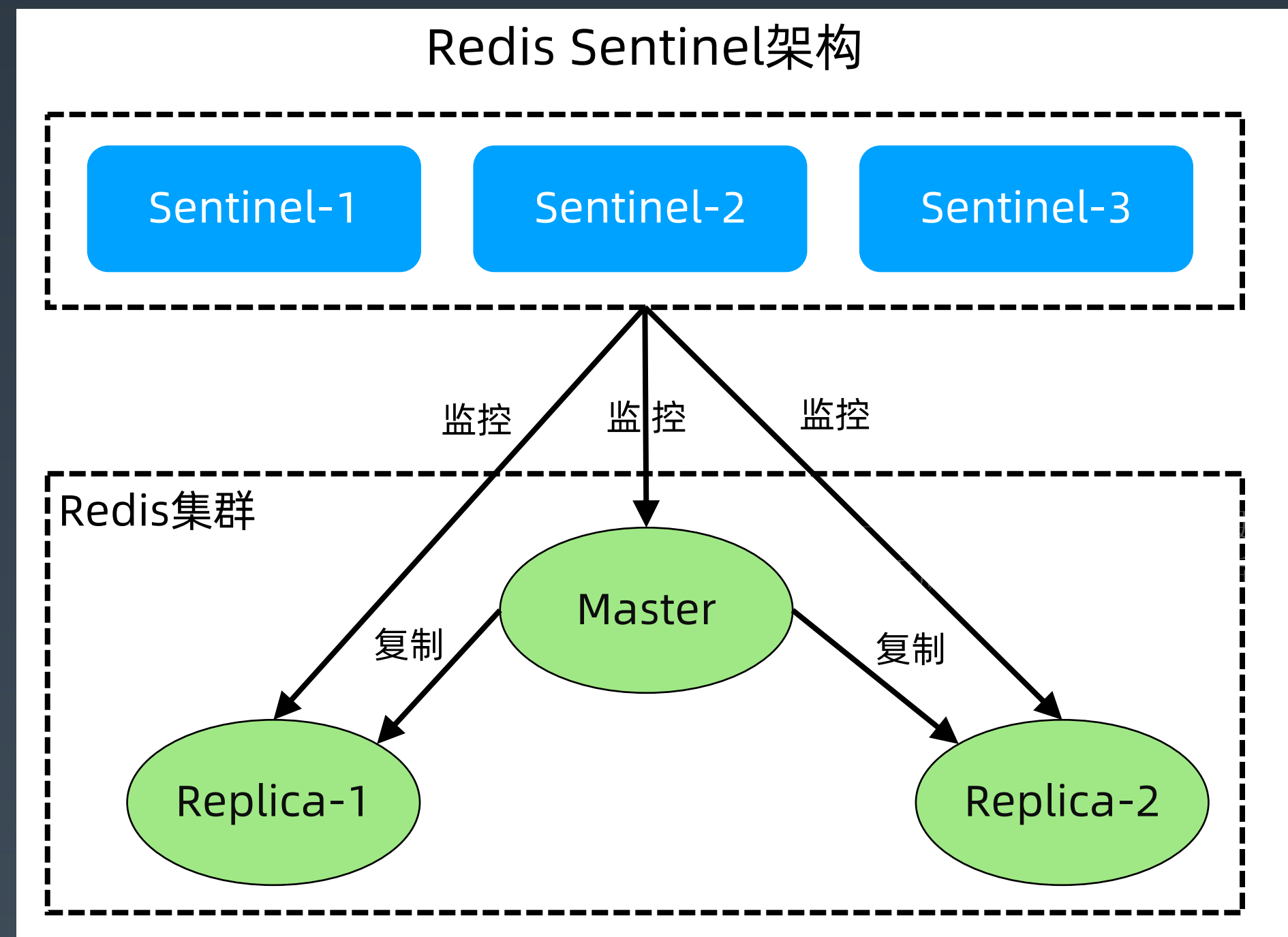
2. MongoDB Replication 设计技巧

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 1. Redis Sentinel 设计技巧

手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# Redis Sentinel 基本架构



惠血

## 【Monitoring】

Sentinel 可以监控 Redis 节点的状态。

## 【Notification】

Sentinel 可以通过 API 进行集群状态通知。

## 【Automatic failover】

Sentinel 实现故障自动切换。

## 【Configuration provider】

Sentinel 为 client 提供发现 master 节点的发现功能，如果发生了切换，Sentinel 会通知 client 新的 master 地址。

## 【实现细节】

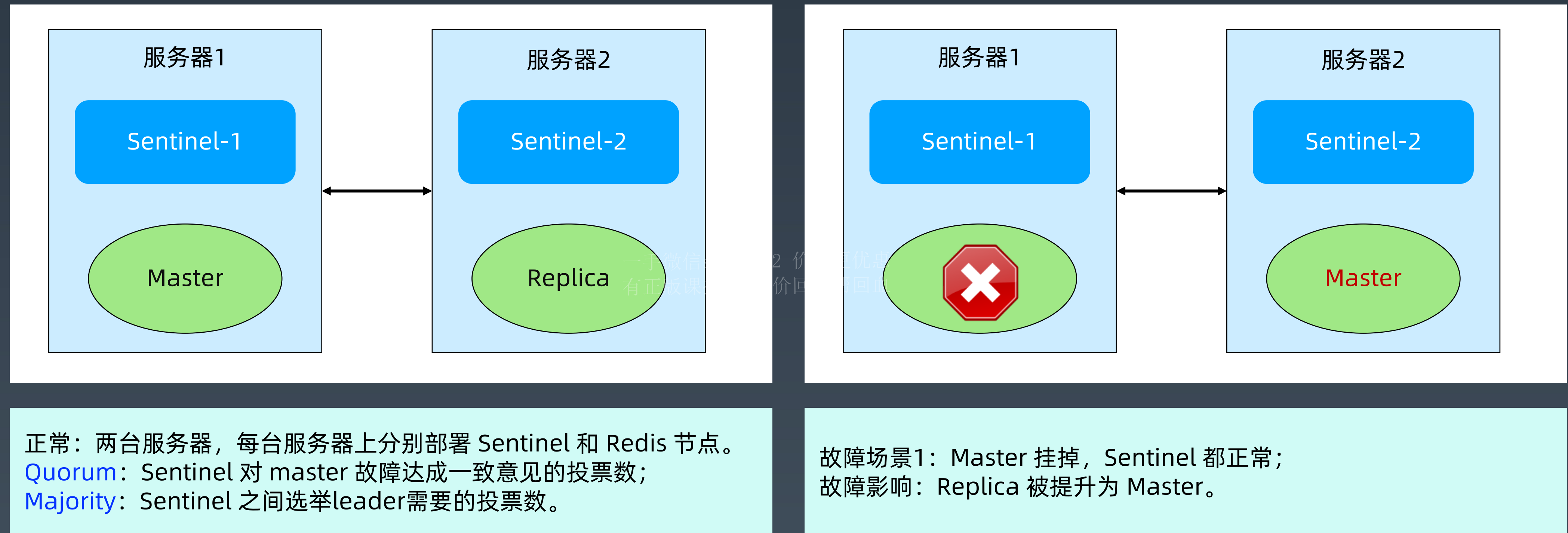
1. Sentinel 的选举是 Raft 算法；
2. Sentinel 是独立运行的程序，但不是独立的代码：  
`redis-server /path/to/sentinel.conf -sentinel。`

[学习链接](#)



