



IT大咖说  
知识共享平台

# 从工程师到技术管理

魏自立

# 个人简介

- 2009~2011 HIPIHI创业公司
  - 3D虚拟世界、IM
- 2011.4~2017.10 奇虎360
  - 360云查杀引擎、Poseidon百万亿日志搜索引擎
- 2017.11~至今 今日头条
  - 技术总监



# 成长是自己的事情



# 好好学习，天天向上

- 不积跬步，无以至千里；不积小流，无以成江海

# 学生时代

- 好好咀嚼计算机经典书籍
- 做项目
  - 用C++写一个网络库
  - 实现一个网络服务器，例如带持久化存储的redis或memcached
  - 压测性能，优化性能
- 找大公司实习，最好是核心团队

# 入职职场

- 加入一家高速成长的公司修炼技术，快速提升自己
- 多学
- 多问
- 多做

# 怎么学

- 学习现有项目的代码、架构、依赖库和依赖的上下游服务
  - 关于单例模式代码的讨论
- 学习项目相关的业务知识
- 了解项目背景
  - 上游、下游、服务器部署拓扑图、机房部署、客户端量级
- 多看经典书籍
- 善于总结
  - 写技术博客

# 怎么问

- 看代码过程中的疑问记下来，多攒几个一起问
- 制造机会
  - 一起喝茶
  - 一起吃饭



# 怎么做

- 了解项目背后的业务知识
- 从小事做起，学会吃亏
- 从hold住自己的模块开始
  - 从模块到项目
  - 项目上下游
- 抽象公共组件，并积极推广
  - 例子1：nginx扩展模块抽象
  - 例子2：加解密库
  - 例子3：规则引擎

# 优秀程序员的品质

- 认真负责
- 积极主动
- 追求极致
- 承诺必达

# 积极主动

- 主动沟通、主动询问进展
  - IM、邮件是最主要的沟通形式，但有时不是最有效的
  - 当面沟通，电话沟通，召集会议都是有效的形式，但要留下文字如wiki
    - “我以为发出邮件，他就知道了”，实际上他根本不在邮件列表中
- 主动思考，有效沟通
  - 沟通前要独立思考，并提供充分的上下文和明确重点信息
  - 沟通要达到效果

# 认真负责

- 责任心要强
  - 对自己工作质量负责，不要期待别人发现自己的问题
- 不要想当然
  - 不要默认“没问题”，而是缺省“有问题”，“肯定没问题”一般有问题
  - “我以为他们已经开始做了，但我也没有跟他们确认一下”，导致项目延期
  - “我以为这个接口是这样定义，但谁想到是那样”，导致程序崩溃
- 形成正循环
  - 做好一件小事情，再做另一件稍大一点的事情，慢慢担负起更多责任

# 从项目负责人开始

- 带1~3人负责一个项目
- 熟悉项目的业务、系统、流程、对接人
- 意见领袖：项目的需求迭代开发、重构、技术选型
- 需要重视
  - 培养新人，让他们能力快速提供，成长为自己的左膀右臂
  - 培养出项目的下一个负责人
- 例子：云查杀重构项目
  - 带领3人小组

# 小组Leader

- 5~10人
- 技术带头人、救火队长
- 培养业务负责人，培养接班人
- 关键标签：技术强人
- 提升技术氛围

# 问题：如何服众？

- 技术leader一定要在技术上让大家信服
- 团队里有一个技术比较资深的老人？
  - 相关事情都找他商量给足面子

# 技术经理

- 10~30人
- 团队规划
  - 人员梯队
  - 技术规划
  - 业绩目标规划
- 绩效管理
- 培养小组Leader
- 培养接班人
- 提升技术氛围



# 绩效管理很重要

- 不要搞一言堂，跟核心成员一起商量
- 优秀和待改进最好能让核心成员投票决定
- 慎重给优秀
  - 小心捧杀

# 技术总监

- 数十人
- 抓重点项目
- 抓住团队核心成员是关键
  - 业务线负责人、技术专家
  - 调动并提升核心团队成员的积极性和自主性

# 来点问题吧

- 技术、产品、管理怎么选？
  - 技术专家、综合型的产品经理、技术管理、纯管理
  - 先了解自己
    - 更在乎什么？成就（回报、名气）、兴趣
    - 更擅长什么？
  - 纯管理岗位市场上比较少见，不到高位不要选这条路
- 机遇和努力哪个更重要？
  - 机会是可遇不可求，努力是实实在在可控的
- 广度还是深度？
  - 没有绝对，但某一个方向的深度一定要有，深度就像T型的竖线



先付出、先积累，后收获



Thank You

Q && A