

# 专精 or 博学，多少人输 在了技术选择上？

四火 @ Oracle

# 我是谁？

- 程序员，热爱工程师文化
- @Huawei -> @Amazon -> @Oracle
- @南京 -> @北京 -> @Seattle
- 专栏 [《全栈工程师修炼指南》](#) 作者
- 长期技术博客 [《四火的唠叨》](#) 作者

我应该专精于发展一个具体领域，还是应该多领域尝试，  
通学广识？

这个问题没有确定的解，正面和反面例子都举不胜举。

我们讲，就要讲大多数人，以及普通人的情况，并且考虑到家庭、生活、性格等等种种的限制，寻求一个对自己职业生涯最优的解。

# 提纲

- 第一部分：技术路线的选择
- 第二部分：认识全栈工程师

# 第一部分：技术路线的选择

- 谈谈一个误区
- 我们该怎样选择技术路线？
  - 面临的困难
  - 技术栈分级
  - 学习的主次
  - 找到竞争力

# 误区：我应当在职业生涯早期就尽快确定自己要重点发展的技术领域

- 关于职业生涯规划
- “道理我都懂，可还是过不好这一生”，做出理性的决策
- 理想的长期的方向：
  - 一要自己感兴趣
  - 二要自己擅长做
  - 三要领域有前景
- 为什么无法在早期就确定技术路线？
  - 它会局限自己未来的选择
  - 视野需要时间的积累
  - 机遇往往是难以预测的

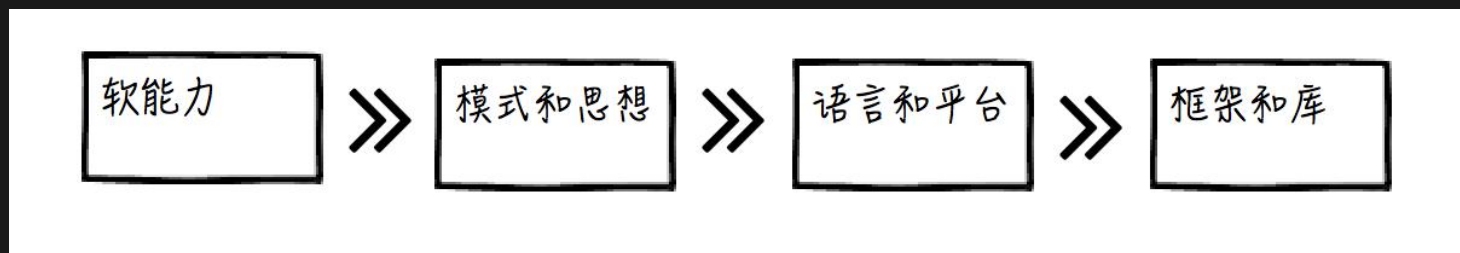
# 技术路线选择：面临的困难

- “迷茫”——技术选择的困惑
- 迷茫不代表等待，不确定时期的策略
- 技术路线的选择
  - 一个开发和测试转型的故事



# 技术路线选择：技术栈分级

- 技术的分级
  - 这是一个完整的解决问题的技术栈
  - 从左 -> 右大致的规律（大体分布，并不严格）：
    - 稳定性：高 -> 低
    - 寿命：长 -> 短
    - 技术积累难度：高 -> 低
    - 问题针对性：弱 -> 强
- 相互之间的关系
  - 解决实际问题缺一不可
  - 相互促进的关系
  - 从右向左的泛化：“玩法”和“套路” + “体系”



# 技术路线选择：学习的主次

- 每个人都有自己的局限性
  - 时间局限
  - 精力局限
  - 意志品质局限
- 基础的强化
- 什么是“基础”？
  - 技术能力
  - 非技术能力
- 需要在工作以外的时间投入吗？
- 理念和习惯的养成

# 技术路线选择：找到竞争力

- “工作为什么越来越难找了？”
  - 向上走，自然而然的结果
  - 竞争力：工作多年的经验 vs 工作一年的经验重复多遍
- “套路” + “体系”
  - 年纪大了，程序员应该学得更快，而不是更慢
- 北美大厂的面试玩法
  - 流程
  - 人员构成
  - 都会问一些什么问题，考察什么方面

