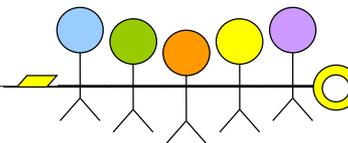


随时听讲座
每天看新文
追随技术信仰

火龙果讲堂

- 一线专家
- 案例回顾
- 经验分享



如何为自己的研发团队 设计开发持续集成系统

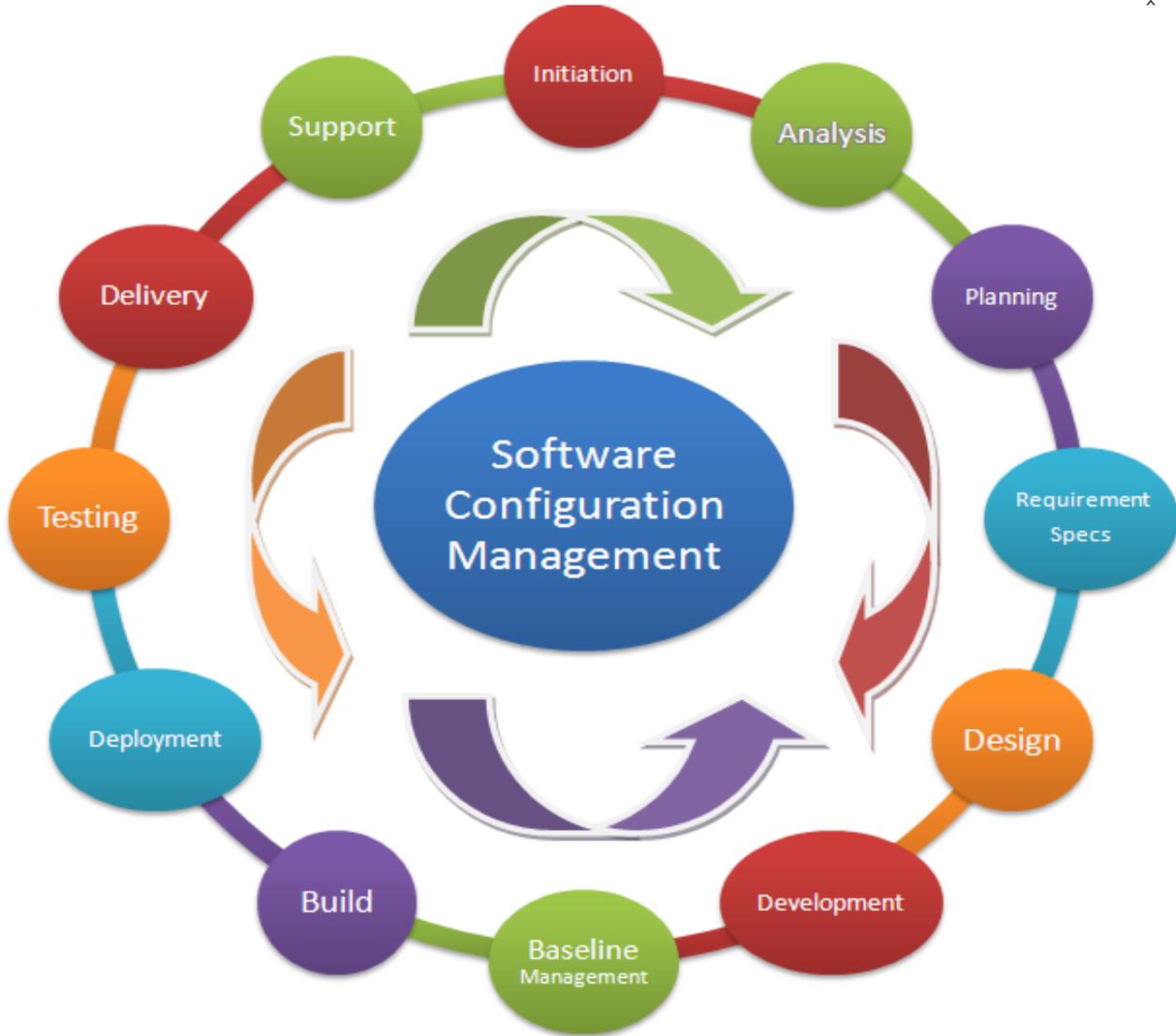
李 鹏

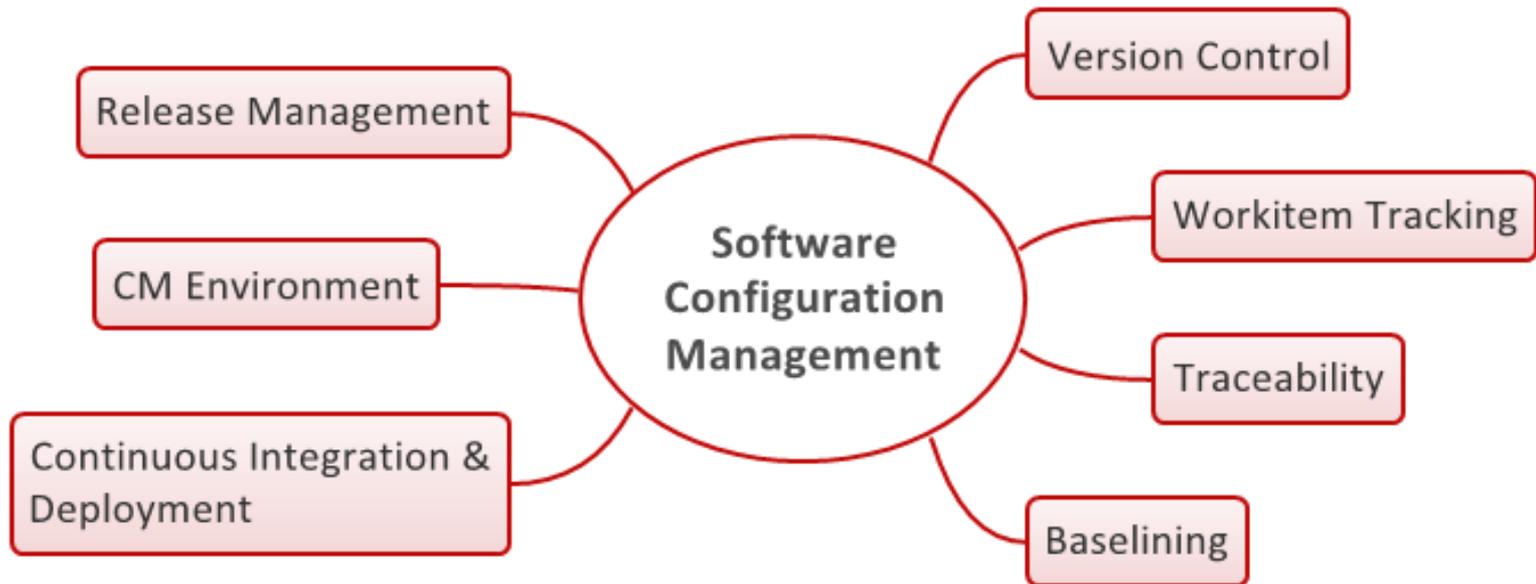
- 软件配置管理
- 分支模型的特点选择
- 环境与构建
- 持续集成与自动化测试和部署
- 持续集成工具
- 答疑

