

dns负载均衡

最早的负载均衡技术是通过DNS来实现的，在DNS中为多个地址配置同一个名字，因而查询这个名字的客户机将得到其中一个地址，从而使得不同的客户访问不同的服务器，达到负载均衡的目的。

DNS负载均衡是一种简单而有效的方法，但是它不能区分服务器的差异，也不能反映服务器的当前运行状态。

DNS负载均衡技术的实现原理是在DNS服务器中为同一个主机名配置多个

IP地址，在应答DNS查询时，DNS服务器对每个查询将以DNS文件中主机记录的IP地址按顺序返回不同的解析结果，将客户端的访问引导到不同的机器上

去，使得不同的客户端访问不同的服务器，从而达到负载均衡的目的。

RR-

DNS方法只是一个简单的负载平衡方案，如果你有更高要求，可以研究LVS集群（IPVS和KTCVPS、TCPHA），实现基于IP的负载均衡和基于内容的负载均衡。反向负载代理均衡

灰度发布

灰度发布是指在黑与白之间，能够平滑过渡的一种发布方式。AB

test就是一种灰度发布方式，让一部分用户继续用A，一部分用户开始用B，如果用户对B没有什么反对意见，那么逐步扩大范围，把所有用户都迁移到B上面

来。灰度发布可以保证整体系统的稳定，在初始灰度的时候就可以发现、调整问题，以保证其影响度。

过载保护

每个系统，自己的最大处理能力是多少要做到清清楚楚。

每个系统要做好自我保护，量力而为，而不是尽力而为。对于超出自己处理能力范围的请求，要勇于拒绝。

每个系统要有能力发现哪些是有效的请求，哪些是无效的请求。

前端系统有保护后端系统的义务，每一个前后端接口的地方，都有明确的负载约定，一环扣一环。

当过载发生时，该拒绝的请求（1、超出整个系统处理能力范围的；2、已经超时的无效请求）越早拒绝越好。

对于用户的重试行为，要适当的延缓。例如登录发现后端响应失败，再重新展现登录页面前，可以适当延时几秒钟，并展现进度条等友好界面。当多次重试还失败的情况下，要安抚用户。

set模型

有效防止故障扩散、单点失效

利用SET思想将一个服务按用户范围分成不同的小单元，每个小单元(SET)具备全部业务服务能力，当一个SET发生故障时候，只影响这一小部分用户从实现到有效防止故障扩散。在带有容错机制的SET架构中，在

SET之间增加容错处理，当一个SET发生故障时，使用类似一致性哈希的算法，调用方可以自动切换到下一个SET来存储，并且将新的位置记录在index上以实现高效容错。

实现快慢分离、读写分离

使用SET模型可以方便的管理服务资源，有效实现服务的快慢分离，避免轻量级服务与重量级服务需求杂糅请求到同一服务点，高效避免了请求拥塞和拒绝服务的发生。

使用SET模型还可以做到读写分离，对DB或Redis的读写实施分离，以实现读写互不干扰、提高海量读取服务速度与质量的目标。在带有容错机制的SET架构中，在SET之间增加容错处理，当一个SET发生故障时，使用类似一致性哈希的算法，调用方可以自动切换到下一个SET来存储，并且将新的位置记录在index上以实现高效容错。