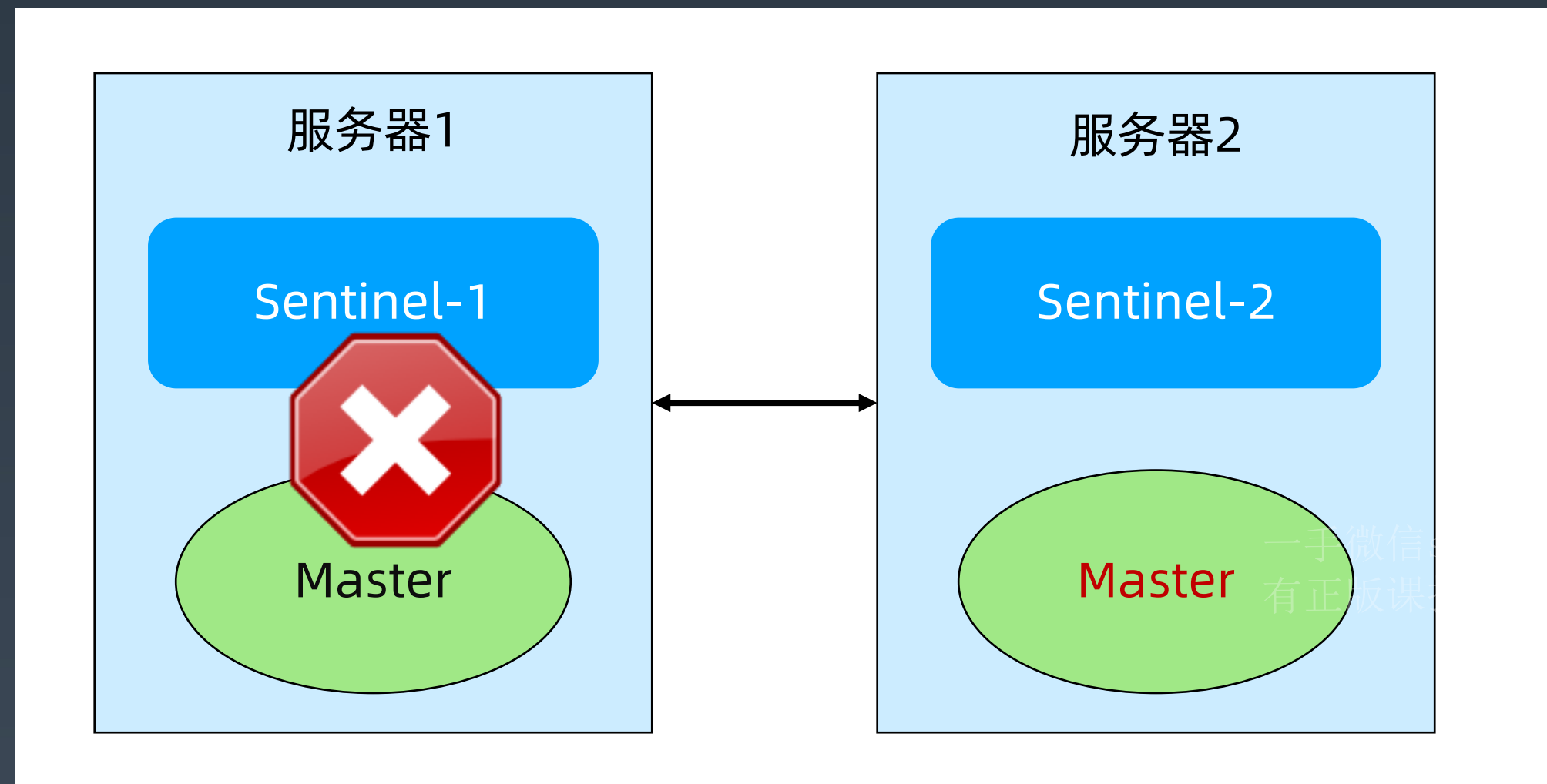
为什么 Redis 采用 Sentinel 这种架构模式，而不是 ZooKeeper，或者像 MongoDB 一样直接内置集群选举功能？