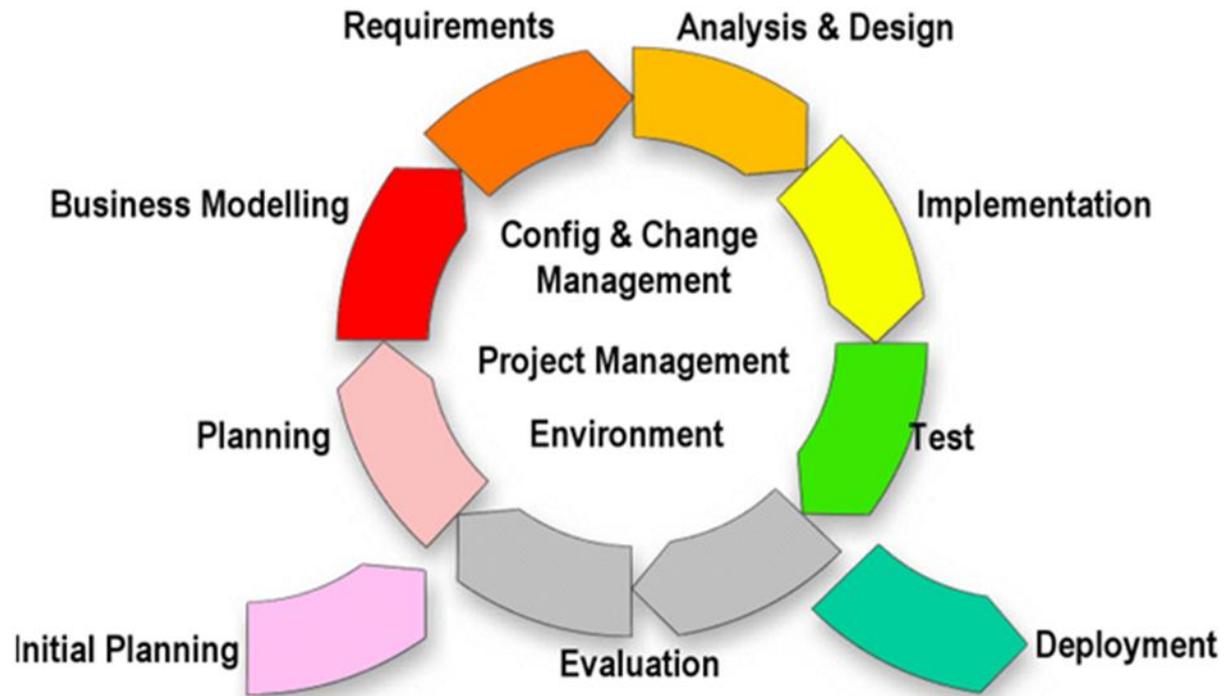
简介，主题之间的关系

软件配置管理与持续集成

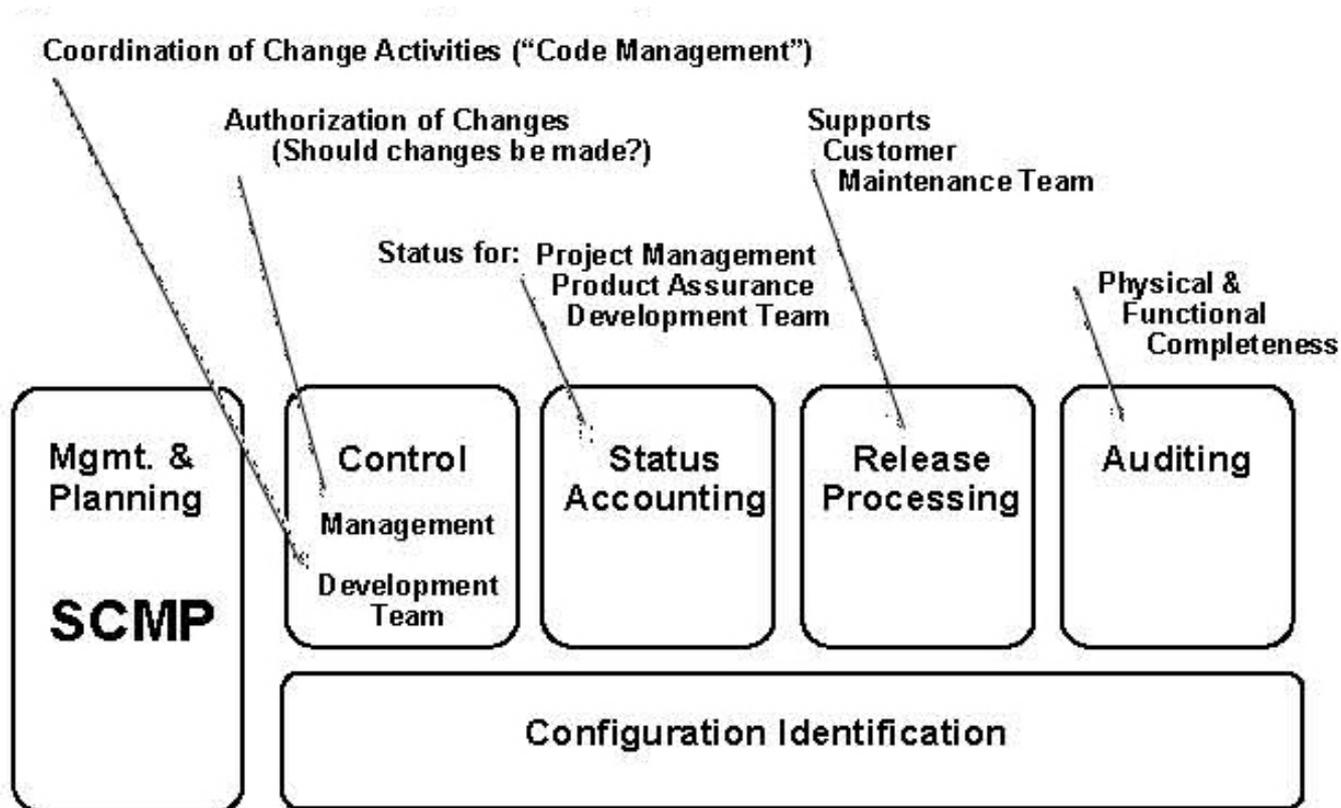
软件配置管理概览

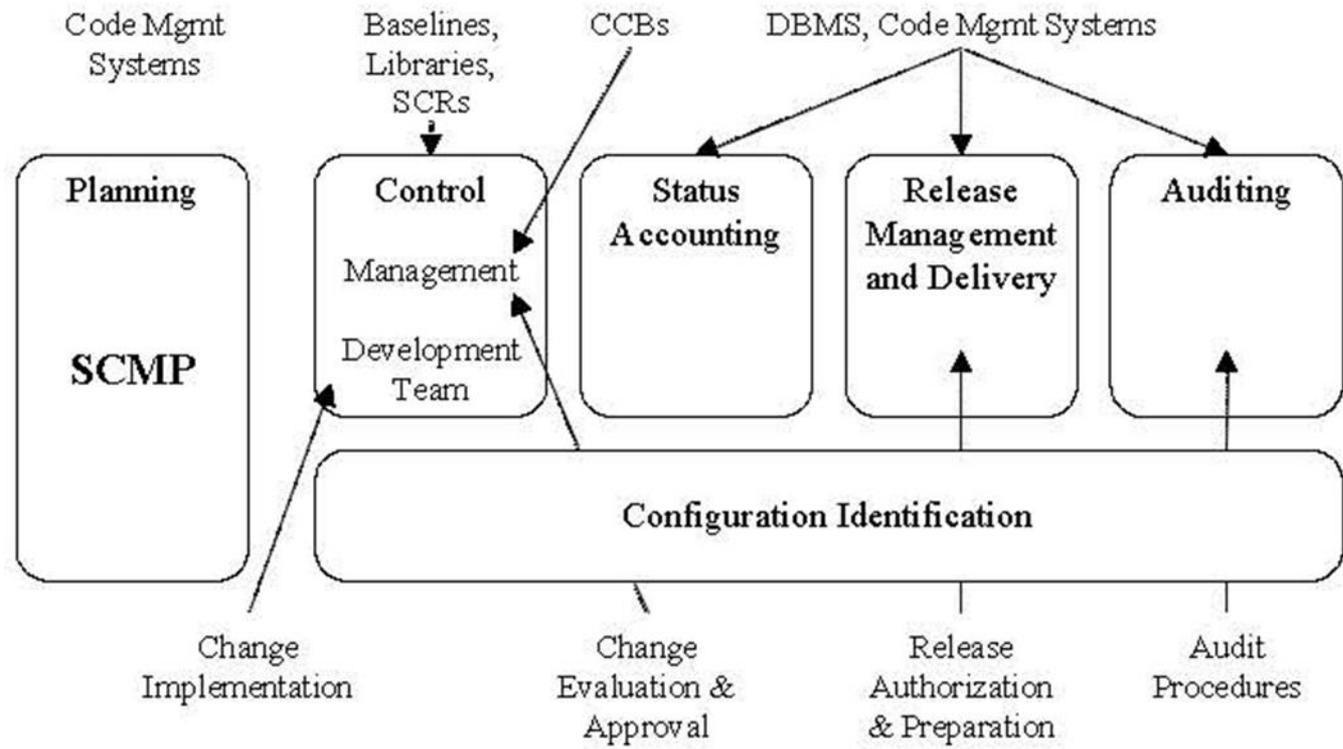


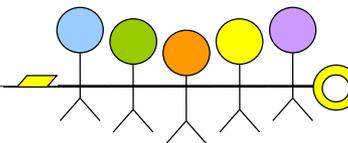




- 配置管理的内容
- 好的持续集成连贯配置管理的所有项







■ 持续集成最基本需要什么？

■ 分支策略，名称规范与约定

– Branch Name

– Tag

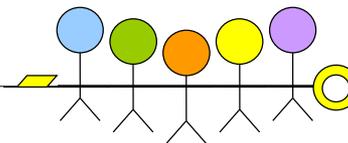
– HOT FIX

■ 代码版本库

■ 说服开发团队持续的代码提交

■ 可控的编译构建环境

■ 持续集成工具



■ 改进的持续集成需要什么？

■ 与缺陷管理系统集成

- Bugzilla, Jira ..

■ 与代码审查系统集成

■ 自动化反馈

