1. LB三种架构

**VS/DR**利用大多数Internet服务的非对称特点（上传与下载速度不同），负载调度器只负责调度请求，而服务器直接将响应返回给客户，可以极大地提高整个集群系统的吞吐量。（吞吐量是指在单位时间内[中央处理器](https://baike.baidu.com/item/%E4%B8%AD%E5%A4%AE%E5%A4%84%E7%90%86%E5%99%A8)（CPU）从存储设备读取->处理->存储信息的量。）在DR模式中，调度器根据各个真实服务器的负载情况，连接数多少等，动态地选择一台服务器，不修改目标IP地址和目标端口，也不封装IP报文，而是将请求报文的数据帧的目标MAC地址改为真实服务器的MAC地址。然后再将修改的数据帧在服务器组的局域网上发送。因为数据帧的MAC地址是真实服务器的MAC地址，并且又在同一个局域网。那么根据局域网的通讯原理，真实服务器是一定能够收到由LB发出的数据包。真实服务器接收到请求数据包的时候，解开IP包头查看到的目标IP是VIP。（此时只有自己的IP符合目标IP才会接收进来，所以我们需要在本地的回环借口上面配置VIP。PS：在环回口上配IP而不在网卡上配是通过BGP引流的话，Loopback口只要Router还健在，则它就会一直保持Active，这样，只要BGP的Peer的Loopback口之间满足路由可达，就可以建立BGP回话，总之BGP中使用loopback口可以提高网络的健壮性。另：由于网络接口都会进行ARP广播响应，但集群的其他机器都有这个VIP的lo接口，都响应就会冲突。所以我们需要把真实服务器的lo接口的ARP响应关闭掉。）然后真实服务器做成请求响应，之后根据自己的路由信息将这个响应数据包发送回给客户，并且源IP地址还是VIP。lvs基本上能支持所有应用，因为lvs工作在4层，所以它可以对几乎所有应用做负载均衡，包括http、数据库、聊天室等等。抗负载能力强。抗负载能力强、性能高，能达到F5硬件的60%；对内存和cpu资源消耗比较低，工作在网络4层，通过vrrp协议转发（仅作分发之用），具体的流量由linux内核处理，因此没有流量的产生。咱们还通过DPDK绕过内核，稳定性、可靠性好，自身有完美的热备方案；（如：LVS+Keepalived）配置 复杂，对网络依赖比较大。

**HAPROXY**

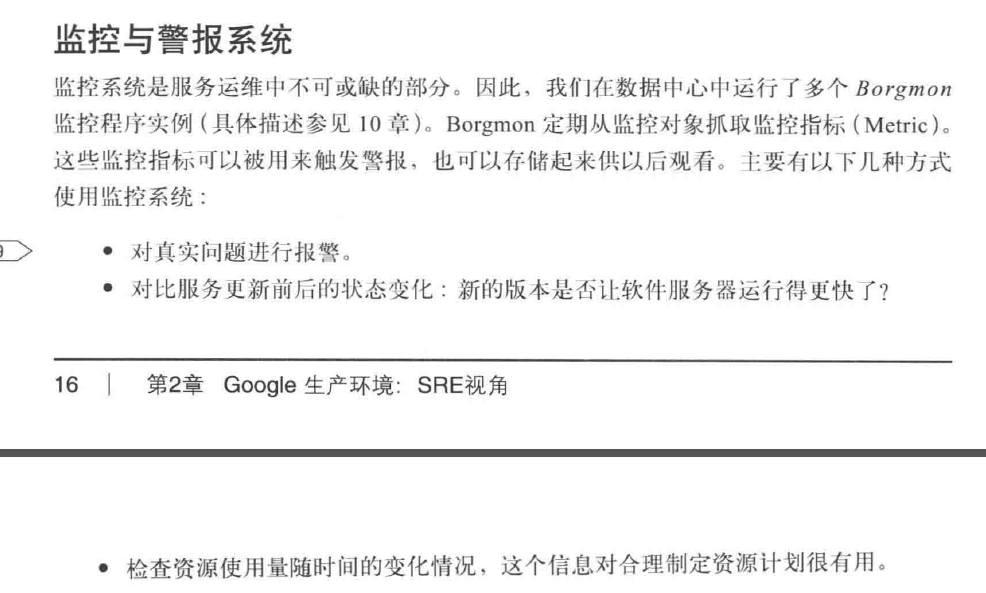
Nginx为支持长连接，需要配置使用upstream 模块的keepalive选项，根据QPS和响应时间计算出需要的长连接量，HaProxy能够补充Nginx的一些缺点比如Session的保持，Cookie的引导等工作。haproxy 将后端服务器产生的session和后端服务器标识存在haproxy中的一张表里。客户端请求时先查询这张表。