# Sentinel 架构模式1 - 双节点(1/2)

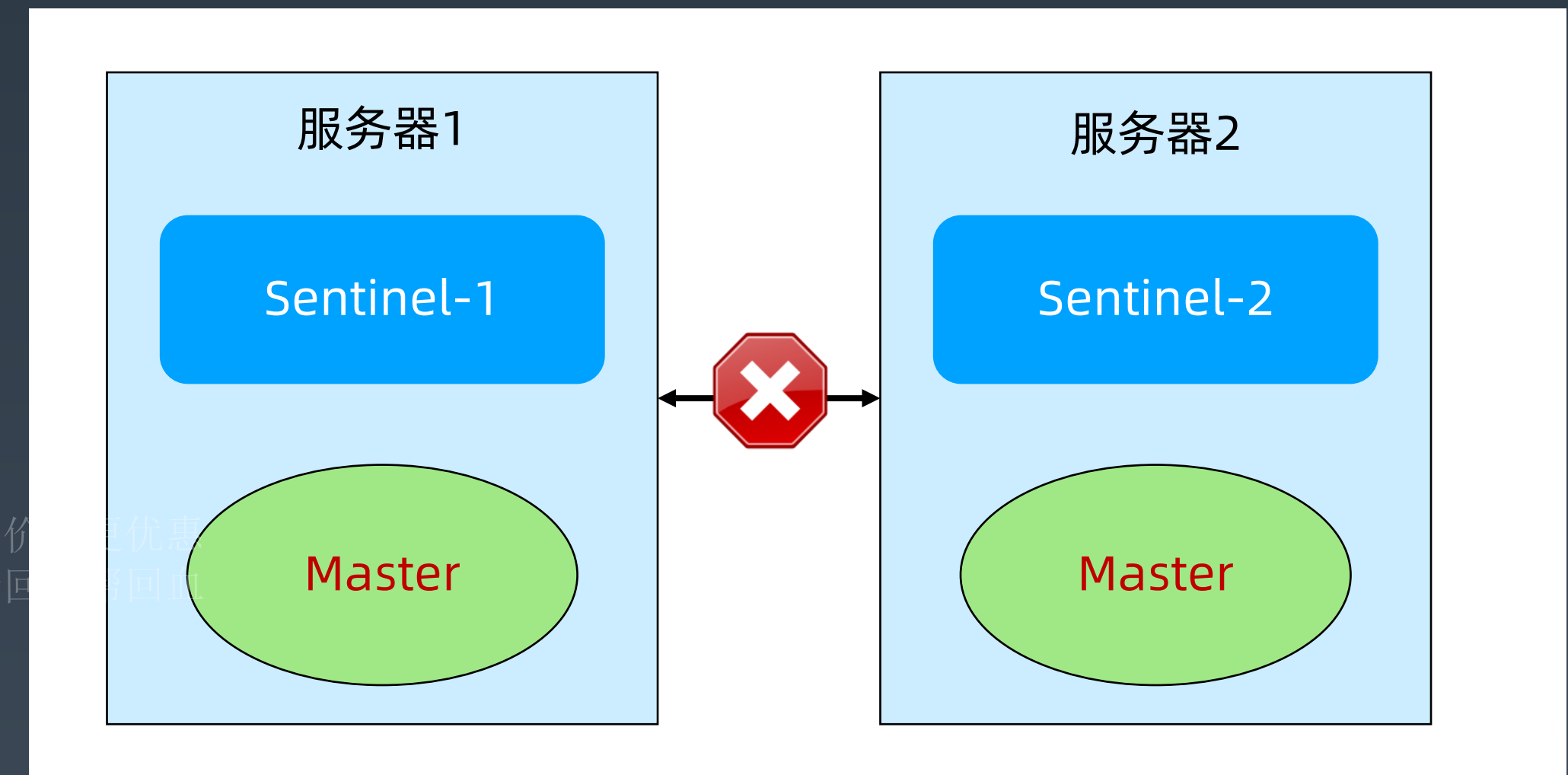


仅做分析，实际应用中不能这么设计！！

# Sentinel 架构模式1 - 双节点(2/2)



故障场景2：服务器1挂掉，quorum=1，majority=2；  
故障影响：无论服务器2是否挂掉，集群都宕机。

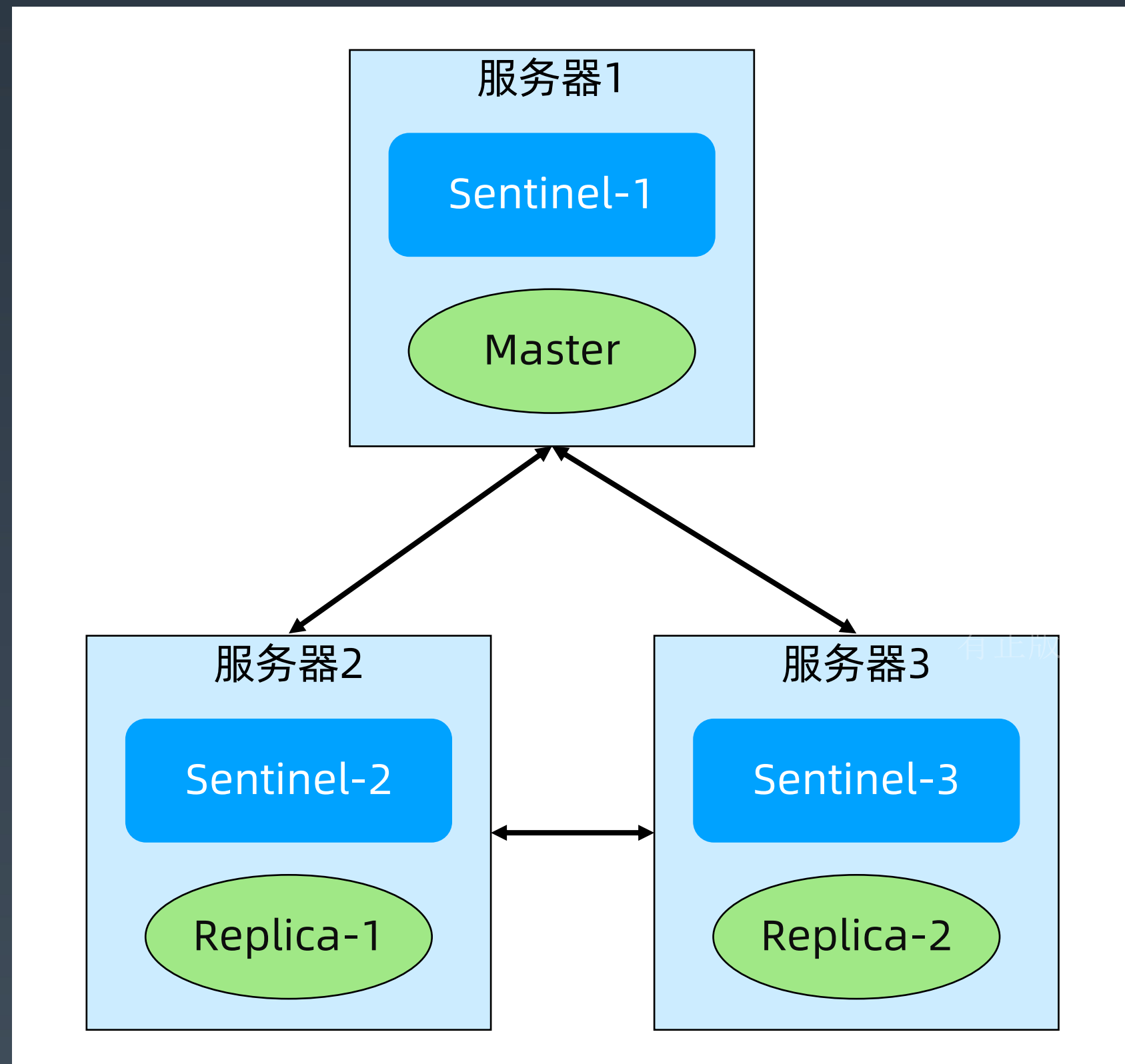


故障场景3：服务器1和服务器2的连接挂掉，quorum=1，majority=1；  
故障影响：双主（脑裂）。