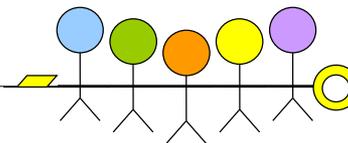
- Email 短信 聊天室信息

- 报表

■ 增加自动化测试

- UT

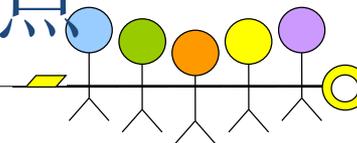
■ 增加自动化部署



■ 优化的持续集成包含什么

- 持续集成平台脚本，任务版本化管理
- 自动测试部署CI improve, CI的CI
- 尽可能可视化一切流程
 - Metrics ,KPI
- 资源监控，自动资源扩展与预警
 - 图形化监控所有的环境
 - Build host , IO, 存储
 - Auto Create Slave
- 高可用性

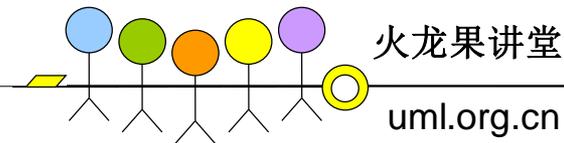
总结：持续集成最佳实践关键点



- 使用版本控制库
- 自动化构建
- 自动化测试
- 每天提交
- 每个提交都应该编译
- 保持快速编译
- 测试环境类似生产环境
- 方便取得最近的发布包
- 每个人都可以见集成的状态
- 自动化部署

