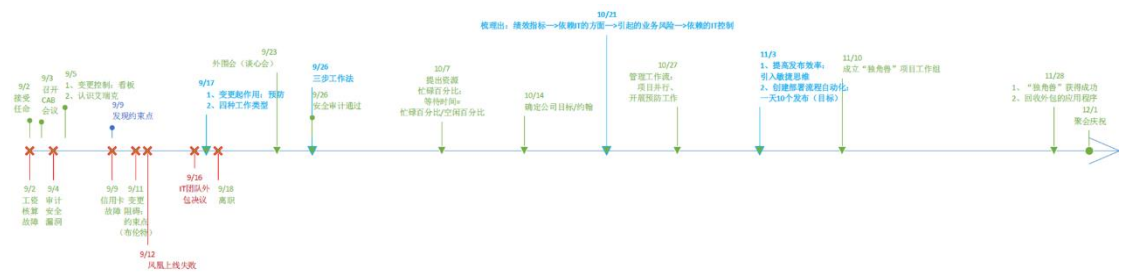


凤凰项目时间轴



第一点：约束点

“哦，天哪。”我喃喃自语，“布伦特。布伦特、布伦特、布伦特！要是没有他，我们就什么事情都干不成吗？”

布伦特：运维技术大牛

关键人物布伦特因处理工资系统故障，延误了凤凰项目环境的准备。

大多数审计发现都需要布伦特的介入，已经聘请了同样资深的人员，但知识都在布伦特脑中，布伦特忙的没有时间对其他人进行培训；

故障讨论会议中，大家彼此推卸责任，互相指责。事故在讨论会议期间被布伦特快速解决，因布伦特意识到是之前的一个操作造成故障发生，快速回滚。

业务部门再次投诉布伦特没有按时完成凤凰项目的任务。比尔到布伦特座位附近观察布伦特的工作，发现布伦特的工作总是不断的被打断。虽然凤凰的优先级最高，但是因为只有布伦特了解关键系统，业务领导威逼利诱，导致他无法聚焦凤凰。

本周有 240 个变更没有完成，一个月以后将会有 1000 个滞留。很多变更需要依赖布伦特。半成品=浪费。所有依赖布伦特的变更需要备注，统一安排。

建立三级人力资源库用来解决问题，让布伦特的电话静音，只有三级人力资源库可以接触布伦特，记录学习到的知识，同样的问题不允许布伦特出手第二次。

在小说中，一名称为布伦特的高级工程师成为了约束点，任何关键任务中他都必不可少。

约束点最大产出就是工厂最大产出。

发现约束点，围绕约束点展开工作，提高约束点的生产效率。

扩大约束点，使约束点不再是约束点。

第二点：开发环境，测试环境，rc 环境，正式环境一样，提高转变效率。

公司	部署频率	部署交付期	可靠性
亚马逊	23 000 次/天	分钟	高
谷歌	5500 次/天	分钟	高
网飞	500 次/天	分钟	高
Facebook	1 次/天	小时	高
Twitter	3 次/周	小时	高
普通企业	9 个月 1 次	月或季度	低/中

版本控制对象： 代码以及创建环境所需的每一样东西

自动化环境创建流程、

在部署管道中创建测试和生产**环境**，然后彻底往里面部署代码，从而缩短准备时间排除故障。从而跟上开发部的工作节奏。

第三点：开发运维。

部署管道：从代码签入到投产的整个价值流

问题：开发环境、QA 环境、生产环境各不相同，同步困难，并且环境无法准时就绪，从而引入很多返工等非价值附加（废料），

解决：

通用环境环境流程，从而使环境标准化，解决环境同步问题。

环境的自动部署，避免由于环境部署的错误而引入的返工。

第四点：三步工作法

第一步：从左到右，开发，运维，客户

第二步：从右到左快速持续反馈流。

第三步：创造公司文化，带动两种风气形成：不断尝试，理解重复和练习是熟练掌握的前提。