Atitit 性能指标与性能提升的5个原则与性能提升模型

[1. 性能的几个指标主要是响应时间（Response time） 2．吞吐量（Throughput） 1](#_Toc26909)

[2. 性能提升的5个原则与性能提升模型 2](#_Toc28725)

[2.1. Kiss原则简化清理不必要的组件 2](#_Toc13512)

[2.2. 分类优先级，与分区域，减少要操作或提取的内容与范围 2](#_Toc8108)

[2.3. div分而治之 聚沙成塔 分布式 并发 2](#_Toc8858)

[2.4. 大力提升单体处理能力 2](#_Toc18096)

[2.5. 障眼法cache异步 提升快组件与慢组件的同步模式，防止快速组件等待慢速组件 2](#_Toc11449)

[2.6. Hybrid 混合综合模式 2](#_Toc14571)

[3. 性能问题的检测方法与工具概览 3](#_Toc31197)

[3.1. 问题程序的全局资源占用统计 Os资源占用统计比如win的资源管理器，与linux的top等工具 3](#_Toc25888)

[3.2. 性能检测工具 Profile工具 3](#_Toc15615)

[3.3. 自制profile工具 3](#_Toc15516)

[3.4. 人工检查法 3](#_Toc27526)

[3.5. 常见性能故障排查表，根据性能异常情况，按图索骥，解决方法 4](#_Toc19252)

[3.6. 性能检测流程，防止遗漏重要的流程节点 4](#_Toc17882)

[3.7. 对常用涉及性能的api，使用团队内部api类库包装代替通用api，固化性能算法 4](#_Toc11274)

[4. 常用的性能问题标准化检测流程最佳实践bp总结 4](#_Toc19840)

# 性能的几个指标主要是响应时间（Response time） 2．吞吐量（Throughput）

其次还有3．资源使用率（Resource utilization） 5．并发用户数（Concurrent users） 并发连接数等。

这里的性能调整主要以缩短响应时间为主 兼吞吐量，有时他们之间是互相矛盾的。要根据实际项目需求调整

# 性能提升的5个原则与性能提升模型

## Kiss原则简化清理不必要的组件

## 分类优先级，与分区域，减少要操作或提取的内容与范围

## div分而治之 聚沙成塔 分布式 并发

工作多，怎么办，招募更多的人手了。 短期内，分而治之，可能是常见的解决方法。就像盖金字塔，在当时的技术条件下，聚合众多人力可能是唯一的方法。

## 大力提升单体处理能力

从长远趋势的来看，提升单体处理能力是长久解决之道。。就像盖大楼，不再使用众多人力的模式了，而是使用大型工具了。

## 障眼法cache异步 提升快组件与慢组件的同步模式，防止快速组件等待慢速组件

提升快组件与慢组件的同步模式，防止快速组件等待慢速组件。。比如常见的倍频同步技术，异步技术 buffer 减少不必要的状态切换和设备I/O **懒加载**

Cache法 空间换时间大法 “凡是看得见的,都是暂时的,只有看不见的,才是永远的

所需要的结果已经预先计算好了。需要的时候，直接使用。。Cache是个很重要的性能提升模式。。

## Hybrid 混合综合模式

架构提升性能

# 性能问题的检测方法与工具概览

## 问题程序的全局资源占用统计 Os资源占用统计比如win的资源管理器，与linux的top等工具

查看 问题程序的资源占用统计监测工具，，统计资源占用情况

一般是统计出现性能问题程序的cpu占有率，线程数量，文件句柄数，内存使用率，磁盘与网络io使用率等指标。

## 性能检测工具 Profile工具

要解决问题，首先要找到问题。。工具可以大大提升准确度与检测效率。。

一般profile工具可以提供线程数量监测，内存对象占用，以及代码执行时间等监测功能。

## 自制profile工具

有时候，你使用的语言或数据库等没有提供自己的profile工具，网上也找不到第三方工具，与其他人沟通交流也得不到相关工具信息，就要自己动手，制作profile工具。。

一般调用time api，毫秒级，类似aop模式，计算每一个方法的执行时间，输出到日志或者数据库，输出到数据库更好，因为可以方便的按照执行时间排序，api调用层次等，优先调整占用时间长的代码片段方法。。

具体工具设计可以参照成熟的profile工具。。

## 人工检查法

优点是快速，很多代码片段一眼看过去就能看出所在问题。但需要一定的经验积累才可。。对技术要求较高。。其次，有些情况下，没有合适的工具或不能插入检测代码，只能人工检查。

