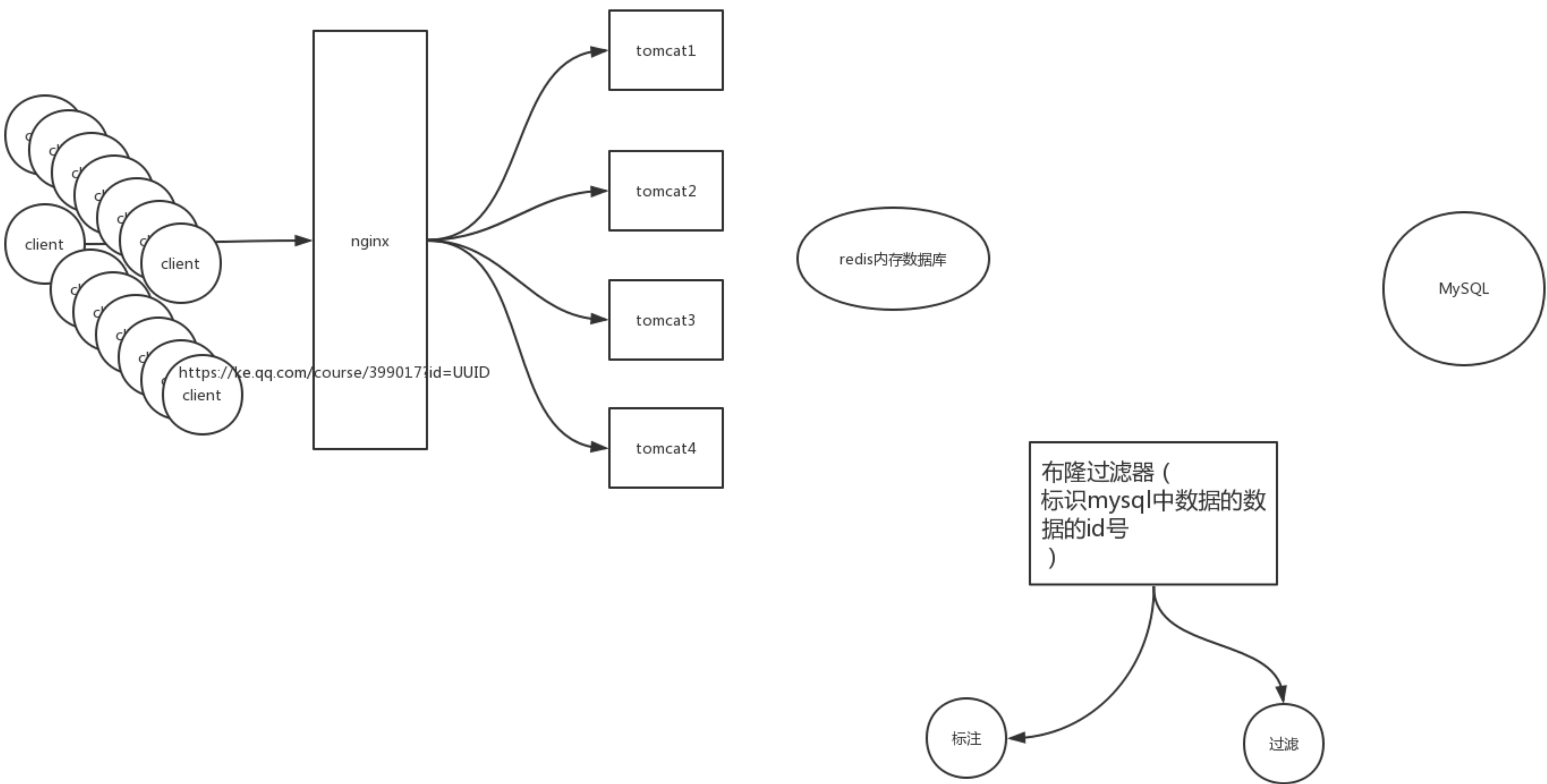


单机->多机->微服务

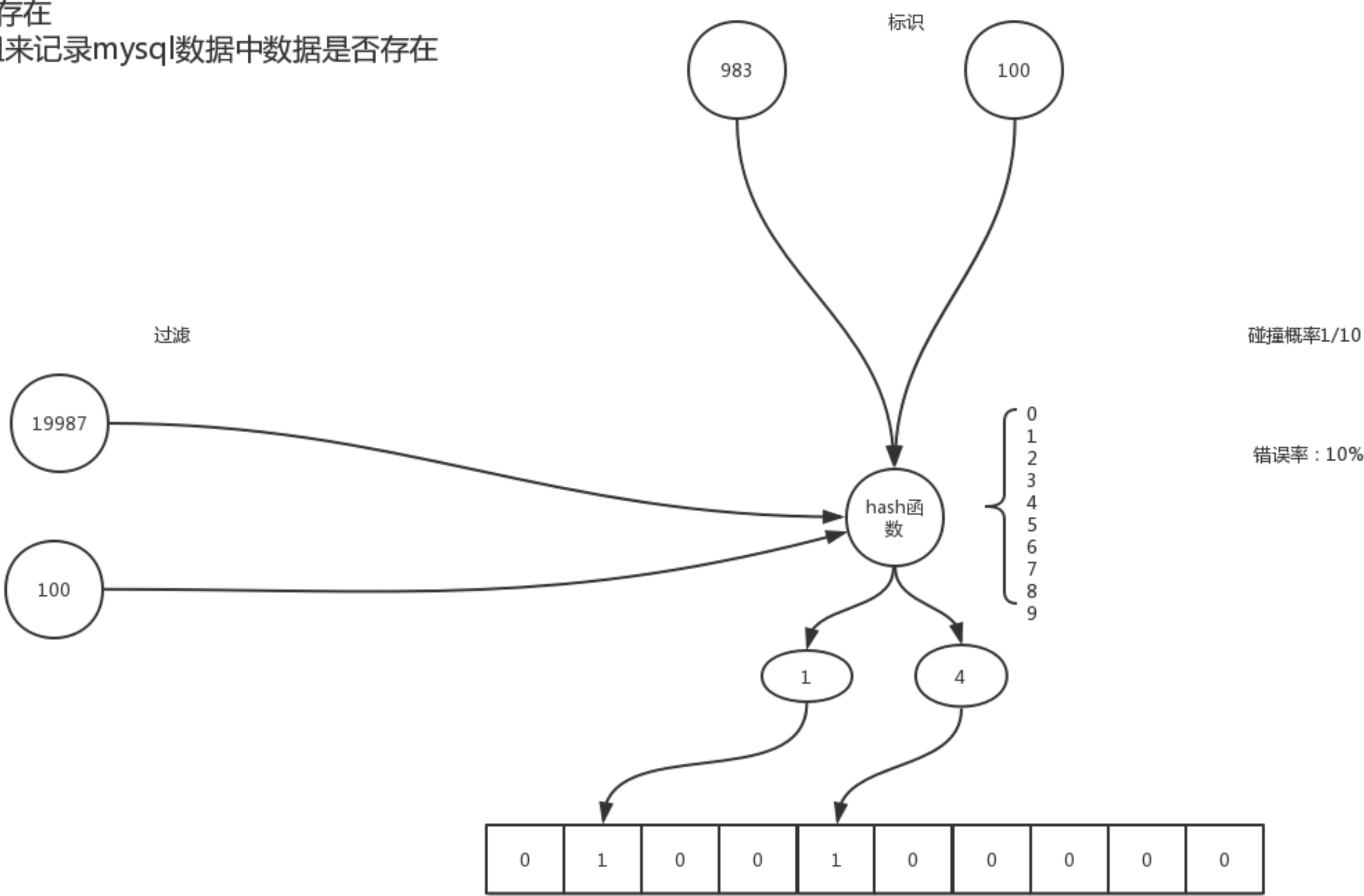
如果发生了大量的缓存穿透。那么mysql很快就会达到瓶颈问题

场景2、黑客攻击：如果有大量的不同的key取后台取数据咋办？uuid，布隆算法



宁可错杀三千，绝不放过一个

布隆过滤器 是一个使用错误率来换取空间和时间的算法
错误率主要体现在：他说数据存在，那么不一定存在
不存在，一定不存在
它是使用bit数组来记录mysql数据中数据是否存在
原理：

