

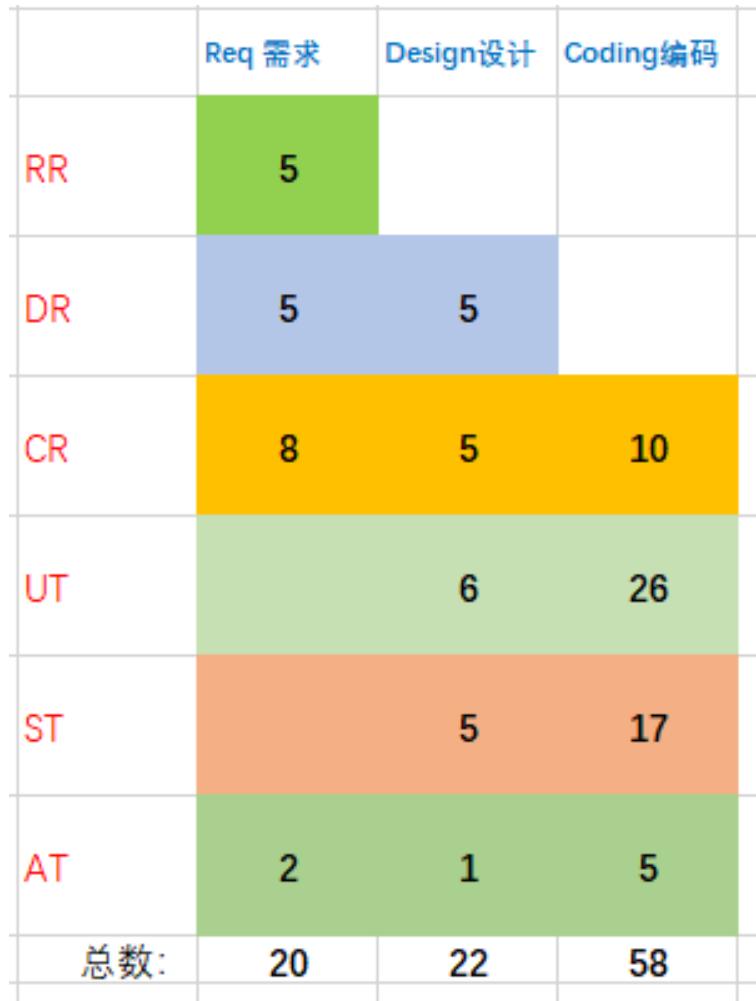
当团队已经掌握好迭代回顾根因分析后，便可以尝试利用蒙地卡罗 (Monte Carlo) 预测模型做定量分析。

下面介绍如何利用蒙地卡罗预测模型帮助团队分析迭代缺陷数据，并预测下个迭代（改进后）的过程缺陷范围，把缺陷管理从以前定性，提升到量化管理。

### 使用缺陷排除率，配合蒙地卡罗预测模型做定量分析

1. 依据客户需求和公司基线与目标，设定下次迭代量化目标。例如希望最终遗漏到验收发现的缺陷数降多少？
2. 并设定过程目标缺陷数，从而预测能否达到最终目标，缺陷越能在前面过程（如需求/设计评审）发现并解决缺陷，后面过程里发现的缺陷会减少（假定每个过程引入的缺陷数没有重大改变）

如果要最终质量好，缺陷排除率就要高。但计算缺陷排除率，必须要等到整个开发完成才可以计算，我们可以建立预测模型，模拟这个过程：需求会引入缺陷，然后需求评审排除缺陷等等。把各引入缺陷数，排除率输入蒙地卡罗预测模型，然后使用它估计每个阶段的缺陷数量范围。



		D.R.E.
需求评审	RR DRE = 5/20	25.00%
设计评审	DR DRE = (5+5) / (20+22-5)	27.03%
代码评审	CR DRE = (8+5+10) / (100-5 - 5 - 5)	27.06%
单元测试	UT DRE = (6+26) / (100-5 - 5 - 5 - 8 - 5 - 10)	51.61%
系统测试	ST DRE = (5+17) / (100-5 - 5 - 5 - 8 - 5 - 10 - 6 - 26)	73.33%
验收测试	AT DRE = (2+1+5) / (100-5 - 5 - 5 - 8 - 5 - 10 - 6 - 26 - 5 - 17)	100.00%

