

# 如何快速上手一个新项目？

今天的文章标题就是我平时被问过的一个高频问题。

确实，很多小伙伴在学习或者接手一个项目的时候，不知道如何快速了解项目。

今天这篇文章我就简单聊聊“如何快速上手一个新项目？面试被问项目经历有哪些小技巧？”。

下面是正文！

## 项目学习五步走

**一般项目都会有遗留文档，不论是传统的项目开发模式还是敏捷开发模式。上项目之前自己抽时间看一下相关文档，大概了解一下这个项目整体的情况比如基本的业务还有技术选型啊这些。**

如果项目是单机的话，大部分就是增删改查的逻辑，主要是对于业务的理解。

如果项目是分布式或者微服务的话，会涉及各个服务之间的调用以及一些其他问题比如限流、分布式锁、分布式 ID 这些，稍微会复杂一些。

不过不论是什么类型的项目，上手的姿势大概是下面几步。

### 第一步:了解业务

先搞清楚你接受的新项目：

1. **是做什么的？** 主要面向什么人群使用？
2. **主要提供了哪些功能？**
3. 项目背景是什么样的？
4. 项目涉及的关键业务流程是怎么样？
5. 项目目前面临的挑战是什么？未来规划是什么？
6. ....

**技术本身就是为了业务而服务，只有首先搞清楚了业务之后你才真正算是步入了这个项目的门户。**

### 第二步:搭建项目开发环境

是骡子是马总要拉出来溜溜。所以，第二步我推荐你简单把开发环境搭建一下。搭建的步骤一般都在项目的 README 文档里面。

搭建完成之后，需要确保项目能够在自己的电脑上正确运行。

### 第三步:看项目技术架构

这个直接看项目的相关依赖就好。拿 Java 后端项目举例子，如果是 Maven 项目的话看 pom.xml，如果是 Gradle 项目就看 build.gradle。

可能会涉及下面这几部分，但是并不完全。

1. 项目最底层框架是什么？是 Spring 还是 Spring Boot，又或者是其他框架呢？
2. 项目依赖了哪些相关的包？挑重点看，比如数据库是 MyBatis 还是 JPA 或者是公司自研的框架呢？
3. 项目使用的什么数据库？是 MySQL 还是 PostgreSQL，又或者是其他数据库呢？
4. 项目用到了缓存吗？是 Redis 缓存吗?有没有用到本地缓存呢？
5. 项目用到了消息队列吗？Kafka 还是 RocketMQ？
6. 项目的权限管理这块是怎么做的呢？
7. ....

### 第四步:看项目的代码结构

项目的代码结构是怎么划分的，比如常见的项目可能会分为下面三层（复杂的系统分层可能会更多）。

1. Repository（数据库操作）
2. Service（业务操作）
3. Controller（数据交互）

如果是 DDD 分层架构的话，可能是下面这样的：

1. User Interface（用户界面层）
2. Application（应用层）
3. Domain（模型层）
4. Infrastructure（基础实施层）

1. 【推荐】图中默认上层依赖于下层，箭头关系表示可直接依赖，如：开放接口层可以依赖于 Web 层，也可以直接依赖于 Service 层，依此类推：



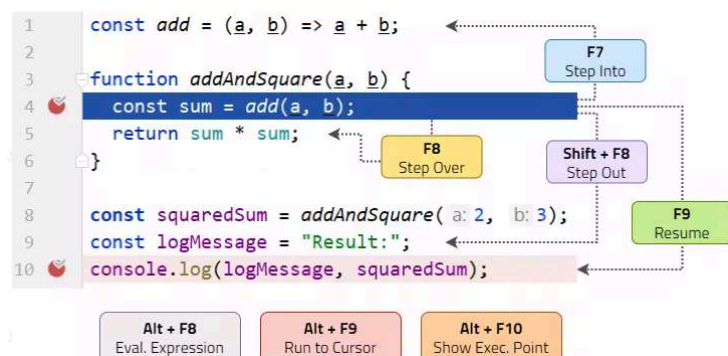
- 开放接口层：可直接封装 Service 方法暴露成 RPC 接口；通过 Web 封装成 http 接口；网关控制层等。
- 终端显示层：各个端的模板渲染并执行显示的层。当前主要是 velocity 渲染，JS 渲染，JSP 渲染，移动端展示等。
- Web 层：主要是对访问控制进行转发，各类基本参数校验，或者不复用的业务简单处理等。
- Service 层：相对具体的业务逻辑服务层。
- Manager 层：通用业务处理层，它有如下特征：
  - 1) 对第三方平台封装的层，预处理返回结果及转化异常信息。
  - 2) 对 Service 层通用能力的下沉，如缓存方案、中间件通用处理。
  - 3) 与 DAO 层交互，对多个 DAO 的组合复用。
- DAO 层：数据访问层，与底层 MySQL、Oracle、Hbase、OB 等进行数据交互。
- 外部接口或第三方平台：包括其它部门 RPC 开放接口，基础平台，其它公司的 HTTP 接口。

