

Java架构师必备技能之 高并发及性能调优系列课程训练营

讲师：JackHu

时间：2020年9月8日

个人简介：

开课吧特约资深讲师&课程研究员
知学科技首席架构师&创始人
博学谷课程研究&技术负责人
51拜师技术架构师&负责人
药房网技术负责人
华北计算研究所技术负责人
机械工业出版社《源码修炼指南》
《RocketMQ实战内幕》



章节内容



1.认识高并发的基本原理

2.服务部署&压力测试&性能参数分析

3.认识一些常用的并发指标

4.服务优化&压力测试性能对比

5.keepalive

6.Java服务常见线上问题定位方法

7.实现高并发的十二条经验分享

01 /

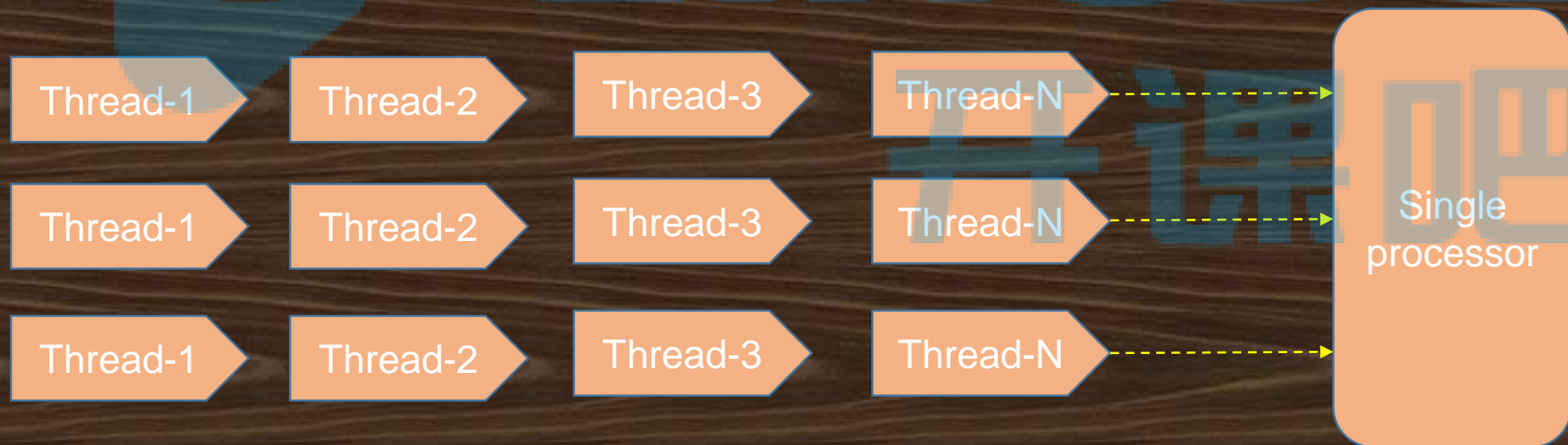
认识高并发的基本原理

kaikedaba

开课吧

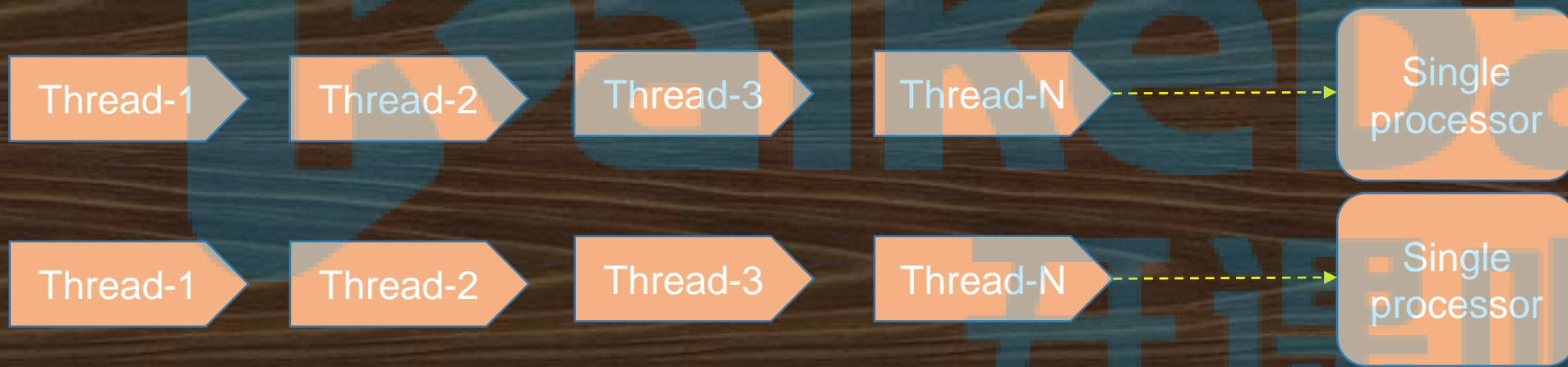
思考问题：什么是并发？

一个时间段中有几个程序都处于已启动运行到运行完毕之间，且这几个程序都是在同一个处理机上运行；因此并发是指在一段时间内宏观上多个程序同时运行



思考问题：什么是并行？

并行指的是同一个时刻，多个任务确实真的在同时运行



并发和并行的区别：

并发，指的是多个事情，在同一时间段内同时发生了。

并行，指的是多个事情，在同一时间点上同时发生了。

思考问题：你真的了解高并发吗？所谓的亿级流量，你真的懵逼了吗？

kaikeba
开课吧

案例：秒杀详情页-流量分析

核心业务流程：

- 1、商品详情页（重点说详情接口：一天 100w单）
- 2、下单
- 3、库存
- 4、支付

高峰时间段是：

11 - 14点， 17 - 23点， 一天24小时，差不多会有 11 h - 12 h 属于高峰期。