分支模型的特点选择

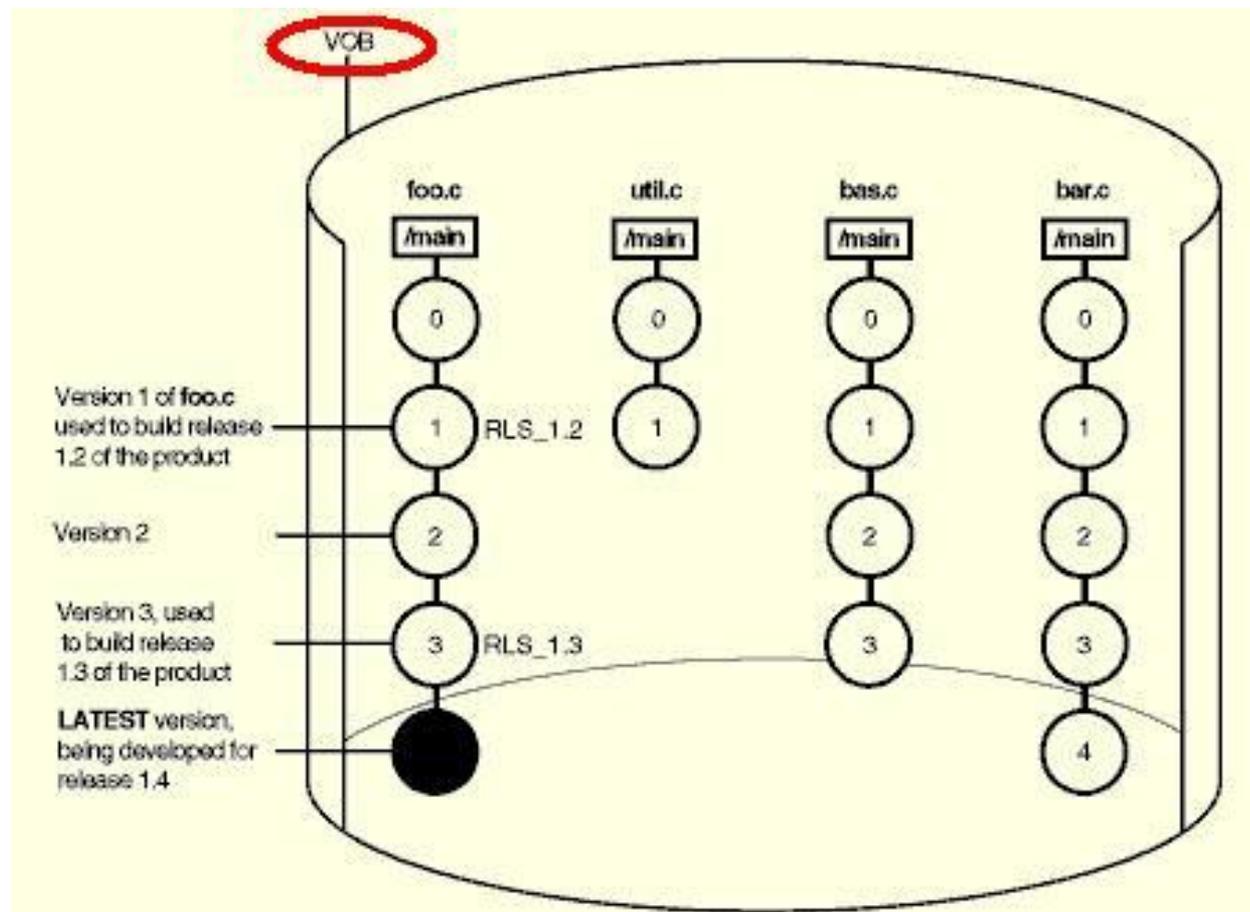
分支模型的特点选择



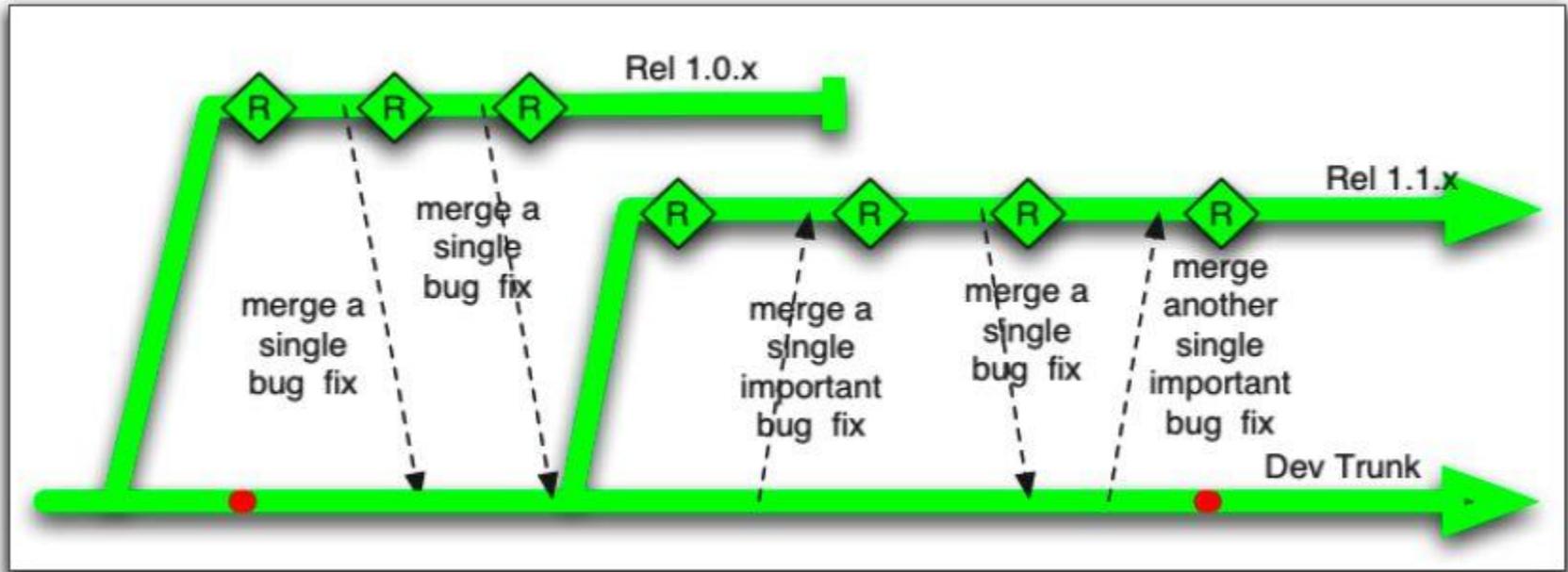
- 基于流的开发
- 主干开发模型
 - 按照发布创建分支
- 功能特性分支
- 团队分支

■ 流如何选择

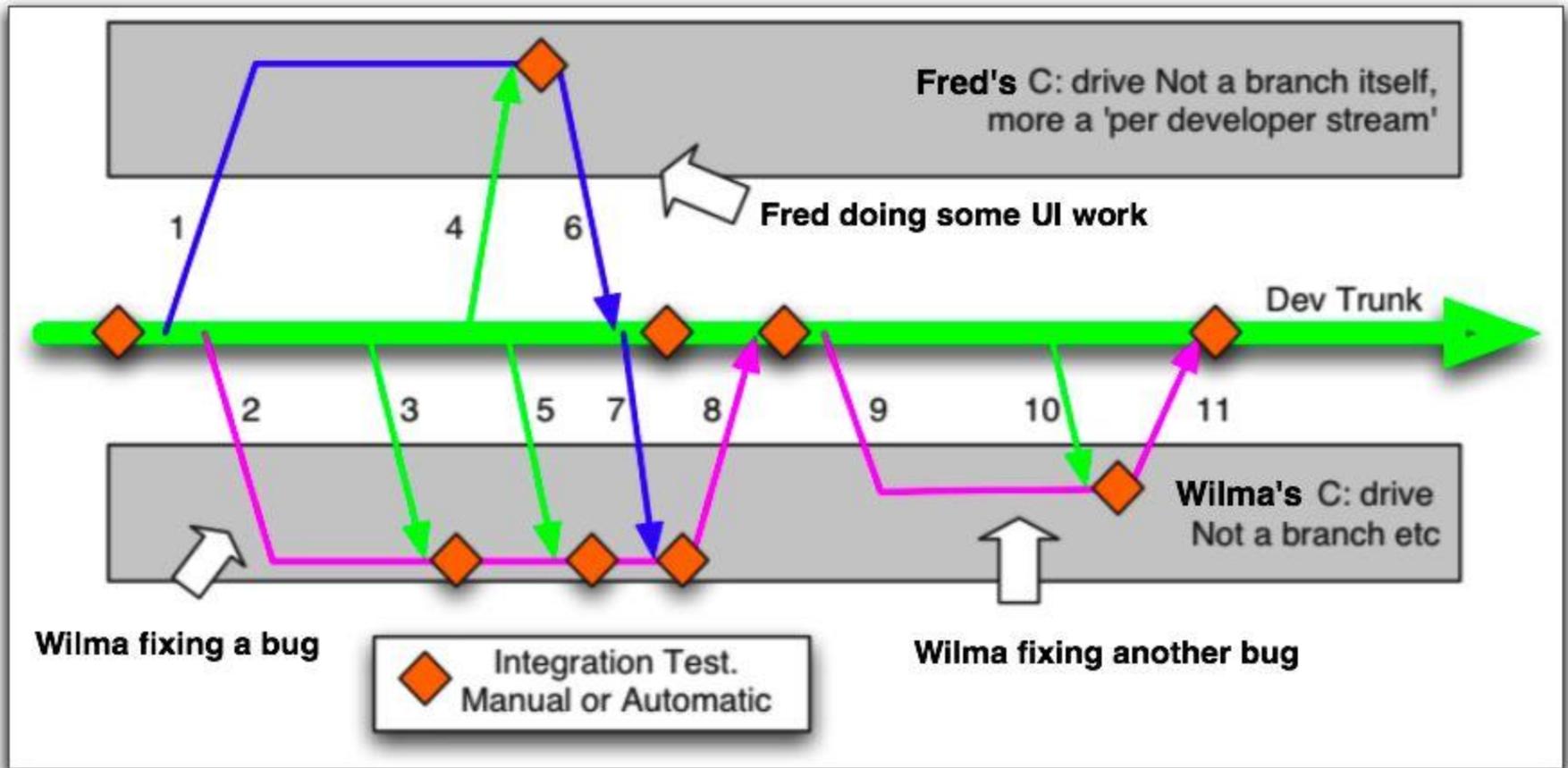
■ 依赖 “CS(configuration specification)”