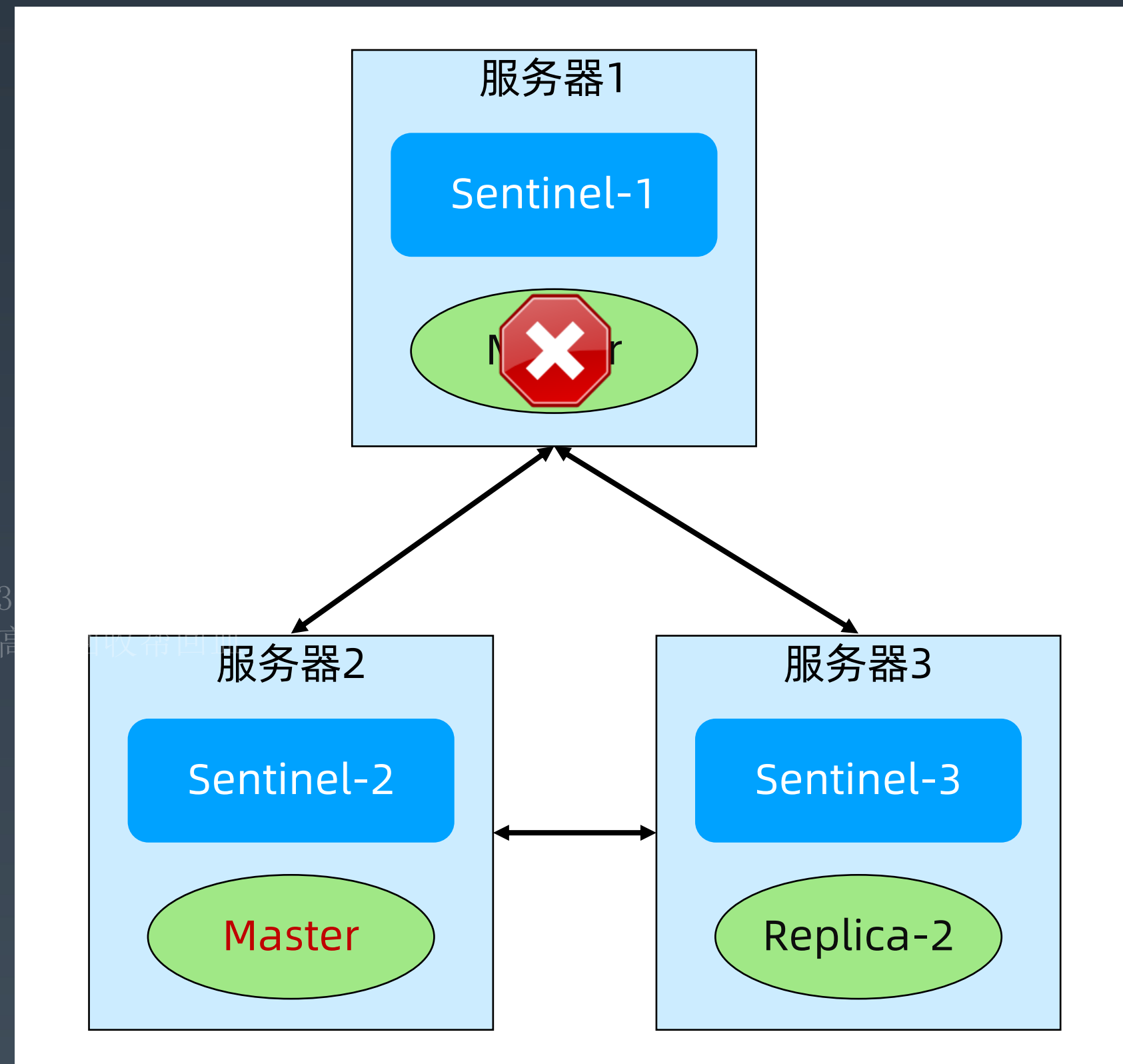


不能提供服务 还是 脑裂 更严重？

# Sentinel 架构模式2 - 三节点(1/2)



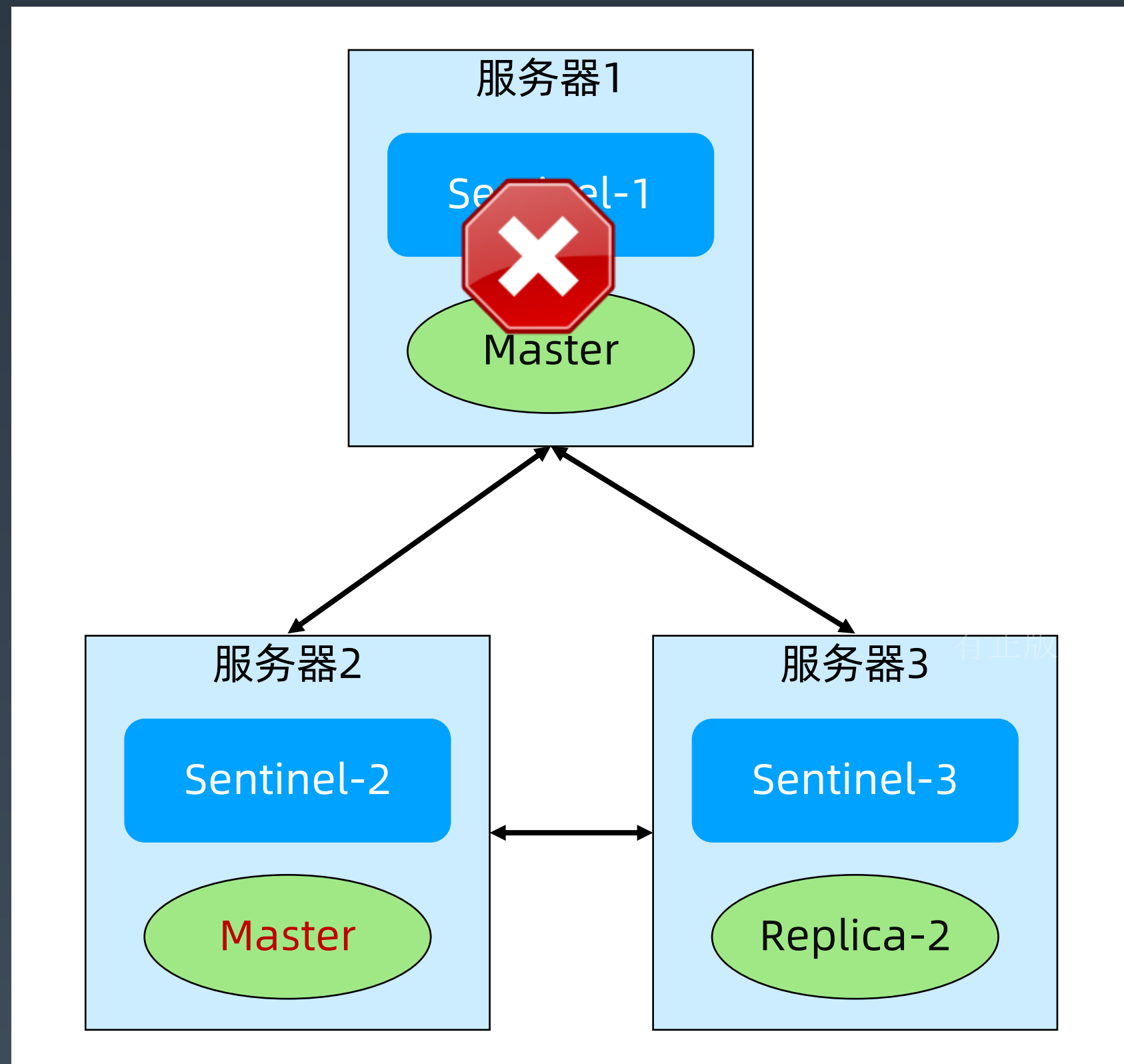
正常：三台服务器，每台服务器上分别部署 Sentinel 和 Redis 节点。



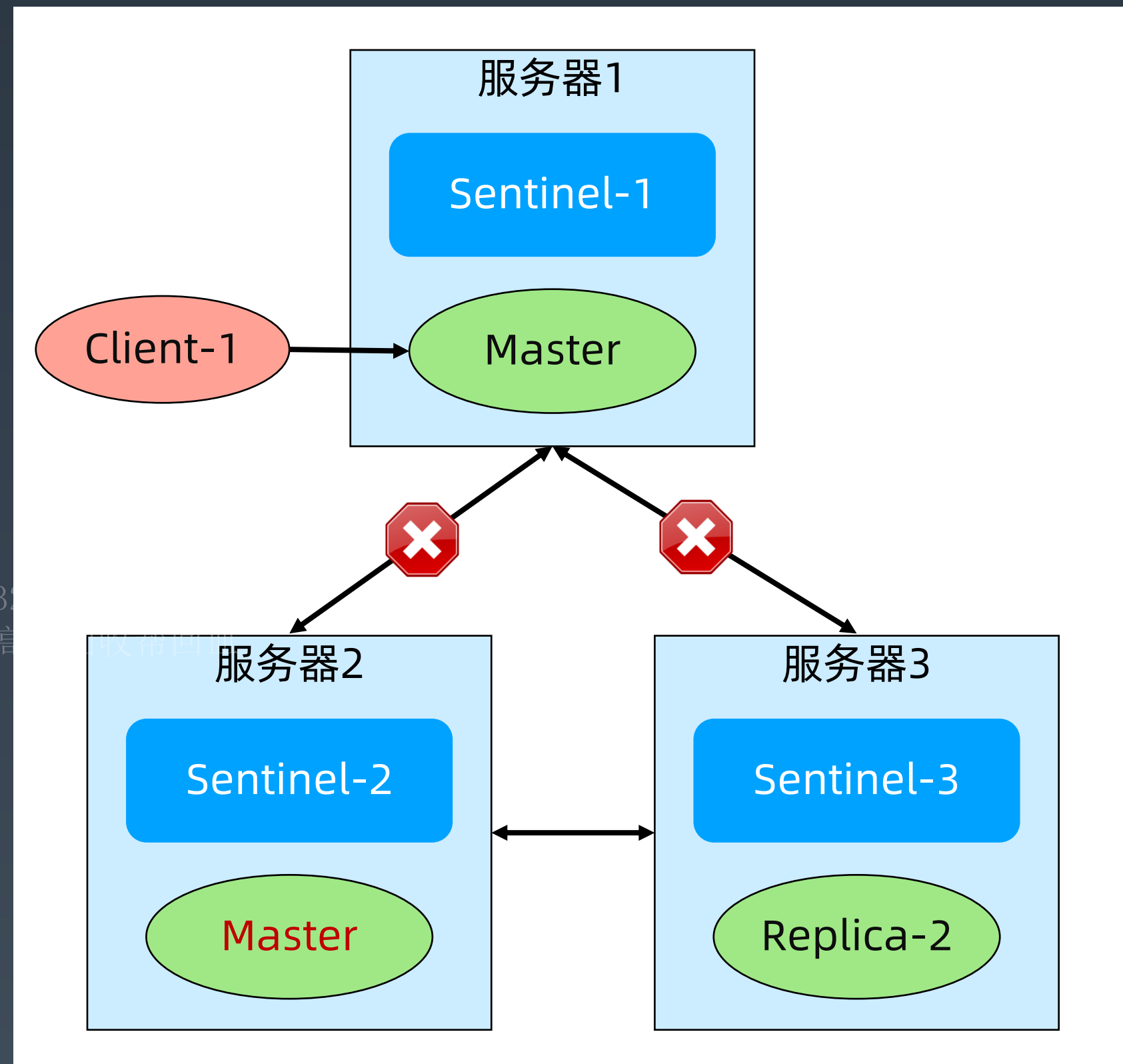
故障场景1：Master挂掉，Sentinel 都正常；  
故障影响：其中1个 Replica 被提升为 Master。



