**请求方式**

新的方式是：

1.浏览器发送请求

2.直接到达html页面（前端控制路由与渲染页面，整个项目开发的权重前移）

3.html页面负责调用服务端接口产生数据（通过ajax等等，后台返回json格式数据，json数据格式因为简洁高效而取代xml）

4.填充html，展现动态效果，在页面上进行解析并操作DOM。

（有兴趣的童鞋可以访问一下阿里巴巴等大型网站，然后按一下F12，监控一下你刷新一次页面，他的http是怎么玩的，大多数都是单独请求后台数据，

使用json传输数据，而不是一个大而全的http请求把整个页面包括动+静全部返回过来）

总结一下新的方式的请求步骤：

大量并发浏览器请求--->web服务器集群(nginx)--->应用服务器集群(tomcat)--->文件/数据库/缓存/消息队列服务器集群，同时又可以玩分模块，还可以按业务拆成一个个的小集群，为后面的架构升级做准备。

**集群**

在多个节点间自动拆分你的数据集的能力。

当部分节点正在经历失败或者不能与集群其他节点通信时继续运转的能力。

**Redis 集群的 TCP 端口**

为了让 Redis 集群工作正常，对每个节点：

用于与客户端通信的正常的客户端通信端口（通常为 6379）需要开放给所有需要连接集群的客户端以及其他集群节点（使用客户端端口来进行键迁移）。

集群总线端口（客户端端口加 10000）必须从所有的其他集群节点可达。

**Redis 集群的数据分片**

Redis 集群有 16384 个哈希槽每一个 Redis 集群中的节点都承担一个哈希槽的子集

**添加新节点（Adding a new node）**

添加一个新节点的过程基本上就是，添加一个空节点，然后，如果是作为主节点则移动一些数据进去，如果是从节点则其作为某个节点的副本。

两种情况我们都会讨论，先从添加一个新的主服务器实例开始。

两种情况下，第一步要完成的都是添加一个空节点。

我们使用与其他节点相同的配置（端口号除外）在 7006 端口（我们已存在的 6 个节点已经使用了从 7000 到 7005 的端口）上开启一个新的节点，那么为了与我们之前的节点布局一致，你得这么做：

在你的终端程序中开启一个新的标签窗口。

进入 cluster-test 目录。

创建一个名为 7006 的目录。

在里面创建一个 redis.conf 的文件，类似于其它节点使用的文件，但是使用 7006 作为端口号。

最后使用../redis-server ./redis.conf 启动服务器。

此时服务器已经在运行中了。

现在我们可以像通常一样使用 redis-trib 来添加节点到已存在的集群中。

./redis-trib.rb add-node 127.0.0.1:7006 127.0.0.1:7000