案例：秒杀详情页-内存预估分析(内存压力/s)

第三方工具：RamUsageEstimator 计算对象内存

```
@Override
public TbSeckillGoods findOne(Integer id){
    //直接从数据库查询
    TbSeckillGoods seckillGoods = seckillGoodsMapper.selectByPrimaryKey(id);
    //计算对象大小
    LOGGER.info("对象占用jvm堆内存大小: {}", RamUsageEstimator.sizeOfObject(seckillGoods));
    //返回结果
    return seckillGoods;
}
```

02 / 服务部署&压力测试&性能参数分析

kaikedaba
开课吧

- 1、服务打包部署
- 2、压力测试
- 3、测试参数解析

aikeba
开课吧

并发测试之并发设置

线程属性

线程数: 200

Ramp-Up时间 (秒): 1

循环次数 ☐ 永远 50

☒ Same user on each iteration

保证测试的请求是并发状态

线程数: $n=5$

循环次数: $a = 1000$

平均响应时间: $t = 0.2s$

Ramp-Up Period $T=10s$

$$S = (T - T/n) = 8$$

循环次数 * 平均响应时间

$$a * t > S \implies a > S/t = 40$$

平均响应时间 (ms)

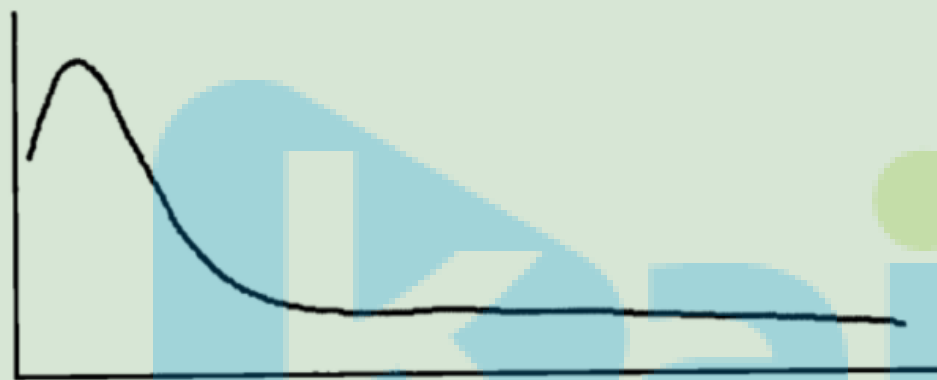


图 12-27 平均响应时长曲线 (1)

平均响应时间 (ms)