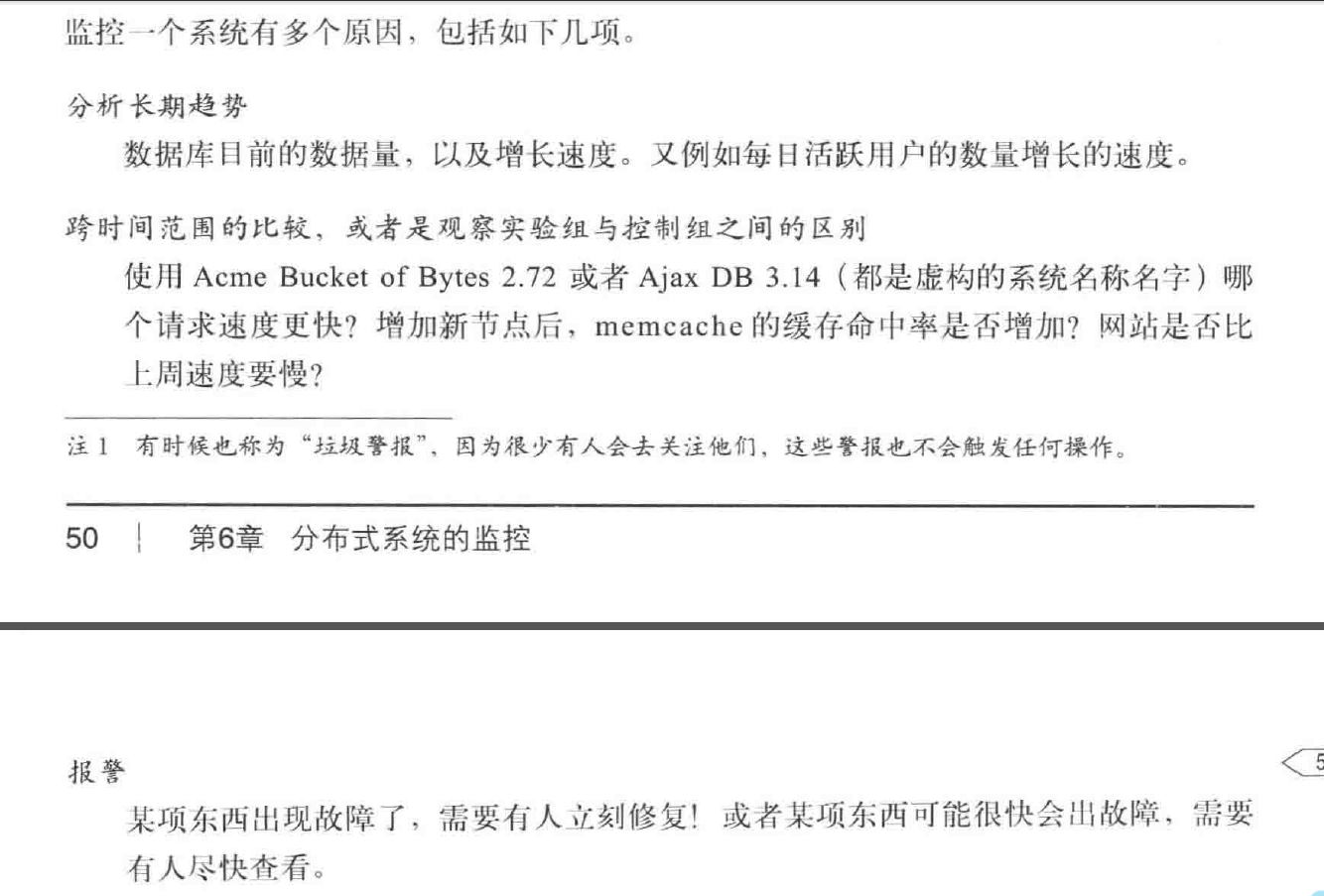
疑问：当客户通过TCP连接访问网络访问时，服务所需的时间和所要消耗的计算资源是千差万别的，它依赖于很多因素。例如，它依赖于请求的服务类型、当前网 络带宽的情况、以及当前服务器资源利用的情况。一些负载比较重的请求需要进行计算密集的查询、数据库访问、很长响应数据流；而负载比较轻的请求往往只需要 读一个HTML页面或者进行很简单的计算。

