# 系统设计面试指北

系统设计的问题是在国内外大厂面试的常考题，这类问题开放性比较强，没有标准答案，比如设计微博，设计微信，或者是设计某个特定功能，如限制用户的访问频率。

面试官的问题可能很大，但是让你设计的东西未必会很多，设计的难度也未必会很大，极有可能从易到难引导你先设计一些简单的结构。所以掌握这类问题的面试技巧很重要。

如何应对这类问题，我在一个教程上学习到了4S法，总结如下：

4S分析法中的4S是指Scenario（场景），Service（服务），Storage（存储），Scale（扩展）。

### 第一步：Scenario 场景

在这一步，你需要询问面试官：需要设计哪些功能（也可以自己想），需要承受多大的访问量。

就以设计微博为例，可以把微博的功能一一列出来，但显然无法在短短的一小时的面试里完成所有的功能设计，所以要筛选出核心的功能，比如发微博和浏览微博。

另外，需要考虑系统所承受的QPS（每秒查询次数）大概是多少，需要考虑并发用户，读写频率并掌握相关的计算方法，对一些经验性的数值要记得。

比如一台web服务器的承受量约为1k的QPS，一台关系型数据库的承受量约为1k的QPS，一台非关系型数据库的承受量是10k的QPS。

### 第二步：Service 服务

所谓服务可以理解为逻辑处理的整合，将同一类问题的逻辑处理处理的整合，对于同一类的问题的逻辑处理可以归并到一个服务。这一步实际上就是将整个系统细分为若干个小的服务。

比如以设计微博为例，我们可以拆分成用户服务，微博服务，关注关系服务，媒体资源服务等等。

### 第三步：Storage 存储

这一步是4S分析法中最重要的一部分，需要根据每个服务的数据特性选择合适的存储结构，然后细化数据表结构。

系统设计中可以选择的存储结构一般有三大类：数据库系统，文件系统，缓存系统。其中数据库又分为关系型数据库（例如MySQL）和非关系型数据库（NoSQL）。

以设计微博为例，根据前面拆分的服务的特点，用户服务适合用MySQL存储，而微博信息适合用文档型数据库MongoDB存储，多媒体资源可以借助文件系统，如AWS S3，对于热点数据，还可以用Redis做缓存。

另外，我们要向面试官展示数据存储和读取过程。就以微博的feed流为例，如何拿到新鲜事列表，可以采取pull和push两种方式，pull的方式实时的去获取关注人的微博，并做多路归并，push的方式会为每个用户维护自己的新鲜事的记录。

我们要在方案的取舍中体现出自己的专业性，以及tradeoff的能力。

### 第四步：Scale 扩展

经过前面3个步骤的分析，我们已经得到了一个解决方案，但这个方案还有很多的缺陷，所以需要4S分析法最后一步，扩展。

这一步分为两个部分，一个是优化，包括解决涉及缺陷，更多功能设计以及一些特殊情况如何处理。另一个是维护，包括系统的鲁棒性和扩展性，比如有一台服务器挂了怎么办，比如鹿晗公布恋情导致流量激增怎么办。

### 总结

最后总结一下系统设计面试过程中的注意点： - Ask before design. 先跟面试官明确需求再动手设计，不要一上来就冲着一个巨牛的方案去设计。 - No more no less. 不要总想着去设计最牛的系统，而是设计够用的系统。 - Work solution first. 先设计一个基本能工作的MVP产品，再逐步优化。 - Analysis is import than soluton. 系统设计没有标准答案，记住答案是没用的。通过分析过程展示你的知识储备，并权衡各种设计方式的利弊。