

QCon
全球软件开发大会

技术管理基本功修炼 ——从骨干到带队的转型实践

顾彬彬

同程旅行/研发总监



转型初期，你是否有这些困惑？

个人能力与团队效果的落差

写代码顺风顺水，带团队却力不从心。自己能搞定的问题，团队却经常卡壳，感觉管理比写代码难太多了。

管理范围的认知偏差

以为只盯代码质量就够了，却发现还要操心效率、成长等“额外事”。代码没问题，项目却延期，成员没进步。

管理目的的迷茫

不知道管理到底为了什么，陷入“瞎忙”。每天做很多事，却没拿到核心结果，感觉缺乏方向感。

目录

01

管理初心：技术管理到底为了什么？

02

转型准备：从技术骨干到技术管理需要做哪些转变？

03

核心职责：哪些事是技术管理者必须吃透的？

04

基础技能：除了技术，还要补哪些管理能力？

05

平衡艺术：怎么管好人、理好事，事半功倍？

📍 北京

QCon

全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AIOps
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

📍 北京

AiCon

全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

📍 上海

QCon

全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月

5月

6月

8月

10月

12月

📍 上海

AiCon

全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

📍 深圳

AiCon

全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

📍 北京

AiCon

全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LMOPs
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播



参会咨询



查看会议

01 管理初心

技术管理到底为了什么？

先想透：技术管理到底为了什么？

不是“当领导”

而是做“双驱动”支撑：既要驱动项目高效交付，又要驱动团队成员成长。管理者的价值在于放大团队效能，而不是展示个人权威。

不是“甩手掌柜”

而是“桥梁与后盾”：向上对齐目标和资源，向下解决障碍和困难，让技术资源在有序调度中创造最大价值。

● 驱动效率与交付的升级：以“有序”释放技术资源的最大价值

1

人力盘点

建立“团队成员任务表”。明确谁擅长什么、当前任务饱和度如何。避免“老员工超负荷、新人没事干”的资源浪费。

2

流程优化

每周排期会，明确“任务优先级+负责人+截止时间”，减少“重复返工”和“任务交叉”带来的效率损失。

3

成果验证

跟踪“交付周期”和“需求返工率”。如转型后交付周期从15天缩至10天，返工率从20%降至5%。

⊗ 效率提升的核心不是“逼成员加班”，而是“让资源有序流动”。

● 助力团队成长迭代：打造可持续的技术梯队

识别优势

识别个体优势，记录组员擅长领域，充分发挥组员的长处，合适的人做正确的事，才能事半功倍

搭建梯队

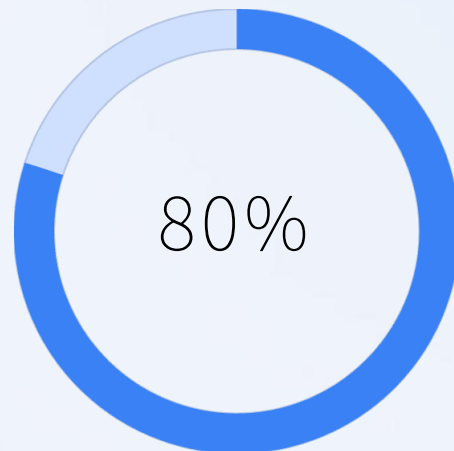
新人→辅助→核心→骨干。每个阶段配”成长任务”：新人改bug+写注释，辅助做简单接口+参与评审。

长效保障

每月1次”成长复盘会”，明确成员想提升的能力和团队能提供的支持。



新人成长



技能覆盖

02 转型准备

从技术骨干到技术管理需要做哪些转变？

从“盯代码”到“盯人盯目标”

技术骨干

- 关注“代码写得对不对、优不优”
- 拿到需求先想“怎么实现”
- 用“代码行数、bug数”衡量成果
- 追求技术完美和个人产出

技术管理

- 关注“人有没有动力、目标有没有对齐”
- 拿到需求先想“为什么做、谁来做、要达成什么结果”
- 用“目标达成率、团队成长度”衡量成果
- 追求团队效能和业务价值

从“单枪匹马”到“跨部门协调”

主动对接

定期和产品、运营开“同步会”，提前确认“需求优先级”、“需求的边界”、“迭代的方向”，避免“临时催活”。

1

2

做好“翻译官”

把产品需求“转化”为技术语言，同时做好技术方案成本和风险的评估。

承担协调责任

运用好资源协调的权利，承担跨部门协调责任。遇到部门间意见分歧，先找“共同目标”，再谈“怎么落地”。

3

从“不管杂事”到“兜底担责”