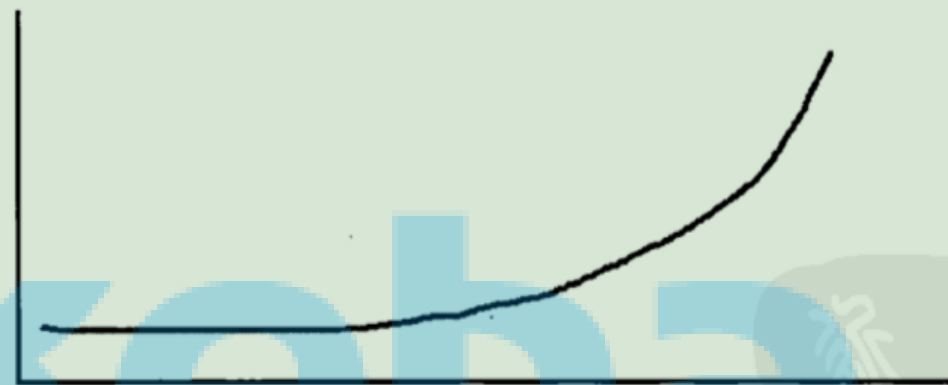


图 12-28 平均响应时长曲线 (2)

平均响应时间 (ms)

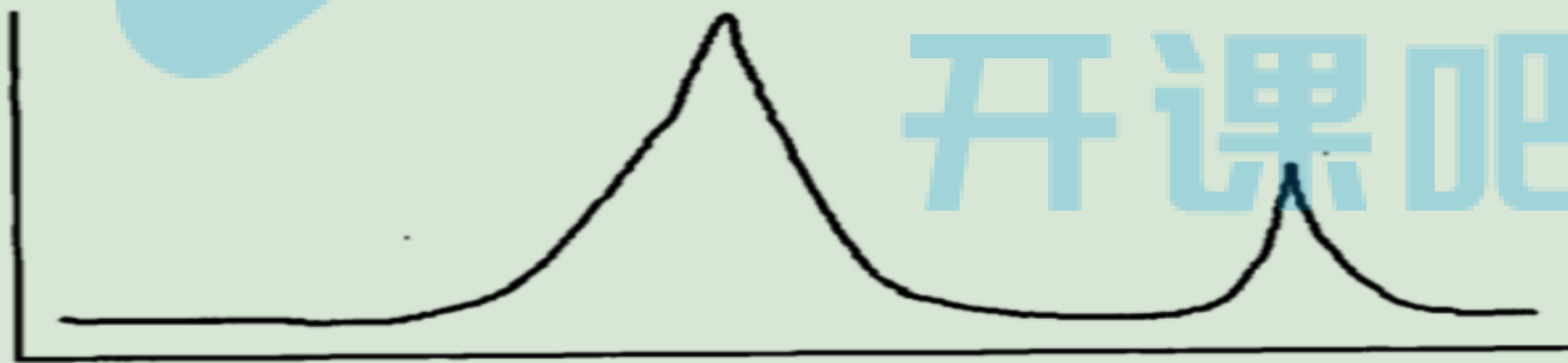


图 12-29 平均响应时长曲线 (3)

03 / 服务性能优化&压力测试

kaikeda
开课吧

1、服务优化：线程池优化

```
server:  
  port: 9000  
  tomcat:  
    uri-encoding: utf-8  
    max-threads: 800  
    accept-count: 1000  
    max-connections: 20000  
    min-spare-threads: 100
```