# Sentinel 架构模式2 - 三节点(2/2)

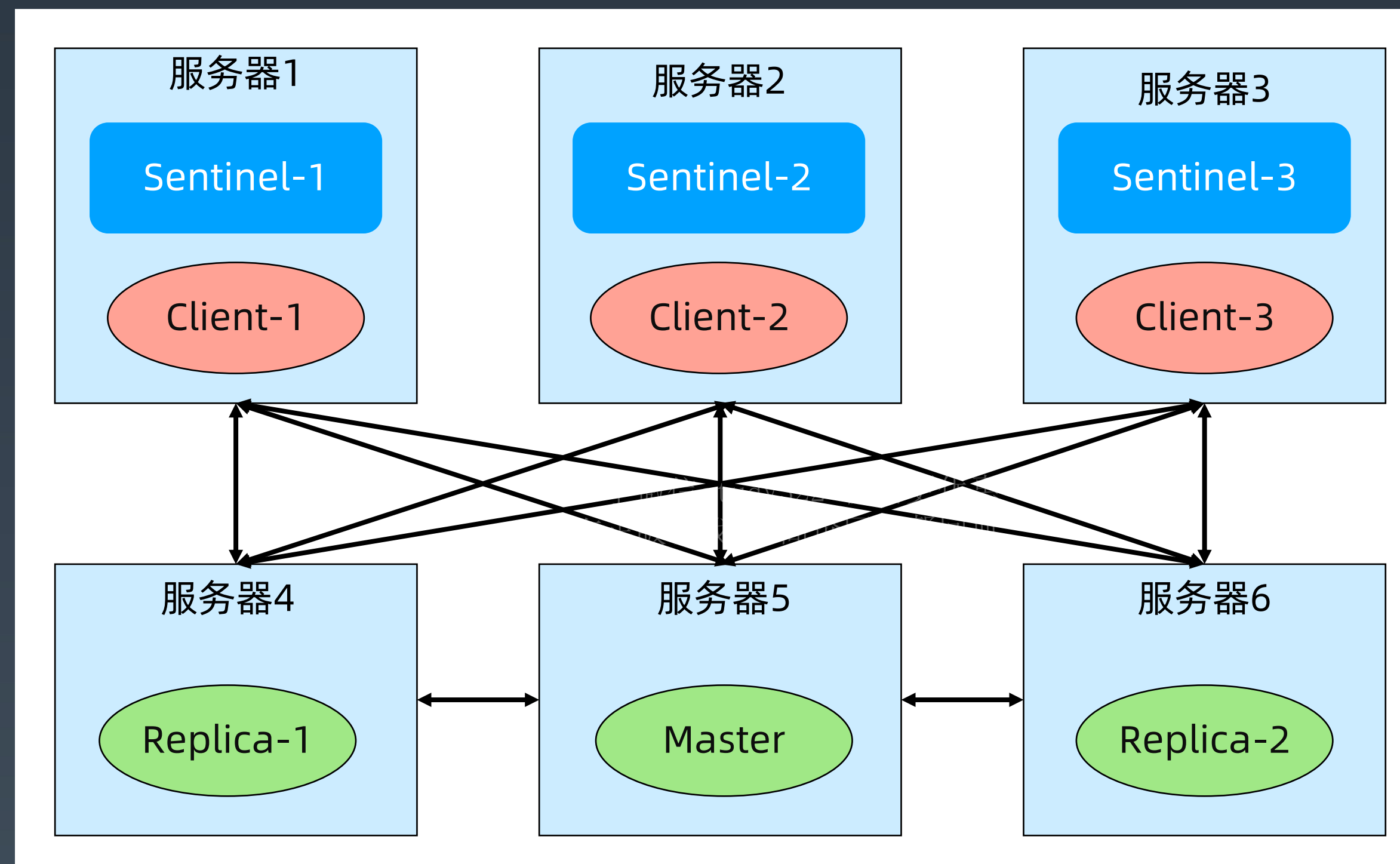


故障场景2：服务器1挂掉，quorum = 2，majority=2；  
故障影响：其中1个 Replica 被提升为 Master。



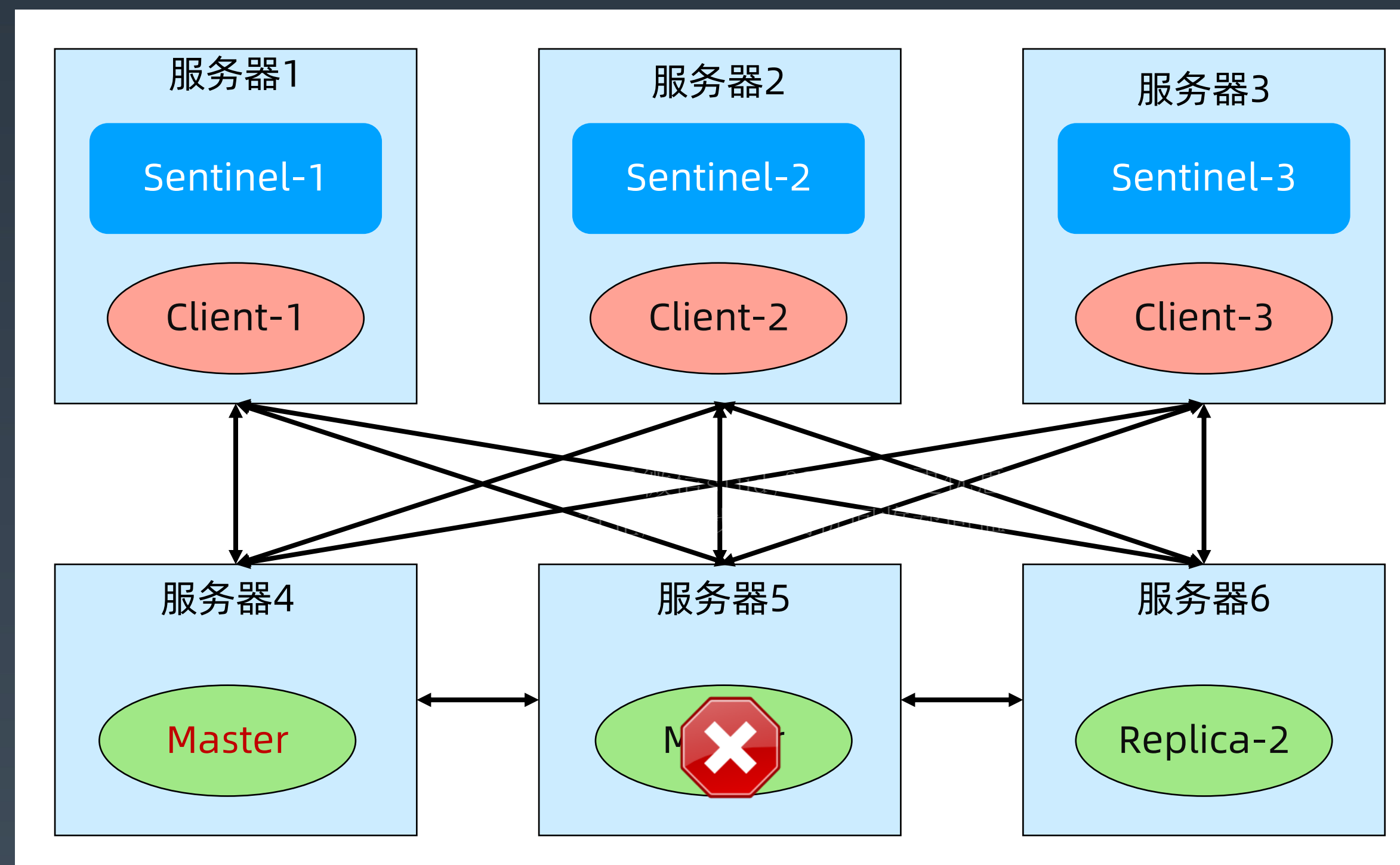
故障场景3：服务器1与服务器2和服务器3断连；  
故障影响：其中1个 Replica 被提升为 Master，可能出现双主；  
解决方案：min-replicas-to-write 1；min-replicas-max-lag 10。

