不同服务之间负载:

大服务组请求到小服务组要注意会不会有把服务压挂的可能性.

Ral调用：

Mcpack/http

Ral/ral\_multi/ral\_reate\_loop()

根据参数请求不同的服务组，

可以配置几种load\_balance策略，一般为random

Nmq

生产者

Nmqproxy

Topic

Pusher

消费者

拆分：

假设我们有一台服务器，它可以承担1百万/秒的请求，这个请求可以的是通过http访问网页，通过tcp下载文件，jdbc执行sql，RPC调用接口…，现在我们有一条数据的请求是2百万/秒，很显然服务器hold不住了，会各种拒绝访问，甚至崩溃，宕机，怎么办呢。一台机器解决不了的问题，那就两台。所以我们加一台机器，每台承担1百万。如果请求继续增加呢，两台解决不了的问题，那就三台呗。这种方式我们称之为水平扩展。如何实现请求的平均分配便是负载均衡了。 另一个栗子，我们现在有两个数据请求，数据1 90万，数据2 80万，上面那台机器也hold不住，我们加一台机器来负载均衡一下，每台机器处理45万数据1和40万数据2，但是平分太麻烦，不如一台处理数据1，一台处理数据2，同样能解决问题，这种方式我们称之为垂直拆分。