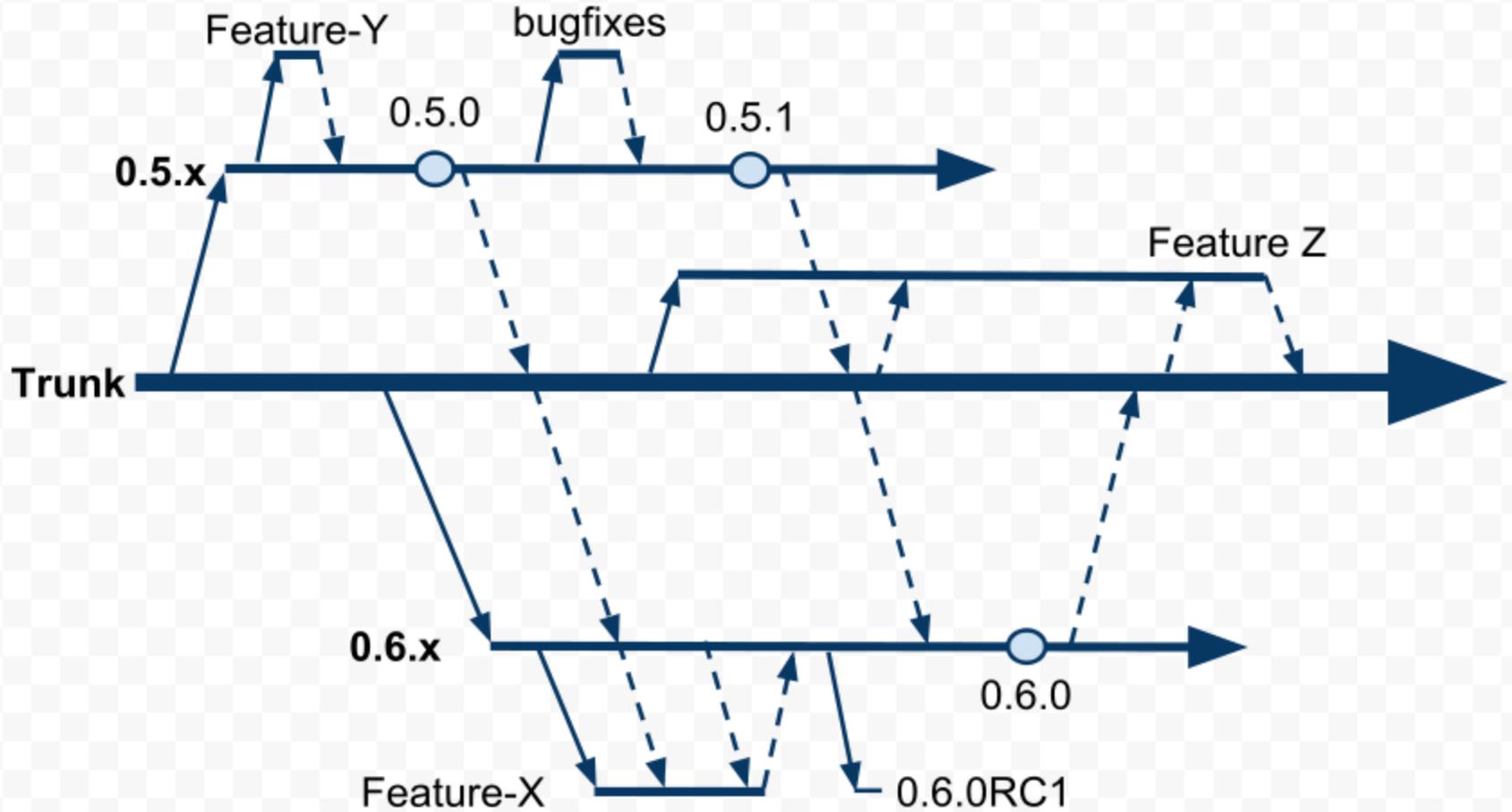
主线开发模式 1/2



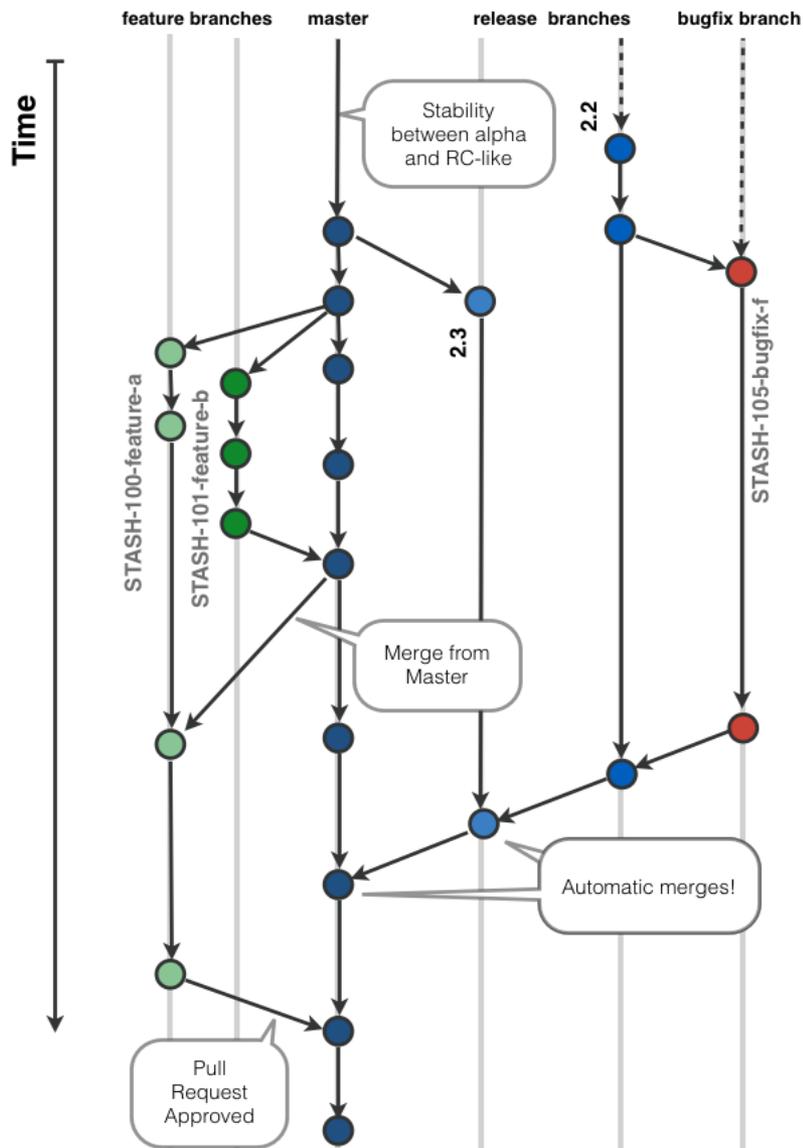
■ 主线开发模式下的开发人员的活动



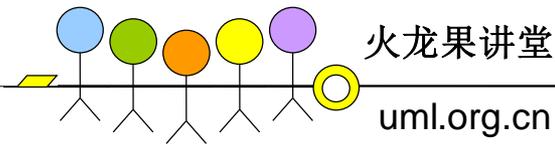
功能特性分支



实际的分支模型例子



如何选择适合自己的分支模型



■ 中小团队首推主线开发模式

■ 考虑因素

- 团队规模
- 代码规模
- 物理地域
- 代码模块化是否清晰
- 第三方依赖

环境与构建

■ 什么可以考虑为环境

■ 最基本的构建/编译/打包的机器

- 硬件配置，操作系统和版本和补丁

■ 依赖的软件包以及版本

■ 网络拓扑结构

■ 需要依赖的外部服务，服务的版本

- 比如第三方的开源软件库

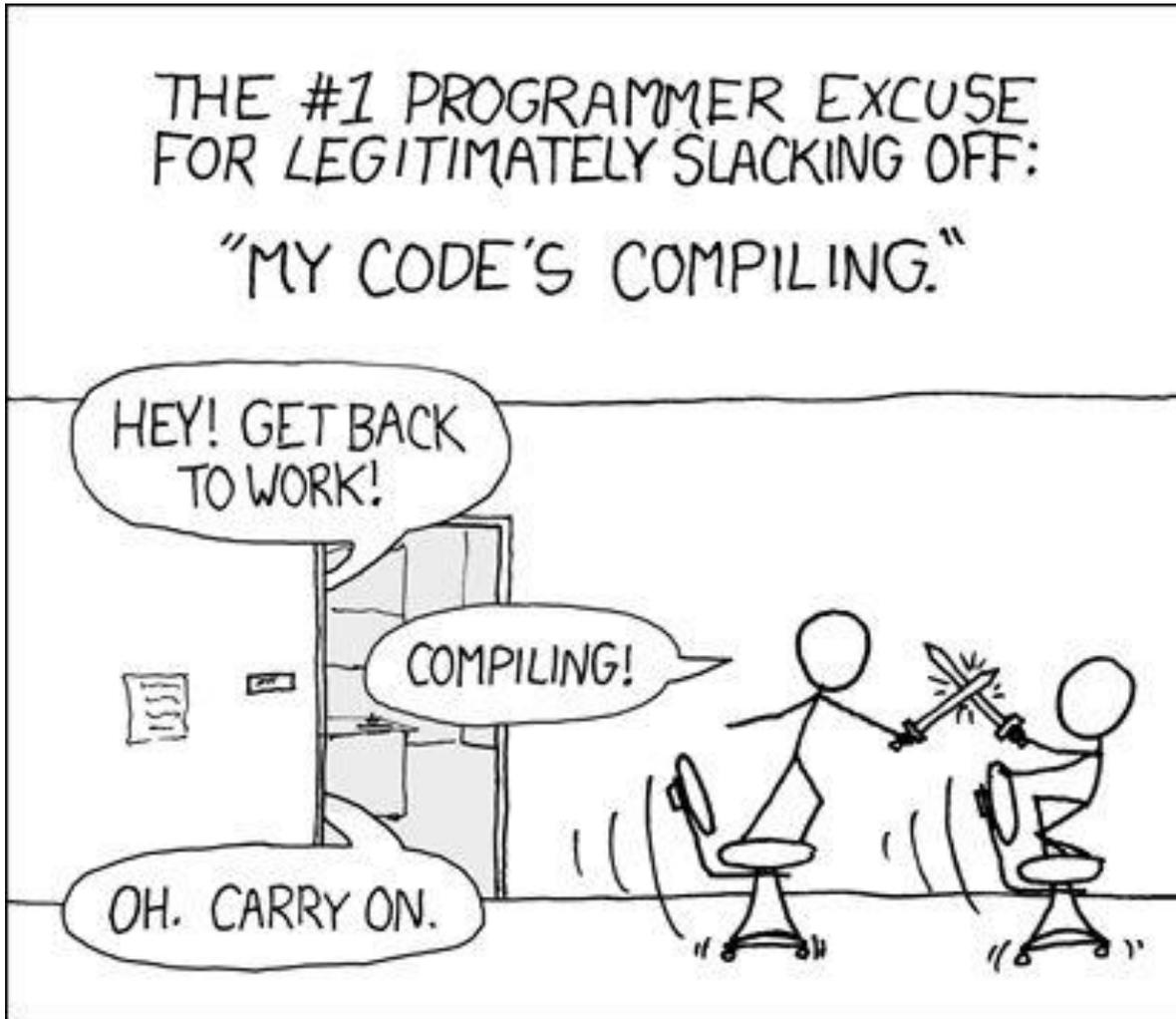
■ 依赖的数据，数据库等信息

■ 测试环境和测试输入数据，比照数据

- 清楚你拥有持续集成环境的资产
- 和其他部门合作
 - 虚拟团队
 - Devops模式
- 定期检查和分析
- 前瞻性
 - 报表和可视化帮助

- 传统的IT模式
- 改进模式
- 自动化环境管理
 - Puppet / CFEngine
 - Ansible
 - 虚拟化
 - Cloud
 - Docker





■ 构建与工具

■ 什么最重要？

■ 可重复

- 知易行难，为什么经常无故失败

- 原因在哪里？

■ 速度，速度，速度

- 机器，CPU core, 网络，系统架构那个最关键

■ 存储

■ 存储速度

■ 定期回顾与改进

改进构建速度

REASONS WHY PEOPLE WHO WORK WITH COMPUTERS SEEM TO HAVE A LOT OF SPARE TIME...

Web Developer



'Its uploading'

Sysadmin



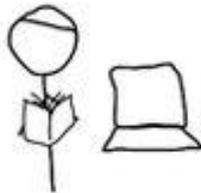
'Its rebooting'

Hacker



'Its scripted'

3D Artist



'Its rendering'

IT Consultant



'Its your problem now'

Programmer



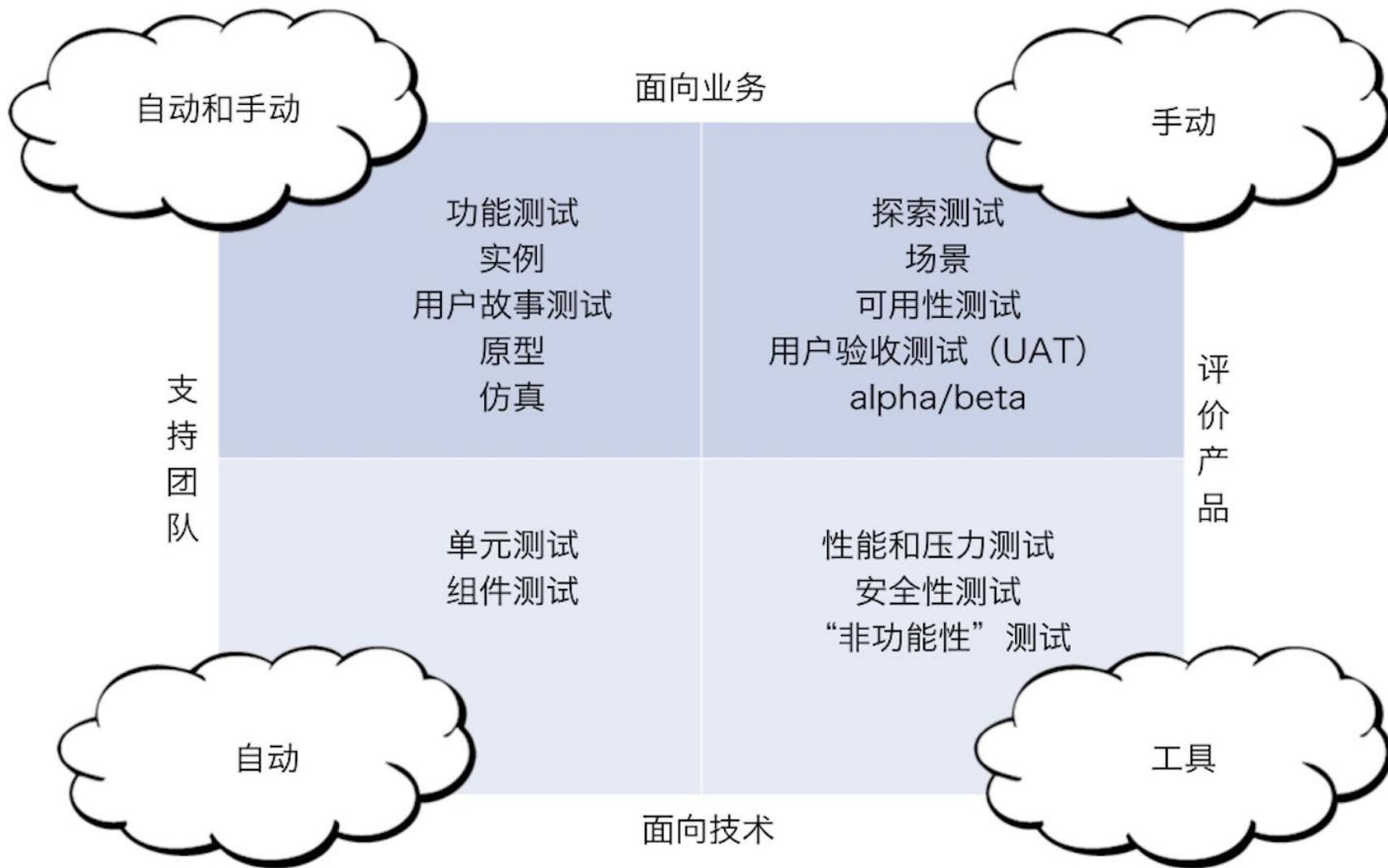
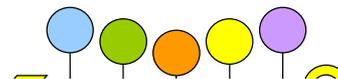
'Its compiling'

- 什么制约了你速度
 - 分析，量化
- 如何改进