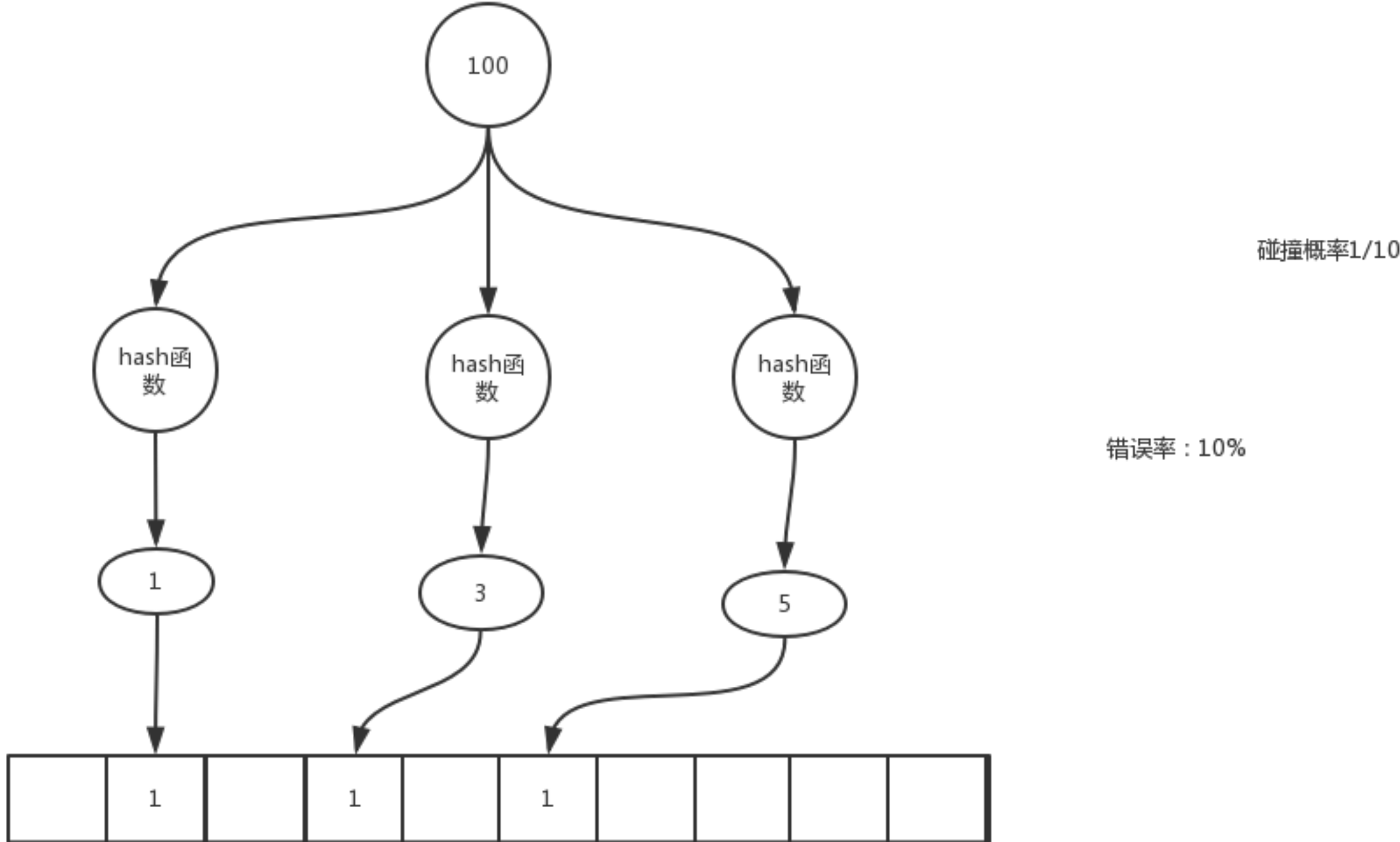


根据预估数据量n以及误判率fpp，bit数组大小的m的计算方式

$$m = - \frac{n \ln fpp}{(\ln 2)^2}$$

由预估数据量n以及bit数组长度m，可以得到一个hash函数的个数k

$$k = \frac{m}{n} \ln 2$$



场景3、大量的客户端拿相同的key去后台同时取数据。比如说爆款商品的数据

互斥锁

```
public String getWithLock(){
    String value = redis.get(key);
    if(value == null){
        Lock locked = tryDistributedLock()
        if(locked){
            value = getValueFromMySQL
            redis.set(value)
            unlock();
        }else{
            sleep(50)
            getWithLock ( )
        }
    }
}
```

同时