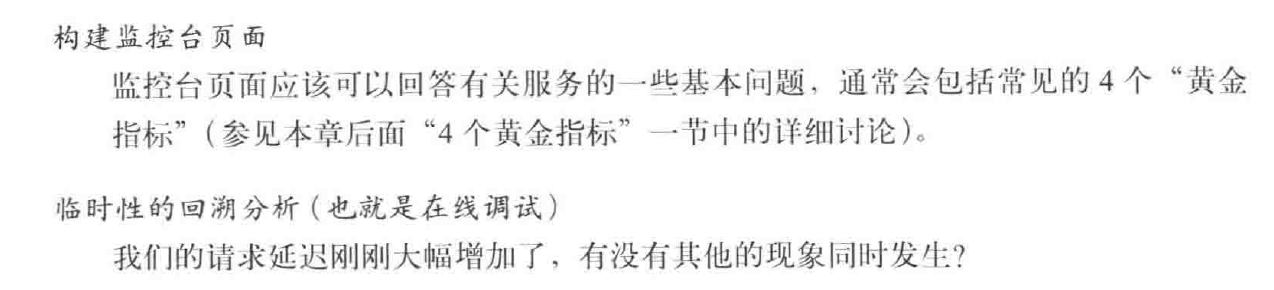
请求处理时间的千差万别可能会导致服务器利用的倾斜（Skew），即服务器间的负载不平衡。例如，有一个WEB页面有A、B、C和D文件，其中D是 大图像文件，浏览器需要建立四个连接来取这些文件。当多个用户通过浏览器同时访问该页面时，最极端的情况是所有D文件的请求被发到同一台服务器。所以说， 有可能存在这样情况，有些服务器已经超负荷运行，而其他服务器基本是闲置着。同时，有些服务器已经忙不过来，有很长的请求队列，还不断地收到新的请求。反 过来说，这会导致客户长时间的等待，觉得系统的服务质量差。

简单连接调度可能会使得服务器倾斜的发生。在上面的例子中，若采用轮叫调度算法，且集群中正好有四台服务器，必有一台服务器总是收到D文件的请求。这种调度策略会导致整个系统资源的低利用率，因为有些资源被用尽导致客户的长时间等待，而其他资源空闲着。