 - 什么才是最终要的？
- 如果你有一个超级大的系统？
 - 逐步改进
 - 邀请第三方公司分析评估
 - 使用商业化工具

- 可视化构建过程
- 定期分析，预防破坏者
- 可视化例子

持续集成与自动化测试和部署

自动化测试概览



敏捷测试象限

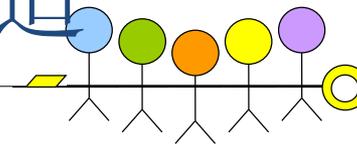
■ 部署

- 只生成一次的二进制包=最终的发布内容
- 不同环境统一部署
- 每次都要在干净等同生产的环境重新部署
- 失败立即退回代码提交
- 自动化

■ 测试，最基本实现

- 最基本的smoke测试或者sanity测试
- 每次提交都要测试
- 测试失败立即退回提交

更多测试什么，那个阶段更合适



■ 持续集成过程中

- 单元测试

- 代码检查

- 静态检查

- 格式检查 语法检查 安全检查

- 动态检查

- 内存泄露 函数安全 移植健壮性 自定义

■ 每晚或者每周，自动或者半自动

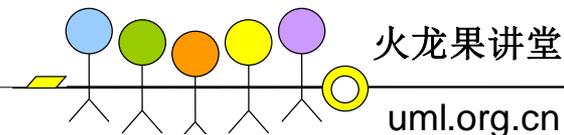
- 验收测试

- 压力测试

- 回归测试

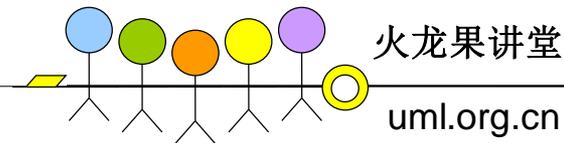
持续集成工具

持续集成工具调查



Option	Adoption	Relevance	Votes	Details
Jenkins	85%	89%	569	Heatmap
TeamCity	72%	81%	192	Heatmap
Bamboo	68%	73%	164	Heatmap
Hudson	62%	69%	130	Heatmap
Cruise Control	56%	54%	129	Heatmap
Go	63%	75%	117	Heatmap
Team Foundation Server	67%	65%	108	Heatmap
TravisCI	64%	75%	98	Heatmap
Anthill	62%	39%	43	Heatmap
Buildbot	56%	61%	37	Heatmap
Continuum	60%	57%	32	Heatmap
QuickBuild	61%	62%	18	Heatmap
FinalBuilder Server/Continua CI	66%	62%	17	Heatmap
Mojo	59%	47%	11	Heatmap
Parabuild	45%	39%	11	Heatmap
Gump	56%	38%	11	Heatmap
Zed	46%	44%	11	Heatmap
easyCIS	46%	40%	10	Heatmap
Sin	47%	48%	9	Heatmap
Pulse	35%	50%	9	Heatmap