# Sentinel 架构模式3 - 分离部署(1/3)



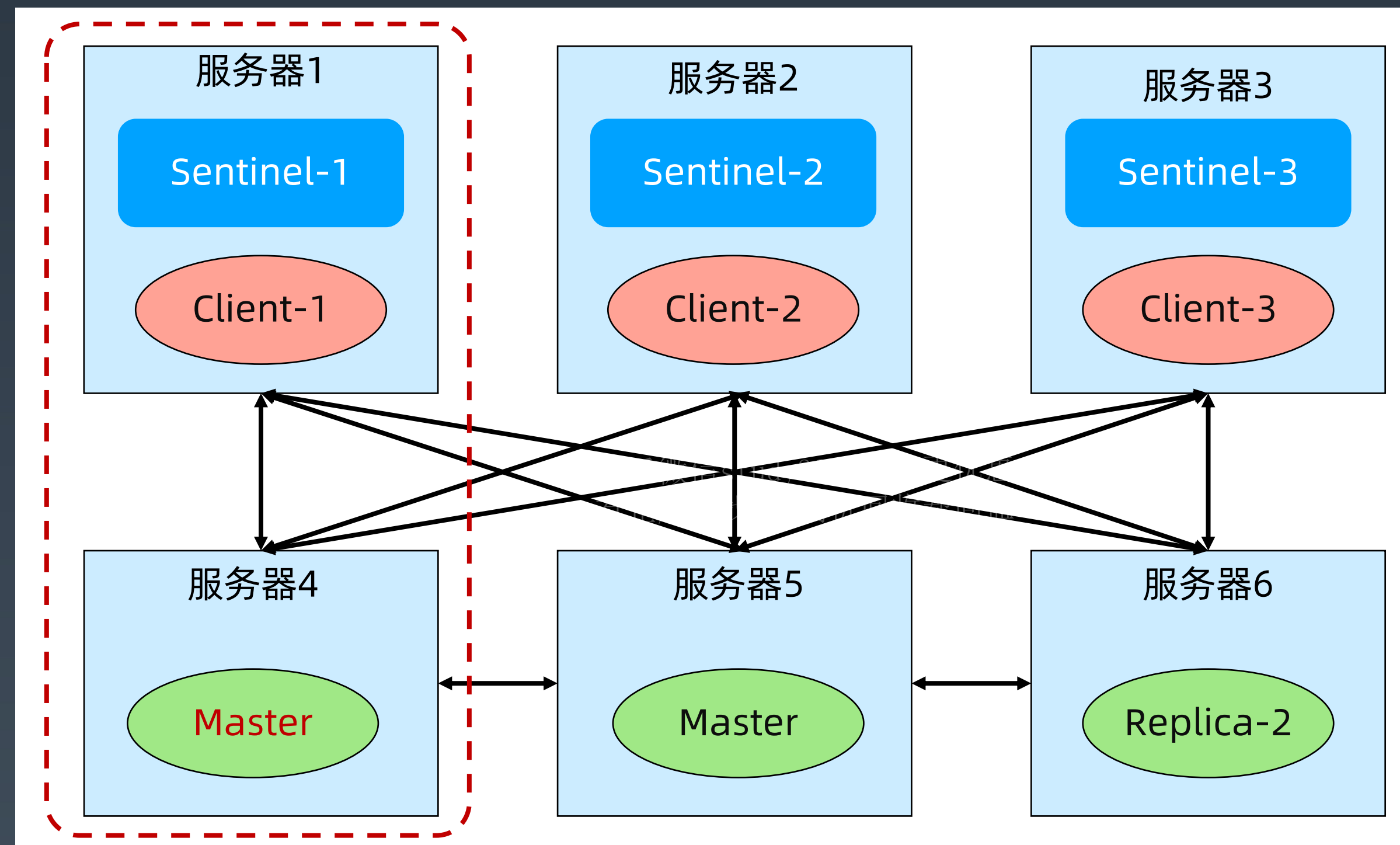
正常：Sentinel 和 Redis 分开部署，可以将 Sentinel 和 Redis 客户端所在的应用部署在一起。

# Sentinel 架构模式3 - 分离部署(2/3)



故障场景1: Master 挂掉, Sentinel 都正常;  
故障影响: 其中1个 Replica 被提升为 Master。

# Sentinel 架构模式3 - 分离部署(3/3)



故障场景2：服务器1和服务器4形成分区，剩余的服务器形成另外一个分区；  
故障影响：双主（脑裂）；  
解决方案：min-replicas-to-write 1；min-replicas-max-lag 10。

# Redis 集群架构模式对比

架构模式	硬件成本	架构复杂度	双主	故障处理能力
双节点	2台服务器	低	是，Majority=1	只能支持Master节点故障，其它故障不支持
三节点	3台服务器	中	是，网络分区	支持Master节点故障，Sentinel节点故障
节点分离	多台服务器，Sentinel可以和业务服务器共用	高	是，网络分区	支持Master节点故障，Sentinel节点故障