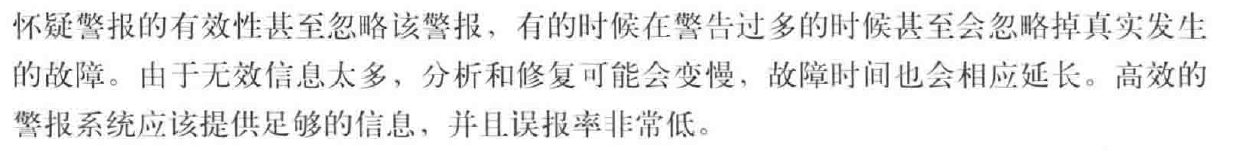
监控

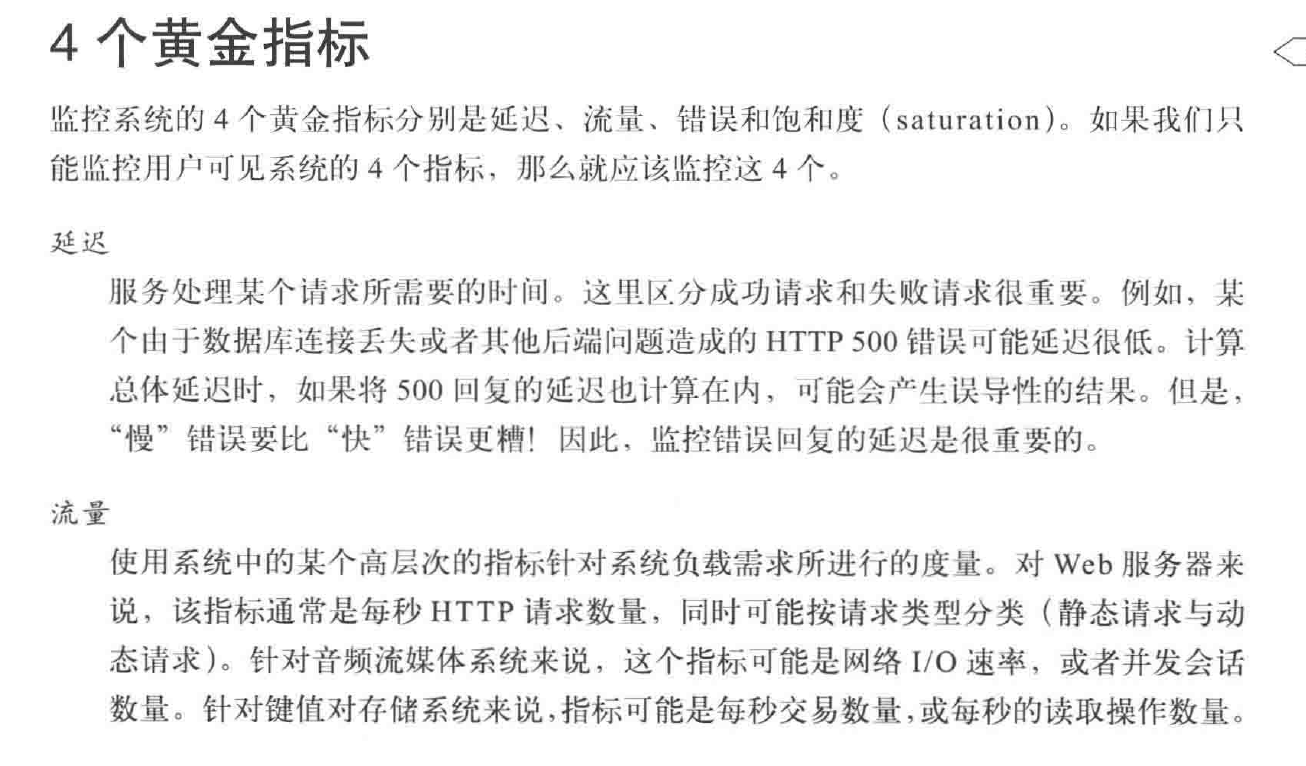