避免误区

- 这个bug不是我写的
- 产品需求不明确才导致延期
- 测试没测出来不能怪我
- 遇到问题第一反应是去解释不是自己的问题

”杂事”不杂

组员请假、设备故障、需求临时变更，这些”杂事”直接影响项目进度，管理者必须主动接手处理，不能视而不见。

兜底做法

- 主动承担团队责任
- 优先解决问题而非追责
- 建立预防机制避免重复错误
- 保护团队成员的积极性

担责不甩锅

项目出问题，对内先找”怎么改”，对外主动承担（”是我没协调好/没预估到位”），事后及时复盘原因。

从“自身成长”到“带团队成长”

自身成长模式

- 关注个人技能提升
- 独自学习新技术
- 积累个人技术经验

带团队成长模式

- 关注团队整体能力提升
- 定期组织技术分享会
- 为成员创造成长机会
- 建立知识传承机制

1

技术分享机制

每周固定时间组织团队内技术分享，轮流由不同的组员主讲，可以自主报名。既能传播知识，也能锻炼表达能力。

2

项目轮岗机制

让组员在不同项目或模块之间轮换，接触不同的业务模块和技术栈，拓展技术视野，避免很多年同一个人负责同一个模块。

3

导师带教机制

为新人安排经验丰富的导师，建立一对一的指导关系，做好新人的带教培养。

03 核心职责

哪些事是技术管理者必须吃透的？

目标锚定：团队才不“瞎忙”

目标误区

以"需求数量、代码行数"论英雄

- 每月必须完成10个需求
- 代码量越多越好
- 加班时间等于工作成果
- 忽视业务价值和质量

正确做法

以"业务价值、目标达成率"为核心

- 用户留存率提升
- 订单支付成功率提升
- 接口性能提升，页面加载时间减少
- 关注最终用户价值

OKR拆解工具示例

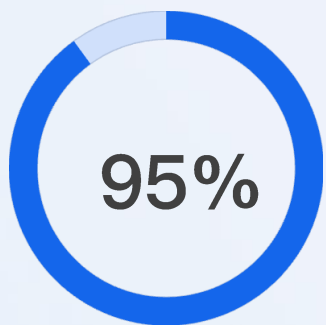
O：Q2完成交易创单成功率从95%提升至98%

KR1：4月份完成重复单优化，失败率从1.2%下降至0.2%以下，提升创单成功率1%

KR2：5月份完成验价失败逻辑优化，提升创单成功率1.5%

KR3：5-6月份完成支付接口超时重拾逻辑优化，提升创单成功率0.5%

需求管理：管好需求，交付不慌



按时交付率



返工率降低

需求接收

建立标准化的需求接收流程，确保需求信息完整准确。设置需求优先级评估机制，避免紧急需求打乱正常节奏。

时长评估

合理评估开发时长。“拍脑袋”“倒排期”不可取，综合考虑技术难度、人员配置、风险因素等，给出相对准确的时间预估。

任务分解

将大需求分解为可执行的小任务，每个任务控制在1-2天内完成。细化的任务颗粒度有助于进度跟踪和风险识别。

过程管控

建立日常站会、项目周会、里程碑检查等过程管理机制。及时发现偏差，快速响应和调整，确保按时交付。

质量和效率双抓，避免“返工救火”

需求阶段

需求评审阶段必须与产品明确边界，尽量避免“开发后改需求”

开发阶段

开发强制单元测试（覆盖率 $\geq 80\%$ ）、代码评审，提前挡bug，减少测试返工

测试阶段

拉齐bug标准（致命bug24小时内修复），避免“上线前还在改关键问题”