得出下面 Figure7.2，缺陷分布是中间最高头尾低，右面与左面不同，有条长尾巴，类似估算软件开发项目工作量的 Rayleigh 曲线。

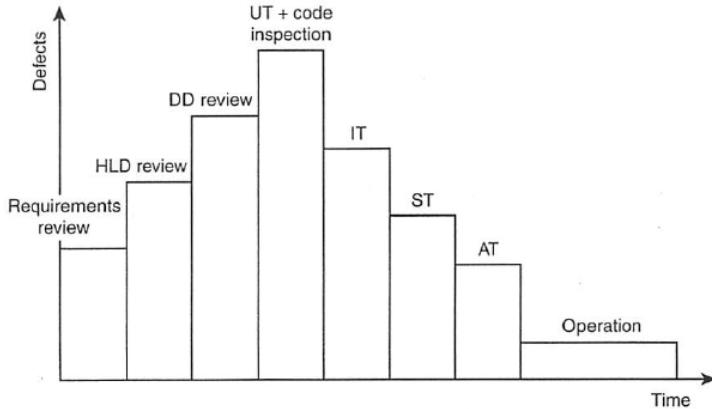


Figure 7.2 Defect detection in a project

模型假定每个阶段的缺陷排除率都比较稳定，在某个范围之内，不同阶段引入的缺陷也在一定范围之内。

### 怎样开始

培训能让团队能先了解利用模拟工具量化管理的原理，也让团队为收集迭代数据做准备，所以建议先做模拟互动培训。下面是一天互动培训的时间安排：（上午是做上一章迭代回顾的互动练习，下午是使用蒙地卡罗预测模型预测下迭代的互动练习。）

500px

### 时间安排

如果是首次根本原因分析培训，建议先在培训当天早上做以下根因分析小组互动练习，让大家先熟悉根因分析的重点：

- 提供一对模拟数据，要求团队利用帕累托图加鱼骨图分析，找出根本原因与改进措施
- 目的：让团队先感受如何基于数据，找出根本原因

### 培训对象

- 后面会参与试点的项目组，培训后可以把学过的在项目里实践
- 所有团队角色都参加

### 准备数据，让团队模拟

要做量化根本原因分析就必须要有数据，所以内部教练须要预先准备下面缺陷与返工工作量数据，类似一轮冲刺后的场景：

1. 系统测试缺陷数据，建议从缺陷管理工具里抽取，40 - 60 项

2. 开发个人系统测试 BUG 返工工时统计表：那个 BUG 号 / 总修复工时 / 备注
3. 开发个人开发期间用于修复评审/单元测试返工工时统计表：那个模块 / 总修复工时 / 备注
- 打印以上的数据表，发给团队做练习时参考

例如，互动练习时，要学员利用开发人员工时表（模拟他们迭代中用于改缺陷的工时，与评审缺陷数，与开发和修正代码的工时）：

系统Bug 修复工时记录汇总		开始	结束	总修复人时	备注
	5384			1.0	初代入修复
	5198			7	老手的修复经验
	5365			5	
	5366			2	
	5203			1	

序号	开发功能	开始	结束	人时		修改 LOC	LOC
				缺陷	备注		
1)	DISPLAY_MAP			6	不能正确绘制		
2)	SORT			9	前后顺序		
3)	LOGIN			3			
4)	MEDIAN			4	计算语音不准确		
5)	RANGE			4			
6)							
7)							
8)							
9)							

### 准备工具

- 分 2-4 组，6-8 人一组，需要有一个大而光亮的会议室，方便小组互动，大白板和架子，各种颜色的水笔，墙壁可以贴上小组的白纸
- 教练提供有 MonteCarlo 工具的笔记本电脑

### 某公司经过培训后第一轮回顾例子

- 团队分析迭代缺陷数据，缺陷是源自需求/设计/编码，得出以下分布表：
- : \_20220316093044.jpg
- 按 DRE 公式，计算出各个过程的缺陷排除率