## 第五步:从功能主线/问题出发研究项目源码

一个比较成熟的项目的源码量是非常多，我们不可能都完完整整地看完，也没有必要。

你可以通过 debug 调试研究项目核心代码逻辑。比较推荐的方式就是通过一个功能主线（比如 Dubbo 是如何暴露服务的？）或者问题（比如？）出发。

对于企业项目来说，大部分还是知道如何进行 debug 调试的。但是，对于 Spring 这种顶级开源框架来说，很多人就不知道怎么打断点了。



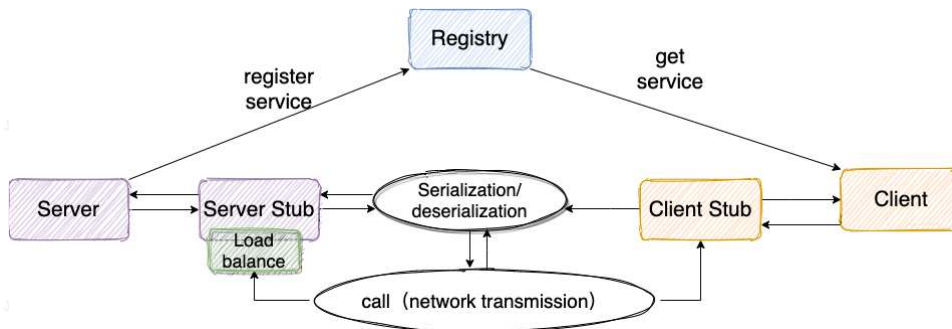
来研究整个过程。如果你觉得这种方法比较难的话，你也可以先去网上看看别人的分析。

## 项目经历的四个小技巧

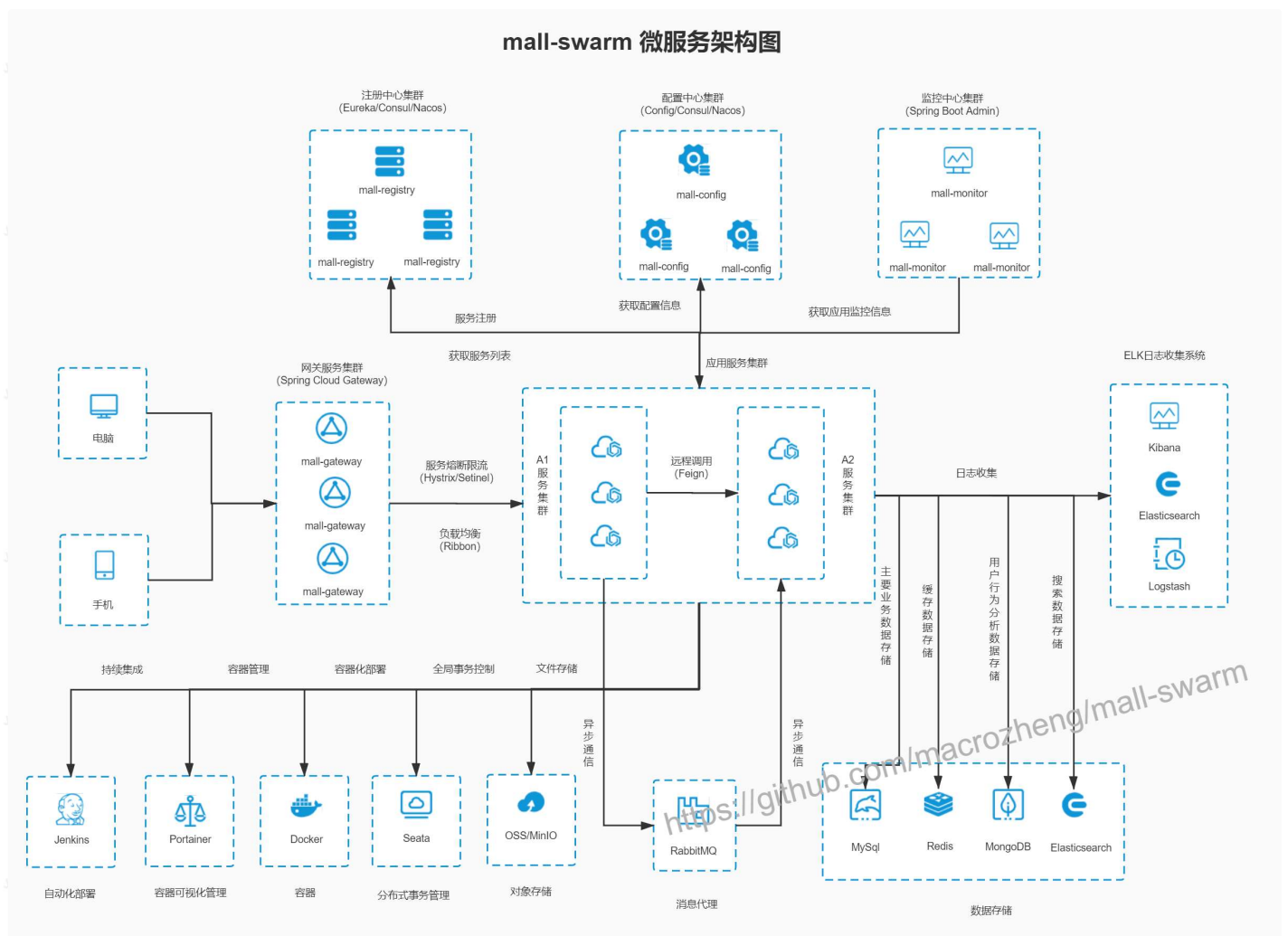
面试中，对于项目经历的考察是重中之重。下面我就分享 4 个面试被问项目经历的小技巧：