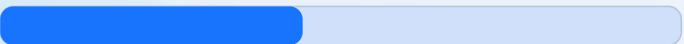
上线阶段

灰度发布，及时组织线上验收和巡检，降低线上故障影响范围和时长



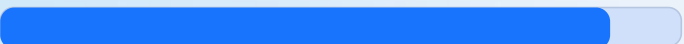
60%

线上bug减少



40%

测试效率提升



90%

系统满意度

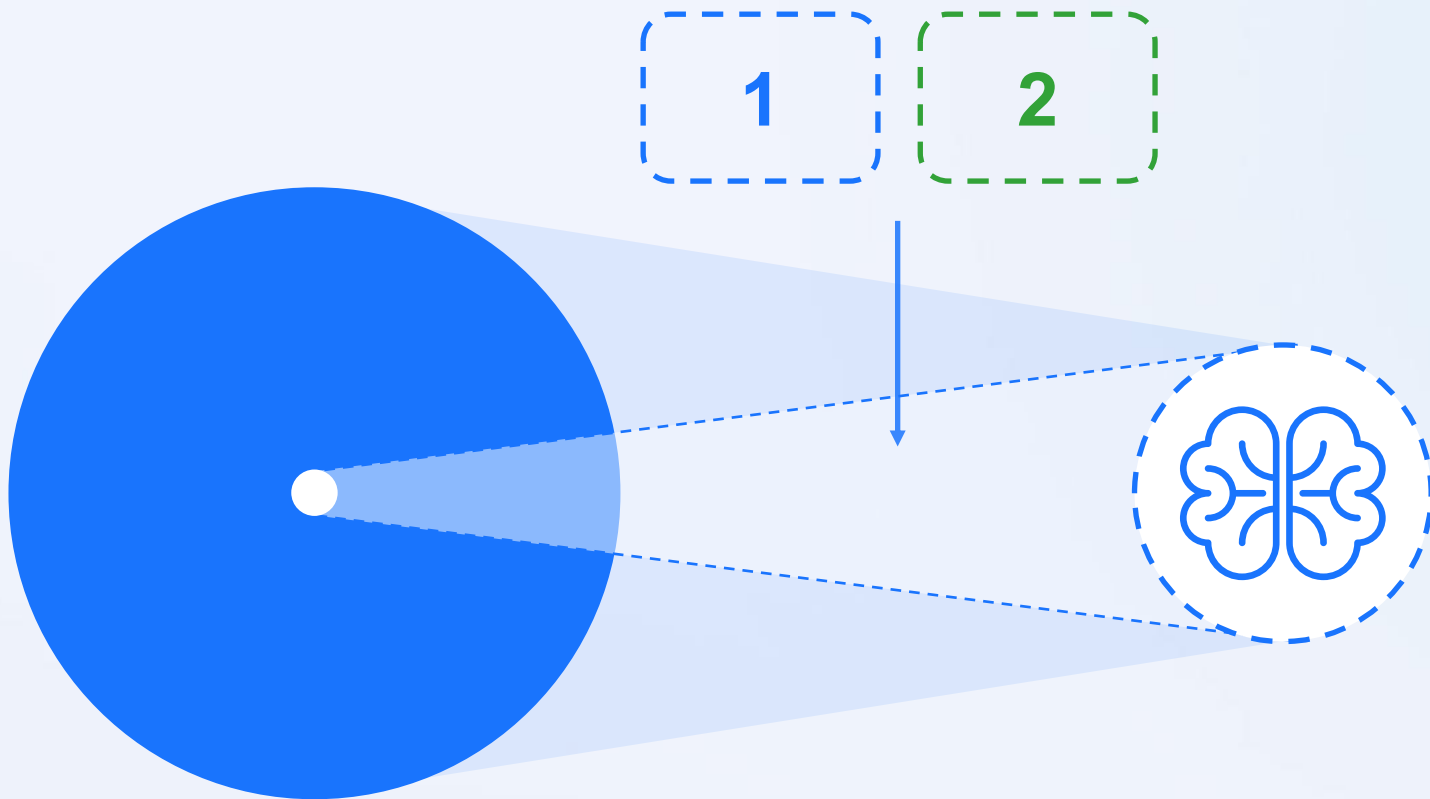
● 做对技术决策，少走“技术弯路”

技术选型的原则

- 成熟度：不要盲目追求新的技术栈和框架，调研好技术成熟度
- 适配性：过分依赖技术驱动，采用复杂的算法和技术实现，导致功能不符
- 团队能力：成员会用或容易上手学习

系统重构的判断

- 决定什么时候重构是“艺术活”
- 系统重构的范围、要解决的问题
- 系统重构失败的备案、成功的目标



● 做好技术运营，别等故障才重视

建立监控体系

制定应急预案

定期健康检查

故障复盘总结

技术指标监控

- **关键指标：**接口相应时间<500ms，服务器CPU使用率<80%
- **监控策略：**设置合理阈值，超过阈值自动告警，异常情况立刻排查

故障响应机制

- **响应时效：**1min发现，5min定位，10min解决
- **处理流程：**先止损，再排查根因，最后复盘总结

定期健康检查

- **健康检查：**定期进行系统健康评估，识别潜在风险点，制定优化计划
- **故障演练：**定期进行系统故障演练，通过故障注入等方式保障系统高可用

04 基础技能

除了技术，还有补哪些管理能力？

基础技能：排兵布阵，识人用人的智慧

能力-意愿矩阵

	能力高	能力低
意愿高	明星员工能拿结果的人	潜力股愿意做事的人
意愿低	问题员工需要激励沟通	风险员工考虑调岗

用人策略

识人重点：关注”能拿结果的人”(能力高+意愿高，主动解决问题)和”愿意做事的人”(能力低+意愿高，有成长潜力)。

用人原则：把复杂模块(核心支付、性能优化)分给”能拿结果的人”，把基础任务+导师带教(改bug、写简单接口)分给”愿意做事的人”。

任务-能力匹配示例

任务	所需技能	匹配成员
支付模块	后端+安全经验	老周
前端优化	前端+性能调优	小张
数据统计	SQL+数据分析	小李
接口开发	基础后端+学习意愿	小王(带教)