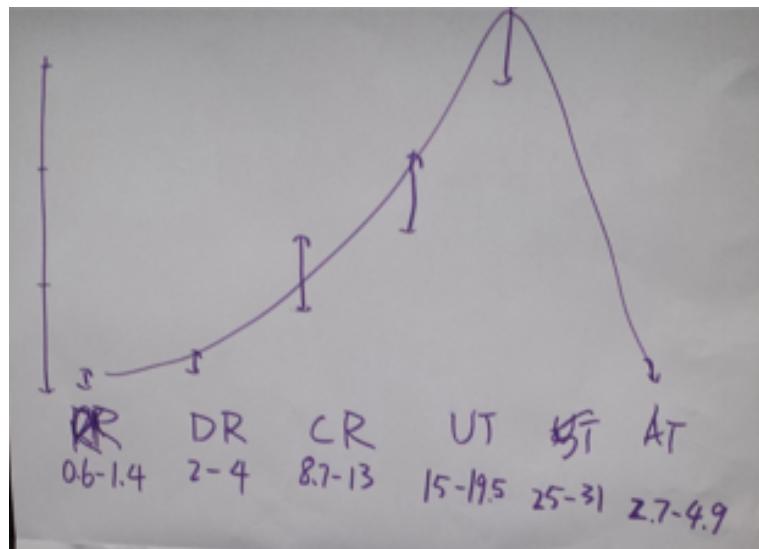
RR	$1/10$	$10\%$
DR	$1+2/(10+20-1)$	$10.3\%$
CR	$2+3+6/(64-4)$	$18.3\%$
VT	$2+5+10/(64-15)$	$24.7\%$
ST	$3+9+16/(64-32)$	$37.5\%$
AT	$1+1+2/(64-60)$	$100\%$

- 从团队工时表估算各过程的缺陷返工工作量

		修复缺陷总工时
RR+DR 15%	4	$16 \text{ h}/4$
CR+VT 55%	2.9	<del><math>80 \text{ h}/28</math></del>
ST 22%	5.8	$160 \text{ h}/28$
AT 8%	20	$80 \text{ h}/4$

- 把缺陷排除率与返工工作量输入预测模型，得出每个过程的缺陷范围与质

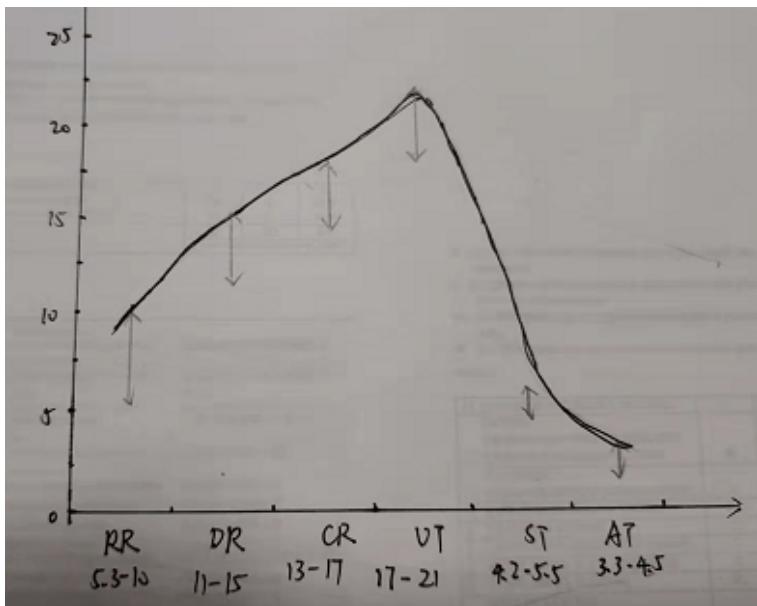
量成本，也验证了是系统测试缺陷率最高



- 团队针对评审排除率低（如需求/设计评审）、利用新的技术和工具，例如用原型法和新评估方法，能提高需求与设计的缺陷排除率。把新方法的预估排除率加进预测模型，然后使用模型比较各种方法，寻找那个搭配的返工工作量最低，模型能预测使用新方法后的各个过程缺陷范围与质量成本。

如果没有预测模型，我们只能主观估计百分比会降低，但有了预测模型，我们确实可以尝试改变参数，看能否估计缺陷分布的变化，也帮我们验证这假定是否可行。最终我们验证了确实这个百分比是可行。如果没有预测模型便难以做到。