# 第二部分：认识全栈工程师

- 什么是全栈工程师，历史背景
- 为什么要成为全栈工程师
- 怎样成为全栈工程师

# 什么是全栈工程师

- 工程师我知道，可何为“全栈”
- 历史背景
- 全栈  $\neq$  全能
- 核心：Web 技术
  - 网络
  - 前端
  - 后端 MVC
  - 持久化技术等

■ RECENT NEWS » Read Our Blog ■

## Full Stack Web Developers

June 03, 2008



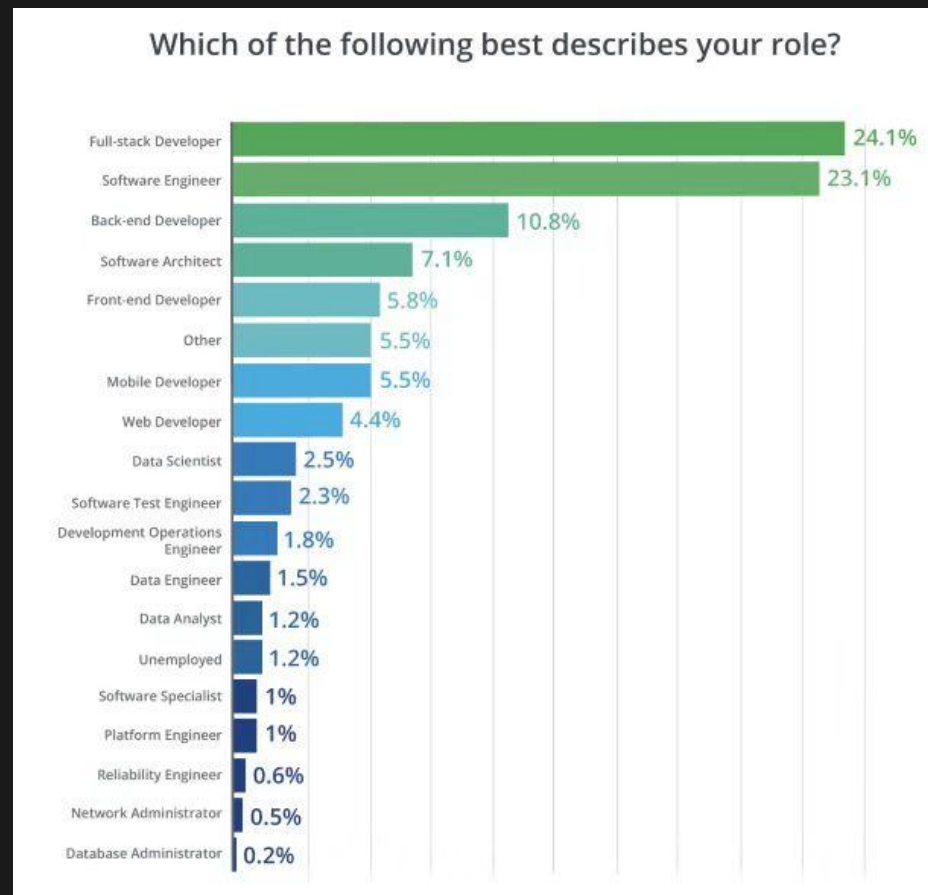
Over the past few months, I have really started to admire what I call “Full Stack Web Developers”. A full stack web developer is someone that does design, markup, styling, behavior, and programming. Some examples of these types of developers are [Jason Tremblay](#), [Geoffrey Grosenbach](#) & [Tom Preston-Werner](#).

[Jason Tremblay](#) is a Philadelphia local who I met a little over a year ago. Recently, we started working together on a little project called [iSepta](#) where he did all of the user interface design. I was blown away with what he was able to produce as well as the development feedback he gave. I literally could not stop grinning from ear to ear.

[Geoffrey Grosenbach](#) is fairly well known in the Ruby on Rails world for the screencast and ebook company he put together called [Peepcode](#). He has done everything for that site from design to development as well as all of the editing and

# 为什么要成为全栈工程师

- 对于个人发展
  - 全栈和互联网的紧密联系
  - 快速反馈的学习方法
  - “技术进程多风雨，唯有套路得人心”
    - 看问题套路化、体系化
  - 视野的拓展：学得杂 -> 学得精
- 对于就业创业
  - 超强的研发独立性
  - 最多的就业机会
  - 职业延伸能力



# 怎样成为全栈工程师

- 先成为合格的工程师，再谈全栈
- 比较 + 思考
- 亲自动手实践
- 目标：套路 + 体系