2、压力测试



1、keepalive

2、压力测试



1、undertow服务

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
    <exclusions>
      <exclusion>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-tomcat</artifactId>
      </exclusion>
    </exclusions>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-undertow</artifactId>
  </dependency>
</dependencies>
```

04 / 高并发性能指标分析

kaikeda
开课吧

- 1、QPS
- 2、TPS
- 3、RT
- 4、吞吐量



05 / 如何快速定位系统问题

kaikeda
开课吧

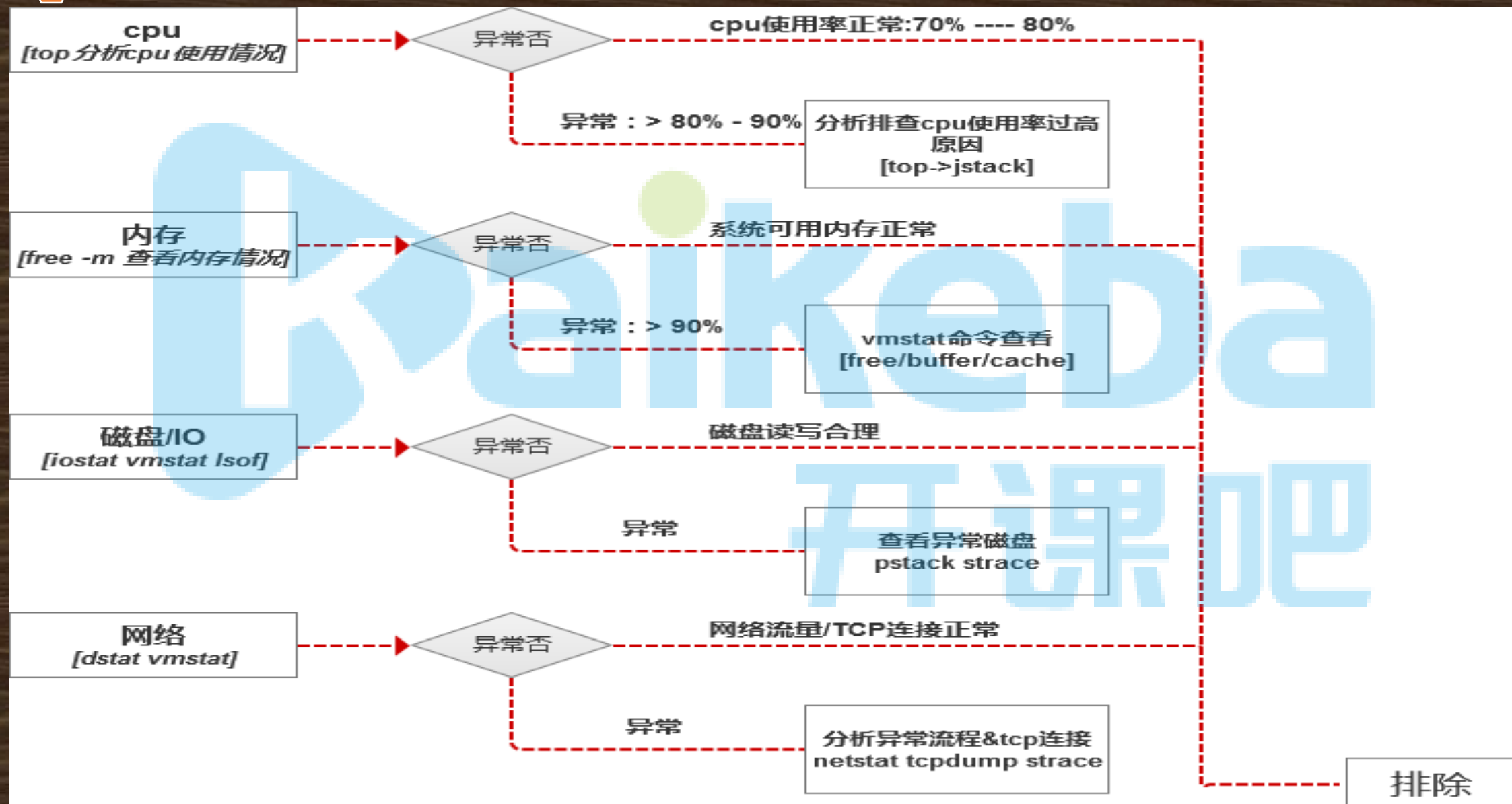
线上问题分类

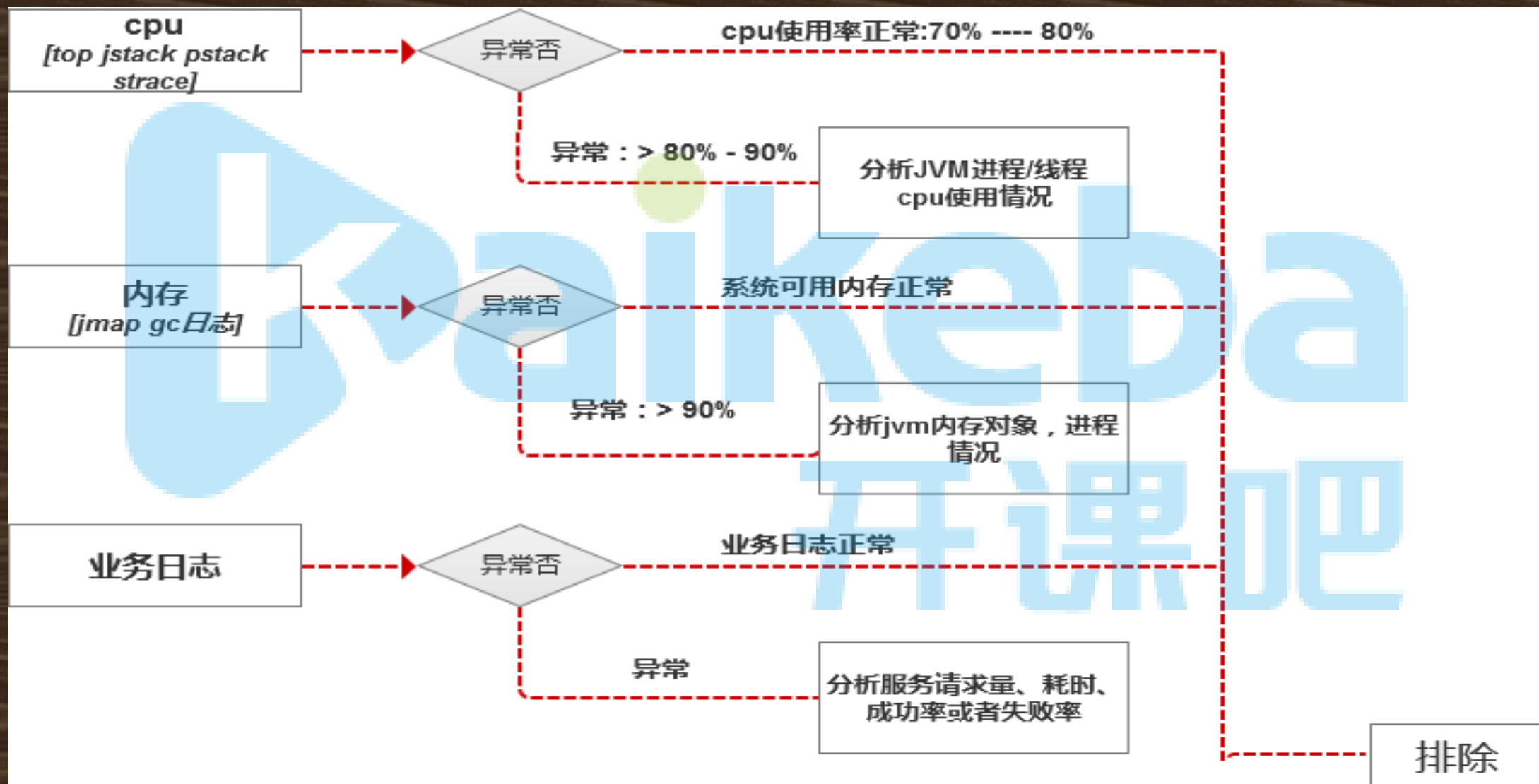
- 系统异常

- 业务异常

aikeba

开课吧





06/ 实现高并发的十二条经验分享