该如何选择持续集成工具



- 自己搭建维护还是托管
- 扩展性
- 易用性
- 自己试用对比
- 安装数量
- 社区活跃程度
- 应用开发者的平台

■ 自己搭建，使用jenkins

- 基于Java，无平台依赖
- 安装数量极大
- 社区活跃
- 插架数量极多，非常容易扩展
- 可自己开发插件
- 支持各种自动化测试与部署工具
- 众多cloud集成支持



Jenkins

■ 托管的持续集成平台

■ Travis CI

- 面向开源项目免费，支持超过12中语言
- 已经先后有超过一千万的构建
- 集成了github支持
- 小项目与个人开发适合

- 关于Jenkins回答大家几个问题
 - 持续集成系统的架设和如何管理JENKINS里的CASE?
- 自动化测试的注意要素及部署
- 方泽文（德赛西威，配置管理工程师）：
持续集成平台的使用对象是开发人员还是测试人员？
- 高焕涛（深圳市梦风科，配置管理员）：
公司编译环境多变，项目打包复杂，持续集成可以实现吗？

交流时间



如何完成由手动部署到自动部署
持续集成与自动化测试的衔接

Backup slide

DELIVERY JOB PROCESS

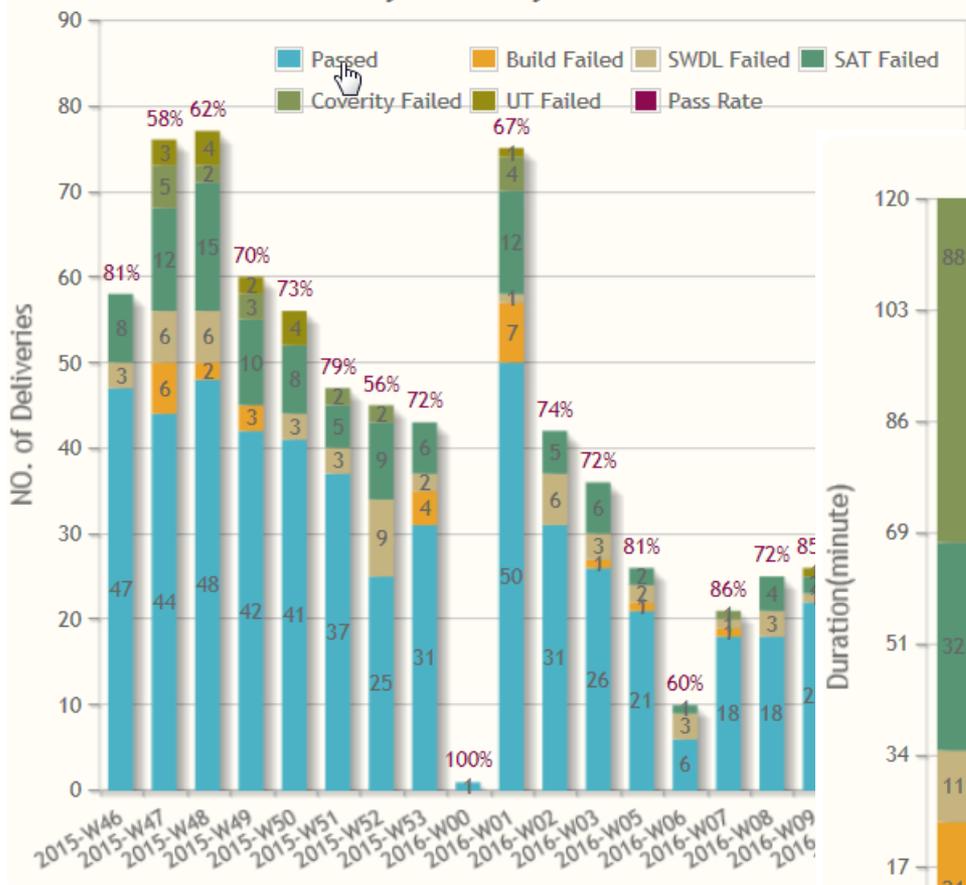
[PoolDetails](#)

Tag	Merge	Build	FUM Code Analysis	P1S Code Analysis	FUM UT	P1S UT	P1S SAT	Push & Rollback
2016-03-24 08:28:52 AM LLV_R1F70_Atlas-001 Atlas	3 m 2 s CAA9013253_1-R1F71	19 m 44 s CXP9027451_1-R1F71_2424_SF-RP1-FUM.ta r.gz CXP9027695_1-R1F71_2424_SF-RP-P1S.t ar.gz	49 m 42 s 0 new defect	46 m 25 s 0 new defect	23 m 27 s 3348 / 3348 Compile errors: 0	20 m 32 s 3366 / 3366 Compile errors: 0	42 m 7 s 10 / 10	10 s
2016-03-23 08:57:30 AM LLV_R1F69_DRAGON-001 DRAGON	2 m 8 s CAA9013253_1-R1F70	22 m 13 s CXP9027451_1-R1F70_2418_SF-RP1-FUM.ta r.gz CXP9027695_1-R1F70_2418_SF-RP-P1S.t ar.gz	1 h 45 s 0 new defect	57 m 27 s 0 new defect	23 m 57 s 3348 / 3348 Compile errors: 0	23 m 12 s 3366 / 3366 Compile errors: 0	42 m 29 s 10 / 10	10 s
2016-03-23 07:21:41 AM LLV_R1F69_Mimas-001 Mimas	2 m 7 s CAA9013253_1-R1F69	20 m 16 s CXP9027451_1-R1F69_2417_SF-RP1-FUM.ta r.gz CXP9027695_1-R1F69_2417_SF-RP-P1S.t ar.gz	50 m 16 s 0 new defect	49 m 51 s 0 new defect	22 m 51 s 3348 / 3348 Compile errors: 0	20 m 53 s 3366 / 3366 Compile errors: 0	42 m 24 s 10 / 10	10 s