基础技能：高效会议，省出时间

需求对接会

参会人：技术+产品

核心目标：对齐需求内容、边界

流程：产品讲需求→技术提疑问→确认需求范围和验收标准

技术评审会

参会人：研发+测试

核心目标：控技术风险

包含：概要设计评审、Code Review会、测试用例评审

项目进度会

参会人：专项全员

核心目标：同步进度、解卡点

流程：每人1分钟说”做了啥/要做啥/卡在哪”，问题当场协调

团队成长会

参会人：全团队

核心目标：带团队成长

流程：团队周会、技术分享、聊成长需求、确定下月计划

注意事项

每类会都有明确的目标，才不会浪费时间，及时喊“停”

按阶段开对会，技术团队不存在汇报会，开会不是“凑人数”

安排合适的人去参加会议，专注与自己相关的内容，不建议带电脑

基础技能：控好风险，避免“突发危机”



基础技能：做好绩效，妥善处理申诉

绩效考核

- 公平公正，不带个人主观偏见
- 目标明确，否则评估会比较模糊
- 分层分类考核，按照组员角色差异化设定指标
- 数据公开，比如个人任务完成率、bug率、返工率等

绩效沟通

- 目标同步：提前沟通月度/季度考核目标，考核前明确指标
- 正向沟通：强化优势，放大动力，及时表扬，针对性激励
- 改进沟通：指出问题，给出改进方案，切忌单纯批评
- 绩效申诉：听诉求→查数据→给方案，透明公正的绩效管理是团队稳定的基石。不要怕申诉，更不能被申诉裹挟。

05 平衡艺术

怎么管好人、理好事，事半功倍？

在实践中练管理能力

管理能力不是"学一次就会"，是"在实践中练出来的"。

● 管人与理事的核心——靠团队成事

误区警示

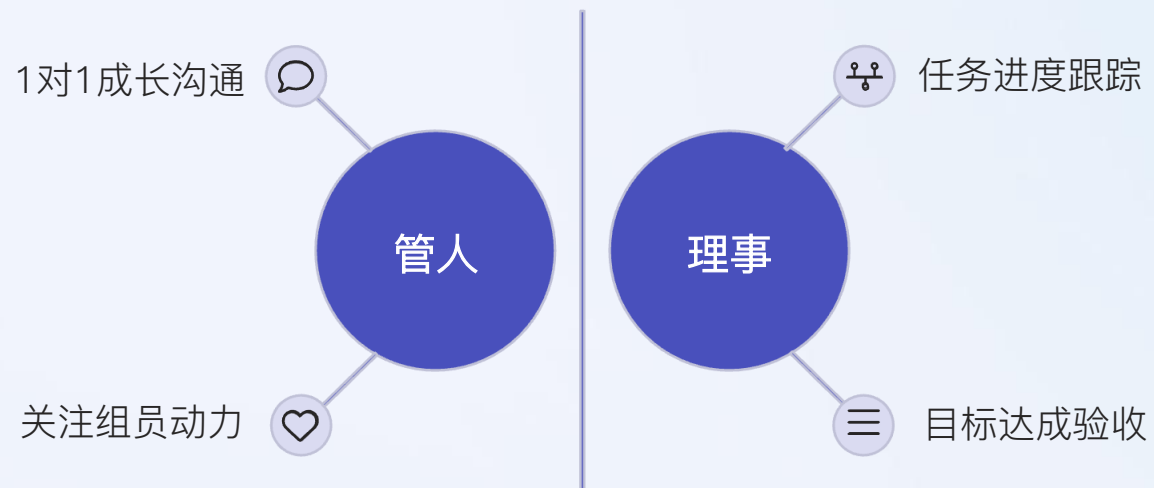
不管人只理事：团队没动力、离职率高

只管人不理事：项目没成果、拿不到结果

平衡做法

管人：每周1次1对1聊成长需求(让团队愿意干)

理事：每天盯3个关键任务进度(让团队知道怎么干)



总结

你不用会所有技术，但要会带团队解决问题；你不用干所有活，但要会让团队一起出成果，这才是转型的核心。

坚守技术管理初心，夯实技术管理基本功，让团队变厉害！

📍 北京

QCon

全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AIOps
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

📍 北京

AiCon

全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

📍 上海

QCon

全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月

5月

6月

8月

10月

12月

📍 上海

AiCon

全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

📍 深圳

AiCon

全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

📍 北京

AiCon

全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LMOps
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播



参会咨询



查看会议

THANKS

大模型正在重新定义软件

Large Language Model Is Redefining The Software