Kaikaiba

开课吧

- (1) 系统的架构设计，如何在架构层面减少不必要的处理（网络请求，数据库操作等）
- (2) 网络拓扑优化减少网络请求时间、如何设计拓扑结构，分布式如何实现？
- (3) 系统代码级别的代码优化，使用什么设计模式来进行工作？
- (4) 提高代码层面的运行效率、如何选取合适的数据结构进行数据存取？如何设计合适的算法？
- (5) 任务执行方式级别的同异步操作，在哪里使用同步，哪里使用异步？
- (6) JVM调优，如何设置Heap、Stack、Eden的大小，如何选择GC策略,控制Full GC的频率？

(7) 服务端调优 (线程池, 等待队列)

(8) 数据库优化减少查询修改时间。数据库的选取? 数据库引擎的选取? 数据库表结构的设计? 数据库索引、是否使用读写分离? 还是需要考虑使用数据仓库?

(9) 缓存数据库的使用, 如何选择缓存数据库? 是Redis还是Memcache? 如何设计缓存机制?

(10) 数据通信问题, 如何选择通信方式? 是使用TCP还是UDP, 是使用长连接还是短连接? NIO还是BIO? netty、mina还是原生socket?

(11) 操作系统选取, 是使用Linux? 或者Unix?

(12) 硬件配置? 是8G内存还是32G, 网卡10G还是1G?