例如，延续前面例子，下面看看团队如何基于练习 2 的数据，加入新方法/新参数，利用模型预估下一轮缺陷分布：



可看到大部分缺陷都前移，系统测试缺陷下降。

## 总结

当数据不是单点而是分布时，蒙特卡罗预测模型可以帮我们更好处理数据。针对如何可以把发现缺陷前移的例子里，模型可以从每个过程的分布预估总返工工作量成本分布，也可以比较不同的搭配，自动挑选哪个搭配的质量成本最低。要做好回顾分析，最困难不是数据分析，而是怎么收集到真实的数据。所以必须在培训里，让学员知道为什么要收集数据，为迭代收集数据做好准备。

整个迭代回顾过程：要做好迭代，必须所有干系人都参与，并要花大概3、4个小时的精力分析、讨论改进行动，所以必须以二八原则挑选最容易出效果的方案，才容易得到高层的支持，也需要培训团队各种分析技巧和原则。

如果希望干系人有执行改进行动的动力，必须要他们在回顾时全心投入参与讨论，一起找根本原因和解决方案。所以培训时，不仅仅是教分析的技巧，更需要让他们可以放心发表意见。如果把培训设计成互动游戏，可避免迭代回顾时，项目经理一言堂的现象。

分析的方法和可以改进的方向很多，这几个章节主要以一些实例带出每一个步骤的重点。大家掌握了这些节奏之后，就可以融会贯通，持续进行根本原因分析，形成不断优化的良性循环。目标是提升产品的质量，而不是习惯于当前的缺陷水平，缺乏改进动力。要做好根本原因分析，收集到正确有用的数据非常关键。下一部分我们会进一步探索度量与分析的重点。

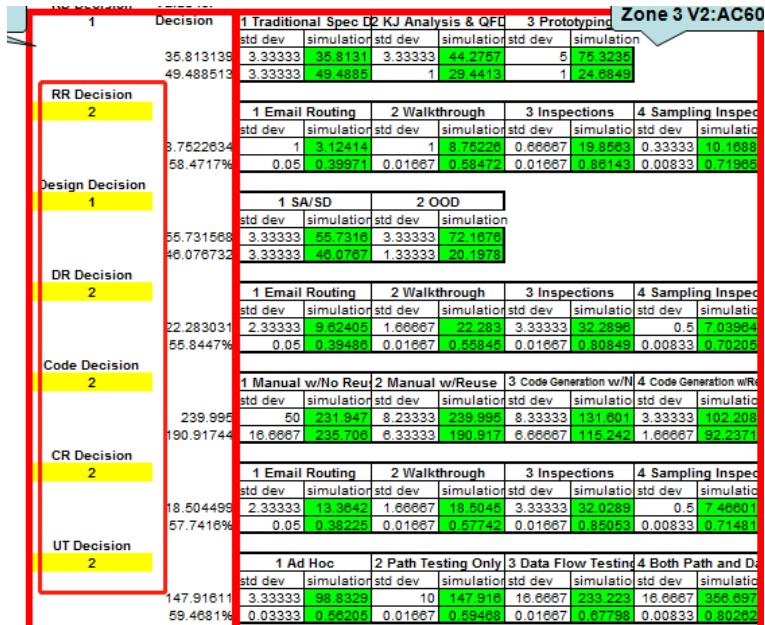
## 附件

### 水晶球蒙地卡罗预测模型

- 按每个过程的统计数据，估计对应缺陷排除百分比的上下限与均值，工作量也类似。例如，从公司基线，代码走查的缺陷排除率是 55% 左右；评审总工作量（不包含修正缺陷）大概为 20 人时。如果开始时没有充足的项目数据，可以先依据行业基线：如代码正式审查应能排除 85%，但平均人时会增加到 35 人时（也是不包含修正缺陷）。得出下图：