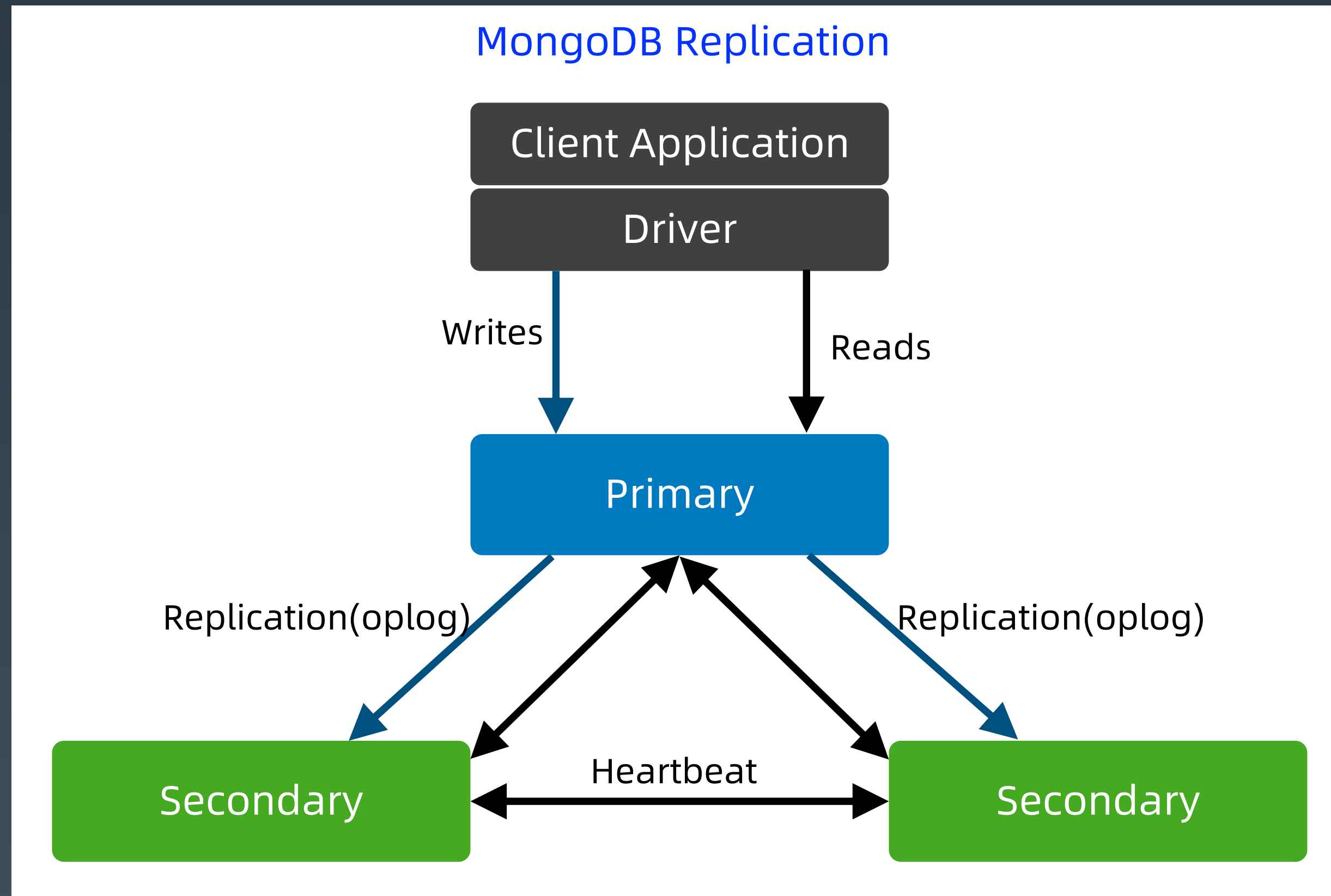


应该优先用哪个架构模式？原因是什么？

## 2. MongoDB Replication 设计技巧

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# MongoDB Replication 基本架构

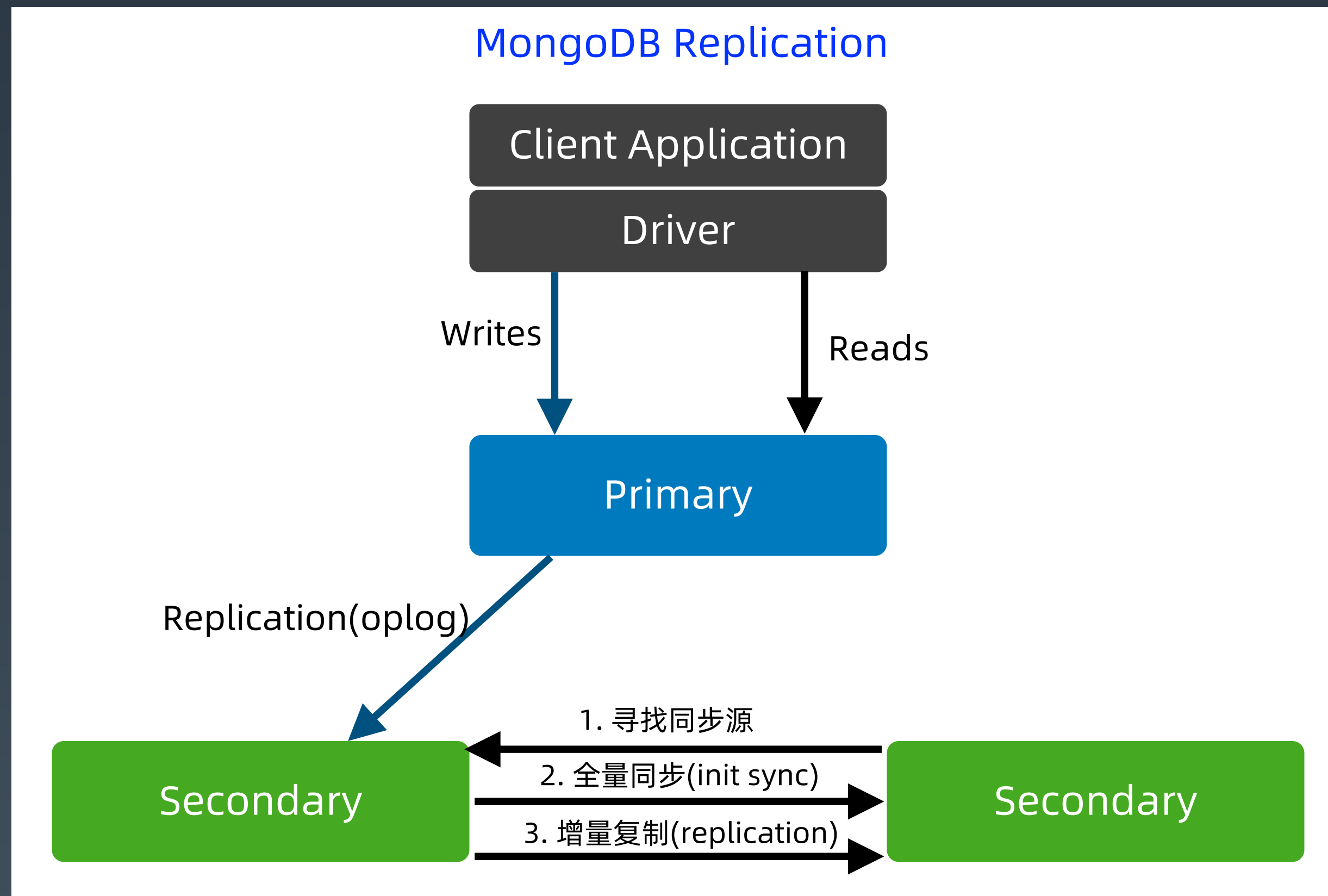


基本实现：

1. Primary 处理所有 Write 请求，Secondary 可以处理 Read 请求，或者只复制数据；
2. 异步复制，复制的是 oplog；
3. 选举算法：3.2.0 以前基于 bully 算法，3.2.0 开始基于 Raft 算法；
4. 最多50个节点，但最多只有7个节点参与选举。

[学习链接](#)

# MongoDB Replication - 新节点同步流程



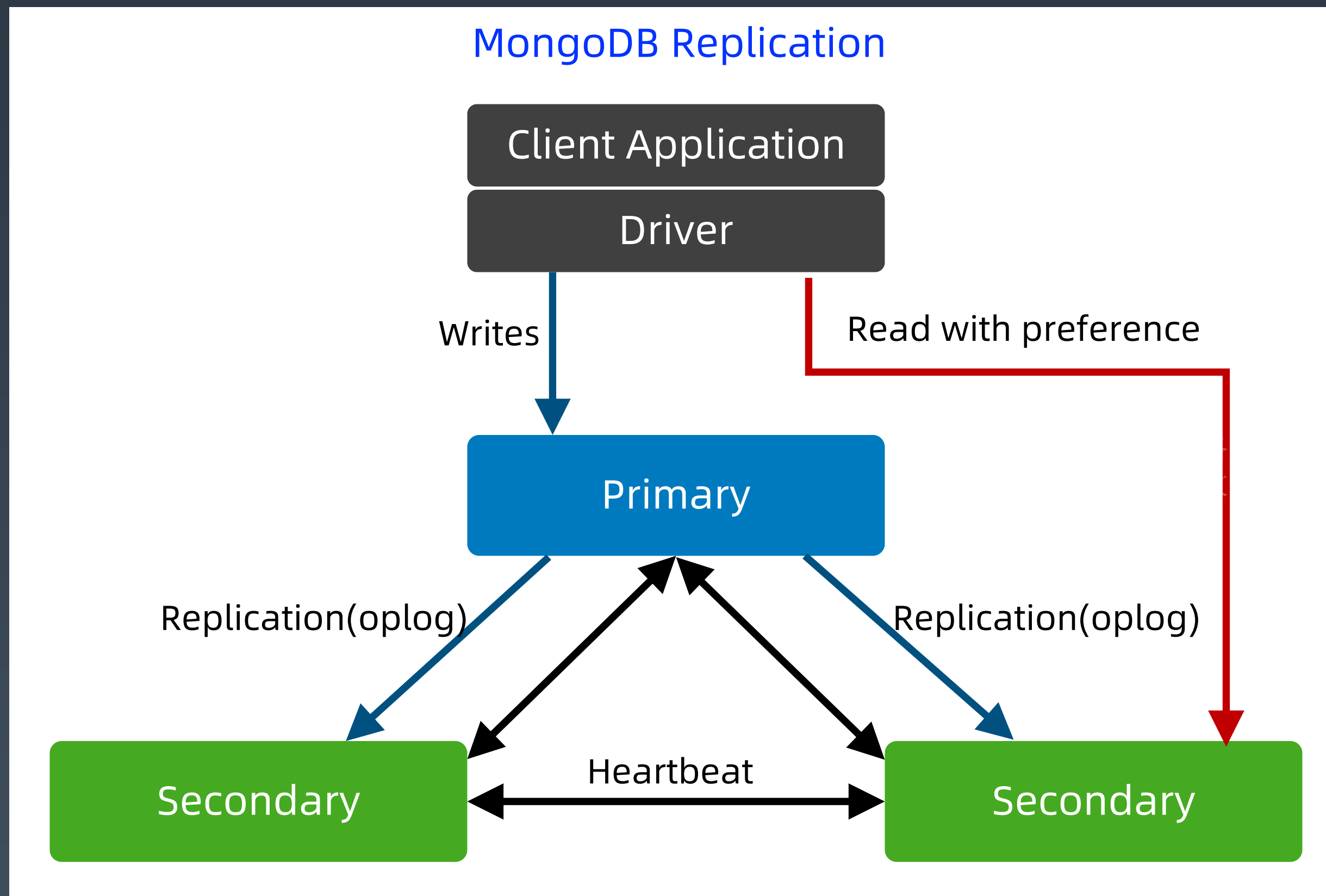
基本实现：

1. 第1步：寻找同步源。MongoDB 支持级联复制，复制源不一定是 Primary，而是通过算法选出来的；
2. 第2步：全量复制数据和 oplog；
3. 第3步：同步全量复制后的增量数据，异步复制 oplog。

[学习链接](#)