1.提前搞清楚项目的架构图、技术选型等等。

比如下面这个就是我之前写的一个简易 RPC 框架 ([guide-rpc-framework <https://github.com/Snailclimb/guide-rpc-framework>](https://github.com/Snailclimb/guide-rpc-framework)) 的架构图。

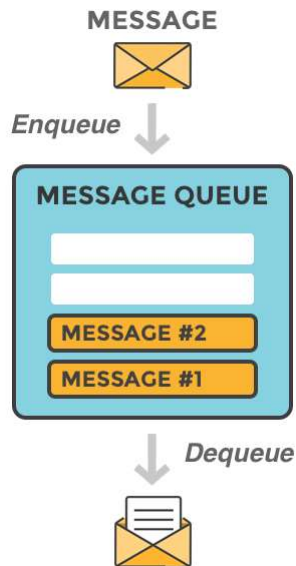


再比如下面这个是一个微服务的电商网站的架构图。



**2.提前想好项目的亮点，针对项目涉及的关键技术进行深度复习。**

比如说，你的项目用了消息队列的话，你就很有必要提前想好怎么回答消息队列相关的一些问题：消息队列解决了什么问题、常见消息队列对比、如何保证消息只被消费一次、如何保证消息不被重复消费……。



### 3.引导面试官问你熟悉的技术。

比如说，你对消息队列比较了解的话，介绍项目的时候就可以多介绍一下自己通过消息队列解决了什么问题。

### 4.突出个人的贡献比如自己在项目中解决了什么问题，而不只是叙述自己做了什么。

## 文章推荐

- [如何阅读代码 - ThoughtWorks洞见](https://insights.thoughtworks.cn/how-to-read-code/) - 2022 <https://insights.thoughtworks.cn/how-to-read-code/>：我司的一位大佬写的，内容很不错，分享了一些他日常读代码的一点经验比如先通过文档了解全景，再逐步深入代码。
- [快速阅读一个SpringBoot工程 - 掘金](https://juejin.cn/post/7250513276236808229) - 2023 <https://juejin.cn/post/7250513276236808229>：这篇文章介绍了一个常规的 SpringBoot 项目应该如何上手阅读，这方面基础较差的同学可以看看。
- [如何吃透一个 Java 项目？（附学习实践） - 淘系技术](https://mp.weixin.qq.com/s/1VVu25bzea1jzAWz7fKEUA) - 2021 <https://mp.weixin.qq.com/s/1VVu25bzea1jzAWz7fKEUA>：介绍了很多吃透 Java 项目的使用建议。
- [你可以 3 分钟熟悉一个系统吗？ - 阿里中间件](https://mp.weixin.qq.com/s/odjcRpVrDDghWFuISVOK5w) - 2020 <https://mp.weixin.qq.com/s/odjcRpVrDDghWFuISVOK5w>：从三个方面进行总结，提供一个系统的方法帮助快速上手一个完全陌生的系统，同时也可以用来 review 已有的系统，查漏补缺。