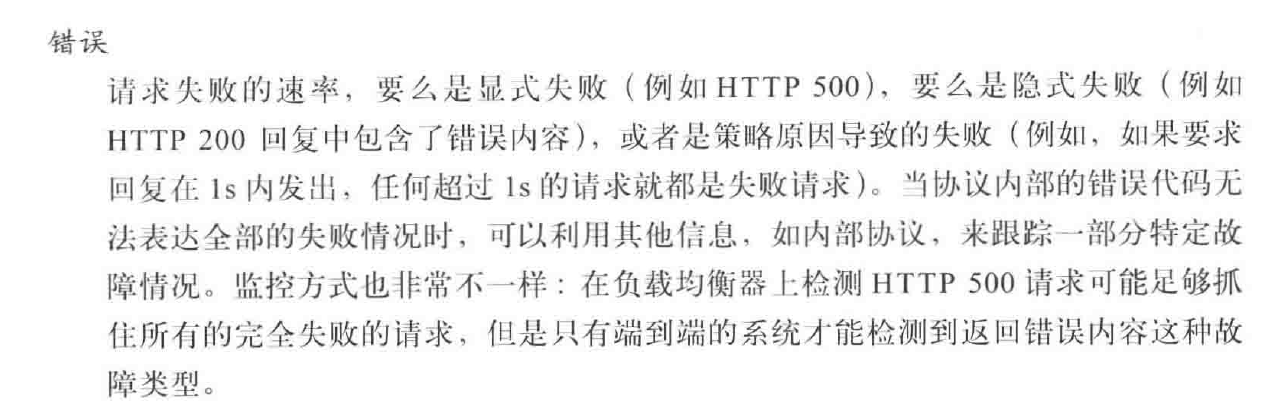


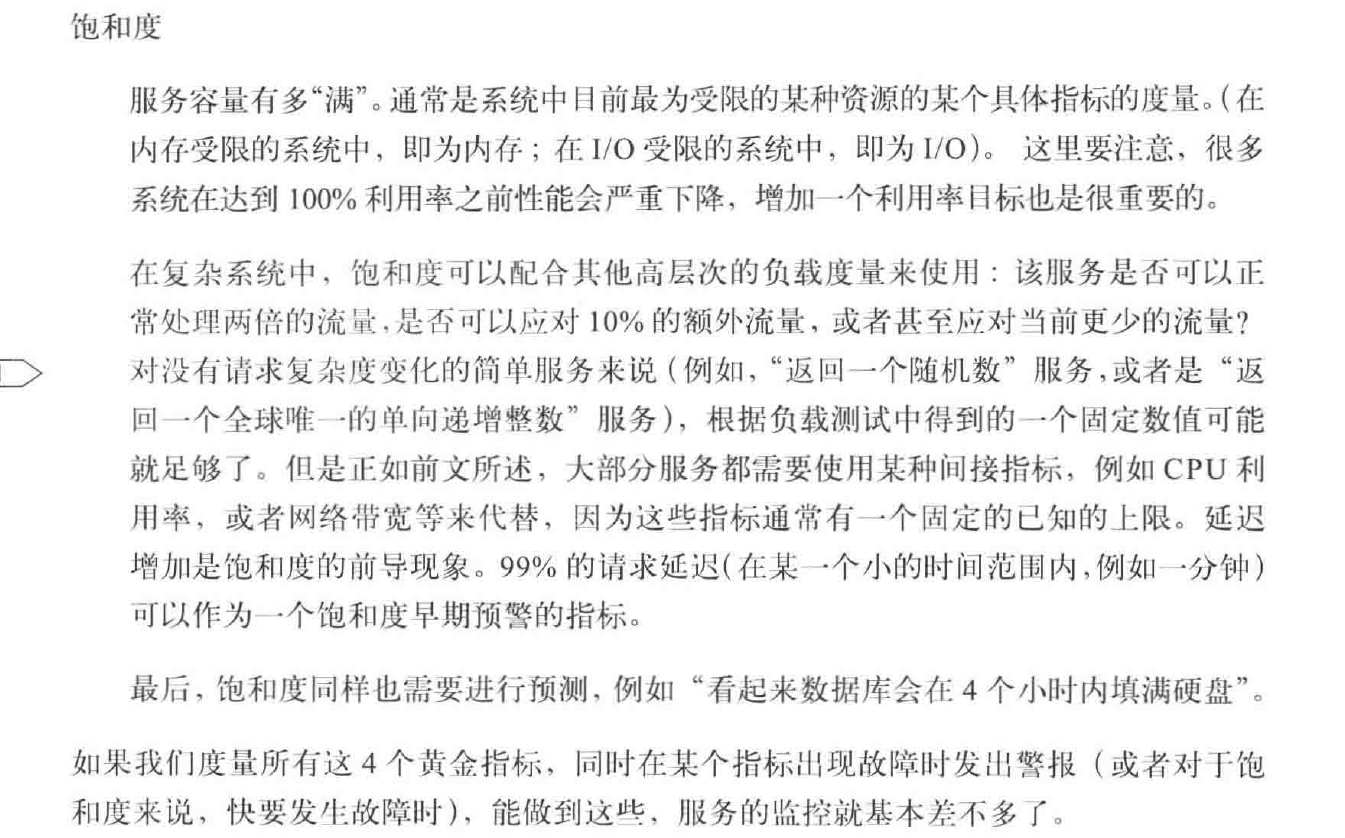


思考：









三、SRE