# MongoDB Replication 架构技巧1 - Read reference

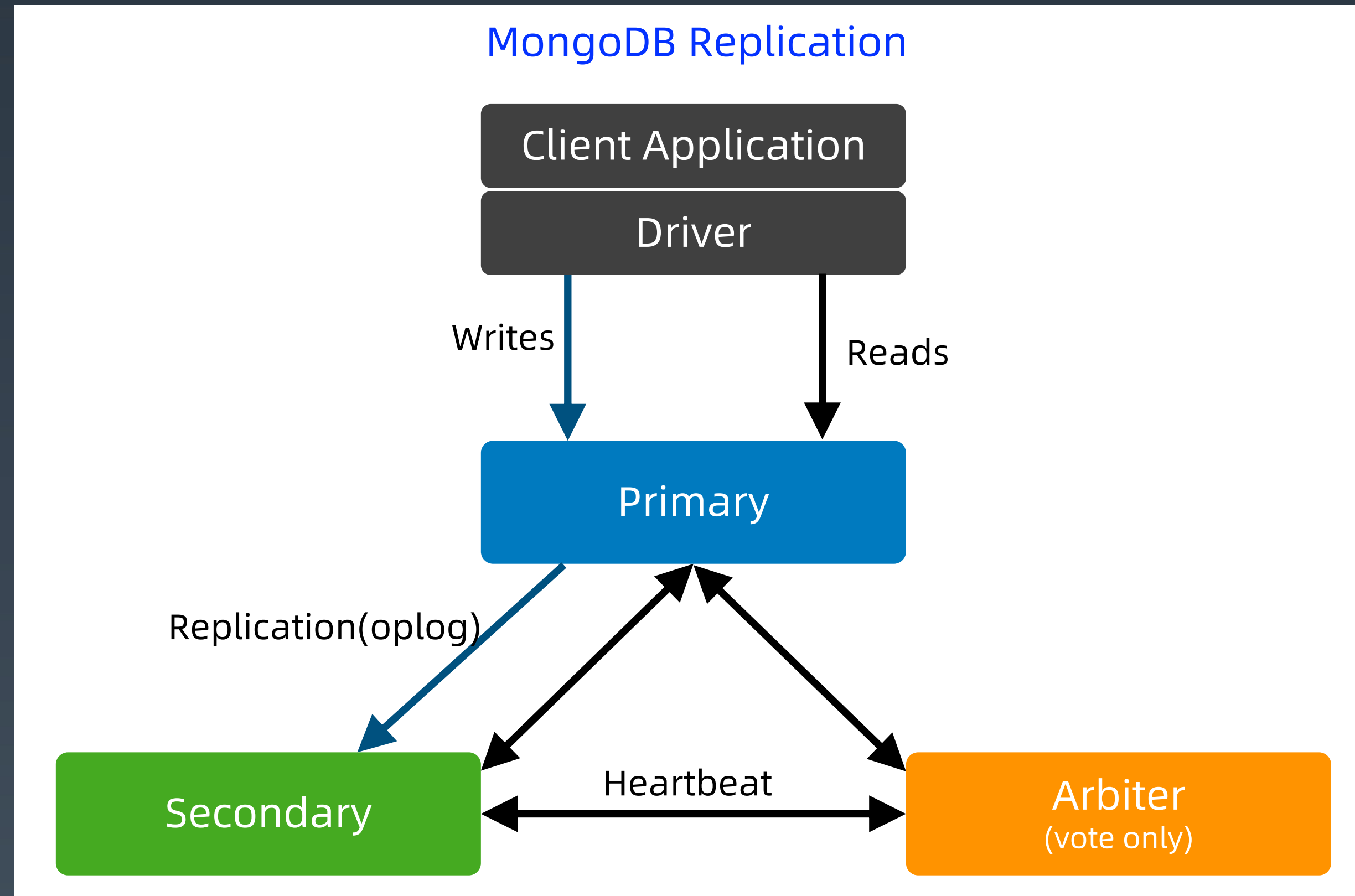


基本实现：

1. 默认读 Primary，指定 read preference 来读取 Secondary；
2. 可能读不到最新数据（why）；
3. 事务必须读 Primary；
4. 包含5种 preference：primary, primaryPreferred, secondary, secondaryPreferred, nearest.
5. 通过 API 或者连接配置参数：[学习样例](#)。

[学习链接](#)

# MongoDB Replication 架构技巧2 - Arbiter



基本实现：

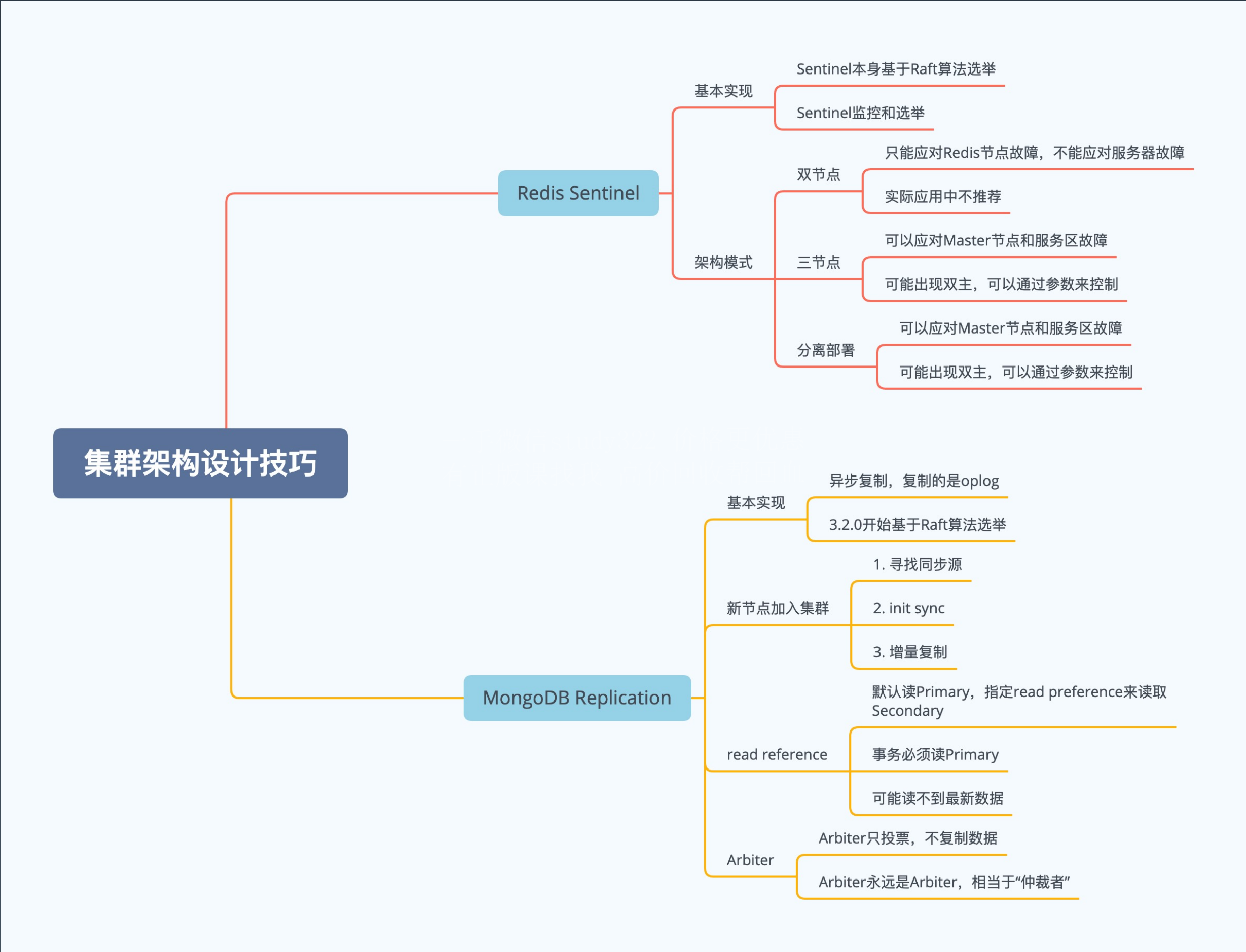
1. Arbiter 只投票，不复制数据；
2. Arbiter 永远是 Arbiter，相当于“仲裁者”。

[学习链接](#)



Arbiter 具体有什么用？直接两个节点做主备架构不行吗？

# 本节思维导图



# 随堂测验

## 【判断题】

1. Redis sentinel 选举功能和 MongoDB 的选举功能都是内置的。
2. Redis sentinel 的 quorum 是 sentinel 选举所需的最低票数。
3. Redis sentinel 集群可能出现双主现象。
4. MongoDB replication 架构一定是读写分离的。
5. MongoDB arbiter 节点可以参与选举并成为 master 节点。

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

## 【思考题】

对比一下 Redis sentinel 和 MongoDB replication 的实现异同和优缺点。

# Q&A





# 茶歇时间



八卦，趣闻，内幕.....

THANKS

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血