## 常见性能故障排查表，根据性能异常情况，按图索骥，解决方法

来源于互联网的资源，以及公司内部的垂直领域的资源整理。。以及项目组与个人的资源积累。。

## 性能检测流程，防止遗漏重要的流程节点

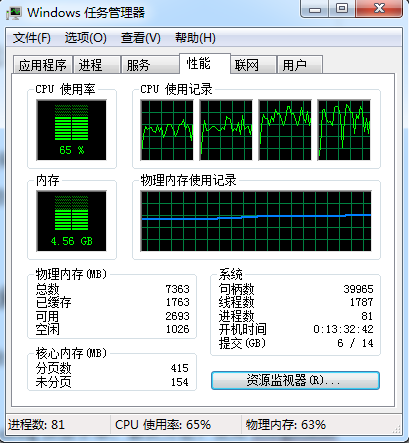
## 对常用涉及性能的api，使用团队内部api类库包装代替通用api，固化性能算法

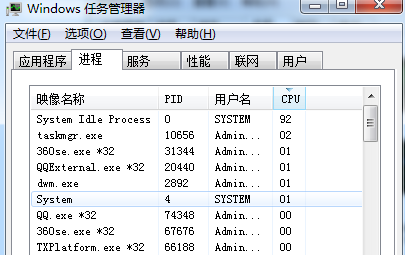
易于使用，提升性能同时简化开发

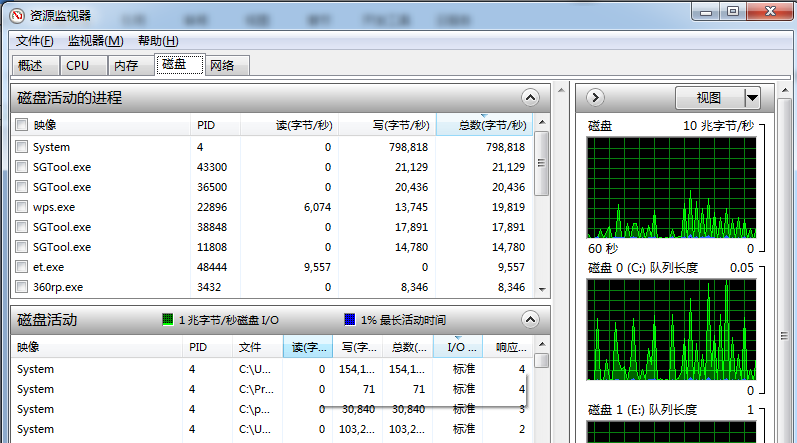
# 常用的性能问题标准化检测流程最佳实践bp总结

首先尽可能排除，os系统资源占用问题，可能是另外的程序占用了大量资源，导致本项目性能较慢。。这个一般使用win系统使用任务管理器即可。。Linux系统使用类似功能的top等性能检测工具。。。

主要检查四大资源占用，cpu，内存，磁盘io，网络io







如果确实是其他程序占用了大量资源，要么kill掉，要么就要移动到其他机器上部署了。。再次观察，如果可以了，则性能提升完成。否则就要继续下一步。

然后，使用相关语言具体领域的性能检测工具profile工具开始检测具体项目的资源占用。。

作者:: 绰号:老哇的爪子claw of Eagle 偶像破坏者Iconoclast image-smasher

捕鸟王"Bird Catcher  kok  虔诚者Pious 宗教信仰捍卫者 Defender Of the Faith. 卡拉卡拉红斗篷 Caracalla red cloak 万兽之王

简称：： Emir Attilax Akbar 埃米尔 阿提拉克斯 阿克巴

全名：：Emir Attilax Akbar bin Mahmud bin  attila bin Solomon bin adam Al Rapanui 埃米尔 阿提拉克斯 阿克巴 本 马哈茂德 本 阿提拉 本 所罗门 本亚当  阿尔 拉帕努伊

常用名：艾提拉（艾龙），  EMAIL:1466519819@qq.com

头衔：uke总部o2o负责人，全球网格化项目创始人，

uke交友协会会长  uke捕猎协会会长 Emir Uke部落首席大酋长，

uke宗教与文化融合事务部部长， uke宗教改革委员会副主席

uke制度与重大会议委员会委员长，uke保安部首席大队长,uke制度检查委员会副会长，

uke 首席cto   软件部门总监 技术部副总监  研发部门总监主管  产品部副经理 项目部副经理   uke科技研究院院长uke软件培训大师

uke波利尼西亚区大区连锁负责人 汤加王国区域负责人 uke克尔格伦群岛区连锁负责人，莱恩群岛区连锁负责人，uke布维岛和南乔治亚和南桑威奇群岛大区连锁负责人

 Uke软件标准化协会理事长理事长 Uke **[数据库](http://lib.csdn.net/base/14" \o "MySQL知识库" \t "http://blog.csdn.net/attilax/article/details/_blank)**与存储标准化协会副会长

uke终身教育学校副校长   Uke医院 与医学院方面的创始人

 uec学院校长， uecip图像处理机器视觉专业系主任   uke文档检索专业系主任

Uke图像处理与机器视觉学院首席院长

Uke 户外运动协会理事长  度假村首席大村长   uke出版社编辑总编

转载请注明来源：attilax的专栏  ?http://blog.csdn.net/attilax

--Atiend  v8