

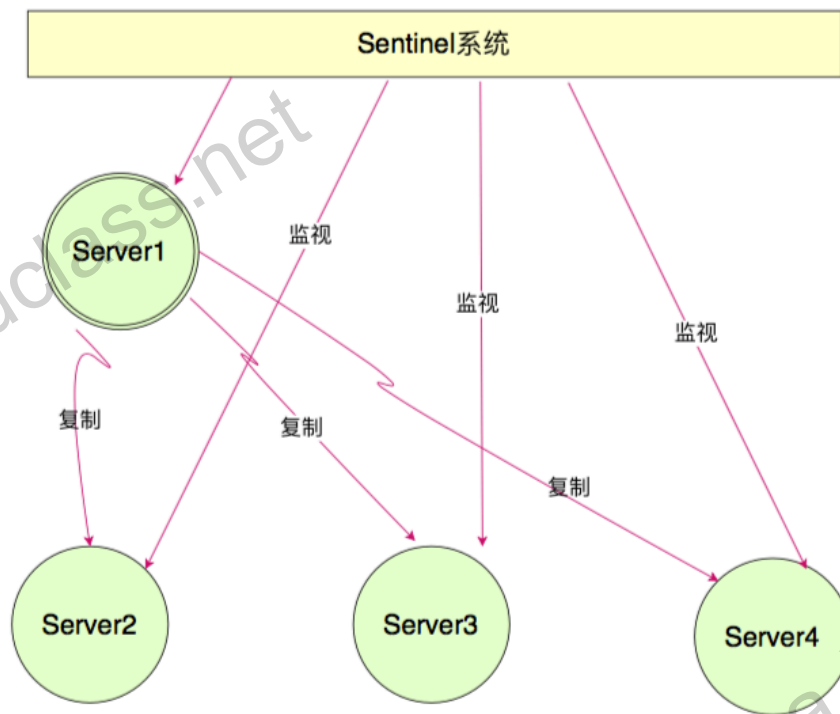


## 说说什么是哨兵机制

- Redis 的哨兵(sentinel) 系统用于管理多个 Redis 服务器,该系统执行以下三个任务:
  - 监控(**Monitoring**): 哨兵(sentinel) 会不断地检查你的 Master 和 Slave 是否运作正常。
  - 提醒(**Notification**): 当被监控的某个 Redis 出现问题时, 哨兵(sentinel) 可以通过 API 向管理员或者其他应用程序发送通知。
  - 自动故障迁移(**Automatic failover**): 当一个 Master 不能正常工作时, 哨兵 (sentinel) 会开始一次自动故障迁移操作, 它会将失效 Master 的其中一个 Slave 升级为新的 Master, 并让失效 Master 的其他 Slave 改为复制新的 Master; 当客户端试图连接失效的 Master 时, 集群也会向客户端返回新 Master 的地址, 使得集群可以使用 Master 代替失效 Master
- 哨兵(sentinel) 是一个分布式系统, 你可以在一个架构中运行多个哨兵(sentinel) 进程, 这些进程使用流言协议(gossip protocols)来接收关于 Master 是否下线的信息, 并使用投票协议(agreement protocols)来决定是否执行自动故障迁移, 以及选择哪个 Slave 作为新的 Master。
- 每个哨兵(sentinel) 会向其它哨兵(sentinel)、master、slave 定时发送消息, 以确认对方是否“活”着, 如果发现对方在指定时间(可配置)内未回应, 则暂时认为对方已挂(所谓的“主观认为宕机” **Subjective Down**, 简称 **sdown**)。
- 若“哨兵群”中的多数 sentinel, 都报告某一 master 没响应, 系统才认为该 master “彻底死亡”(即: 客观上的真正 **down** 机, **Objective Down**, 简称 **odown**), 通过一定的 vote 算法, 从剩下的 slave 节点中, 选一台提升为 master, 然后自动修改相关配置



## Redis高可用之哨兵模式



小D课堂



pdfelement

试用版