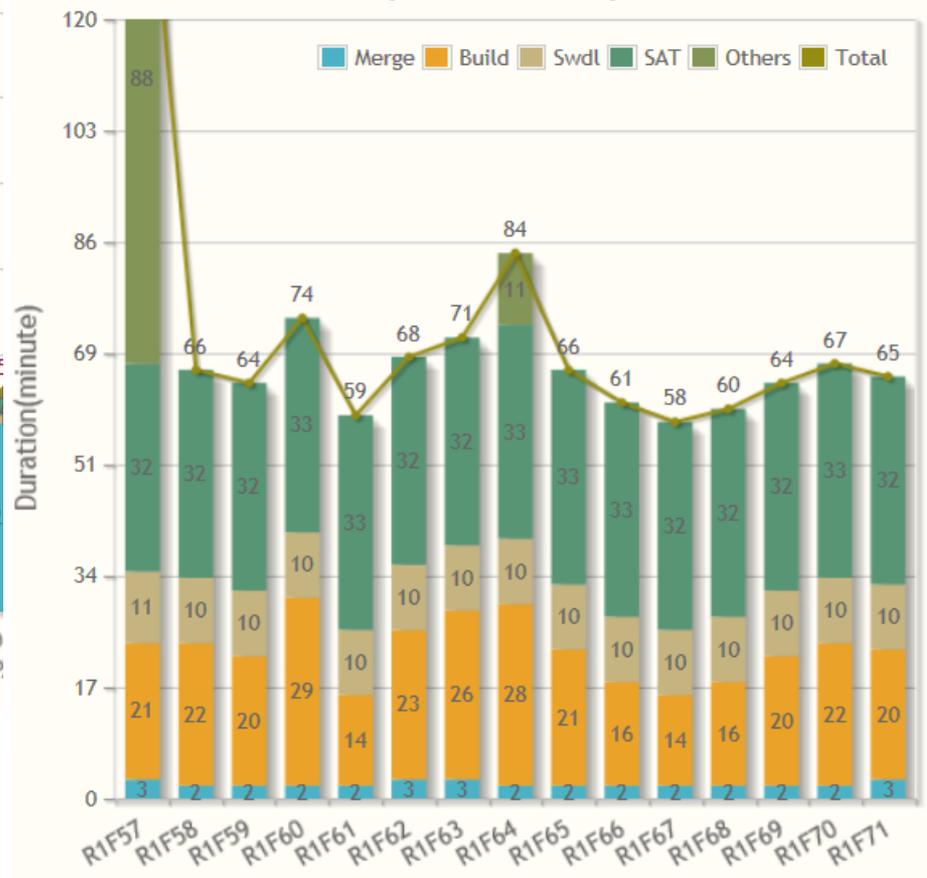
DELIVERY JOB PROCESS

Tag	Merge	Build	FUM UT	P1S UT	P1S SAT
2016-03-24 08:28:52 AM LLV_R1F70_Atlas-001 Atlas	3 m 2 s CAA9013253_1-R1F71	19 m 44 s CXP9027451_1-R1F71_2424_SF-RP1-FUM.ta r.gz CXP9027695_1-R1F71_2424_SF-RP-P1S.t ar.gz	23 m 27 s 3348 / 3348 Compile errors: 0	20 m 32 s 3366 / 3366 Compile errors: 0	42 m 7 s 10 / 10

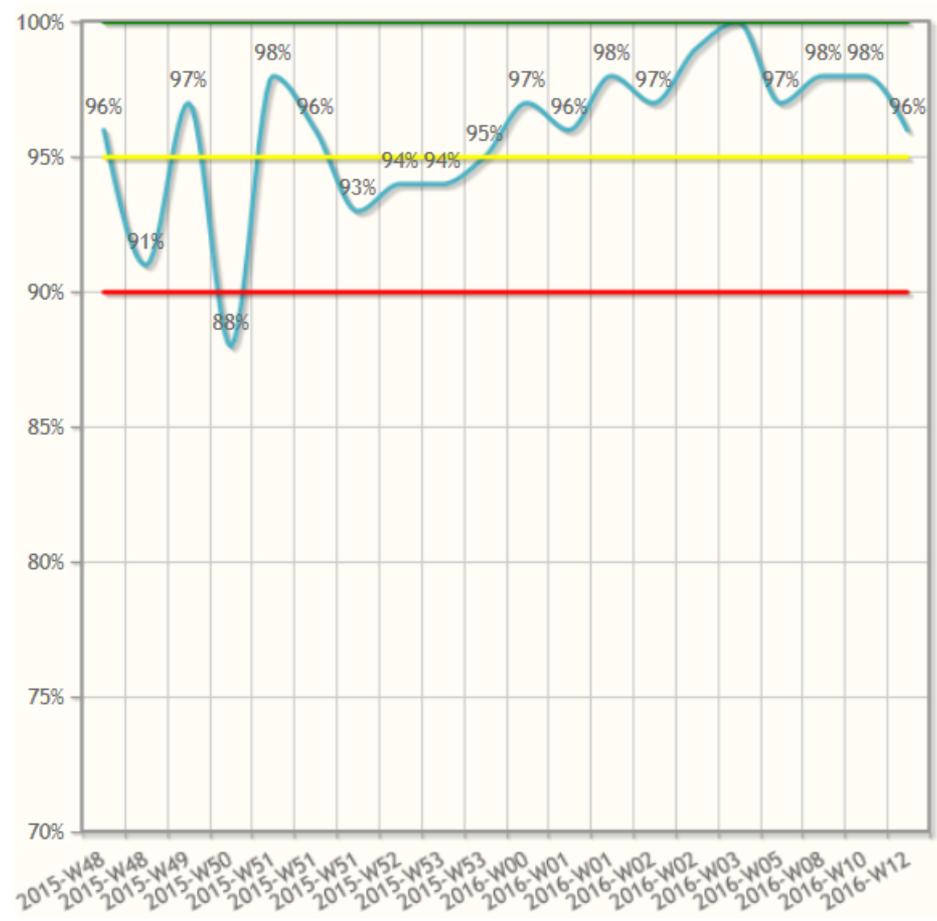
Weekly Delivery Statistics



Delivery Feedback Cycle



REGRESSION [MORE](#)

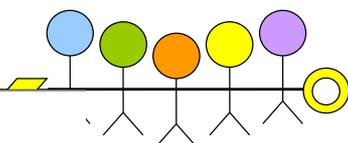


READINESS [TV](#)

Quality	Weekly Regression Pass Rate (passed / total)	A+B Open TR (Internal / IPOS / C-PAAS / BCM / SSR)
2	97.3% (256/263)	5 (3 / 1 / 0 / 1 / 0)
3	98.1% (764/779)	2 (2 / 0 / 0 / 0 / 0)
LS	94.2% (778/826)	3 (1 / 2 / 0 / 0 / 0)
SS	100.0% (160/160)	2 (0 / 1 / 0 / 1 / 0)
ec	97.4% (522/536)	0 (0 / 0 / 0 / 0 / 0)
&OAM	92.2% (599/650)	7 (4 / 3 / 0 / 0 / 0)
OAM	--	0 (0 / 0 / 0 / 0 / 0)
	95.8% (3079/3214)	19 (10 / 7 / 0 / 2)

Weekly Regression Pass Rate	A+B Open TR				
	Functional	Robustness & Stability	Performance & Scalability	Maintainability	
(254/273)	1	7	1	1	
L3VPN	94.3% (200/212)	1	3	1	1
Total	93.6% (454/485)	1	10	2	2

L3VPN	94.3% (200/212)
Total	93.6% (454/485)



Team: Status:

Team	Version	LLV Tag	Delivery Note	Total duration	Tag Marked Time	Processing start time
Atlas	CAA9013253_1-R1F71	LLV_R1F70_Atlas-001		1h 6m 44s	2016-03-24 15:13	2016-03-24 15:28

DRAGON	Code Merging Result	Build Result	Platform	SWDL Result	SAT Result	SAT Pass/Total	UT Result	UT Pass/Total	Code Analysis
		SUCCESS 0h 3m 2s	SUCCESS 0h 19m 44s	Total	SUCCESS 0h 10m 18s	SUCCESS 0h 31m 49s	10/10	SUCCESS 0h 23m 27s	6714/6714
			fum				SUCCESS 0h 23m 27s	3348/3348	SUCCESS 0h 49m 42s
Mimas			p1s	SUCCESS 0h 10m 18s	SUCCESS 0h 31m 49s	10/10	SUCCESS 0h 20m 32s	3366/3366	SUCCESS 0h 46m 26s

		Search: <input type="text"/>			
SUCCESS 0h 2m 8s	SUCCESS 0h 22m 13s	New Defect	Processing end time	Final Status	Failed Reason
		0			Type: <input type="text" value="Select Your Options"/> Comments: <input type="text"/> <input type="button" value="update"/>
		0	2016-03-24 16:35	SUCCESS	<input type="text"/> <input type="button" value="update"/>
		0			
		0			Type: <input type="text" value="Select Your Options"/> Comments: <input type="text"/> <input type="button" value="update"/>
		0	2016-03-23 17:06	SUCCESS	<input type="text"/> <input type="button" value="update"/>
		0			

讲座 4月16日 云之大数据平台架构设计



随时听讲座
每天看新文
——
追随技术信仰