Zone 1 V2:K6U				Zone 2 T2:U						
Requirements Development	读读+检错			展示现有相关产品			高保真原型			
	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL	
	25	35	45	35	45	55	65	80	95	
需求评审 Reqs Review	工时 Effort	45	55	27	30	33	22	25	28	
	质量 Quality	25.00%	40.00%	55.00%	50.00%	55.00%	60.00%	30.00%	35.00%	
设计设计 Design Design	邮件评审 Email Routing	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	1	4	7	7	10	13	18	20	22
	质量 Quality	25.00%	40.00%	55.00%	50.00%	55.00%	60.00%	30.00%	35.00%	35.00%
设计评审 Design Review	合称概要与详细设计	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	50	65	70	75	85	95	45	55	65
	质量 Quality	25.00%	40.00%	55.00%	50.00%	55.00%	60.00%	30.00%	35.00%	35.00%
设计评审 Design Review	邮件评审 Email Routing	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	5	12	19	15	20	25	25	35	45
	质量 Quality	25.00%	40.00%	55.00%	50.00%	55.00%	60.00%	30.00%	35.00%	35.00%
代码代码 Code Code	少或多重复用 Manual w/o Reuse	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	150	200	245	222.3	247	271.7	100	125	150
	质量 Quality	200	250	300	171	190	209	80	110	130
代码评审 Code Review	邮件评审 Email Routing	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	5	12	19	15	20	25	25	35	45
	质量 Quality	25.00%	40.00%	55.00%	50.00%	55.00%	60.00%	30.00%	35.00%	35.00%
单元测试 Unit Test	邮件评审 Email Routing	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	90	100	110	120	150	180	200	250	300
	质量 Quality	40.00%	50.00%	60.00%	55.00%	60.00%	65.00%	65.00%	70.00%	75.00%
单元测试 Unit Test	临时手工 Ad Hoc	LL	Avg	UL	LL	Avg	UL	LL	Avg	UL
	工时 Effort	90	100	110	120	150	180	200	250	300
	质量 Quality	40.00%	50.00%	60.00%	55.00%	60.00%	65.00%	65.00%	70.00%	75.00%

- 模型会自动按输入的最大最小值计算标准差，变成分布（如：正态 (normal) 分布、三角形 (triangular) 分布），水晶球模型会依据黄格里的数字选取对应的分布，



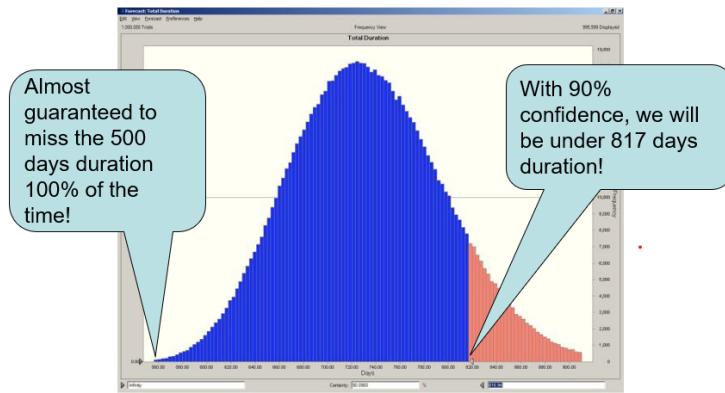
- 取对应的分布，依据项目历史数据统计，估算各个过程缺陷的返工工作量与成本。例如：修复一个系统测试发现的缺陷，平均要用 6000 元，最高 7500，最低 5000。如果缺陷导致的是工时那行，填上对应工种的平均人时总成本。水晶球模型就会依据我们输入的范围估算单位成本的分布，放在绿格里。

	Costs	Unit Costs	Min	Most Likely	Max
Code					
Effort	\$226,846	\$901	\$700	\$800	\$1,000
Quality					
Code Review	Costs	Unit Costs	Min	Most Likely	Max
Effort	\$17,917	\$922	\$700	\$800	\$1,000
Quality	\$219,686	\$1,925	\$1,800	\$2,000	\$2,300
Unit Test	Costs	Unit Costs	Min	Most Likely	Max
Effort	\$280,774	\$800	\$700	\$800	\$1,000
Quality	\$175,983	\$2,422	\$2,300	\$2,400	\$2,600
Integration Test	Costs	Unit Costs	Min	Most Likely	Max
Effort	\$1	\$842	\$700	\$800	\$1,000
Quality	\$5	\$3,973	\$3,000	\$3,500	\$4,500
System Test	Costs	Unit Costs	Min	Most Likely	Max
Effort	\$64,307	\$785	\$500	\$600	\$750
Quality	\$113,610	\$8,021	\$6,800	\$8,000	\$9,200
Acceptance Test	Costs	Unit Costs	Min	Most Likely	Max
Effort	\$23,336	\$768	\$700	\$800	\$1,000
Quality	\$49,906	\$16,506	\$12,000	\$15,000	\$18,000

- 如果从需求到验收测试，每一个过程的成本数都是单一值；我们可以简单用 XLS 把数加起来得出总数。但如果每个过程的成本数都是一个分布，例如三角形分布，

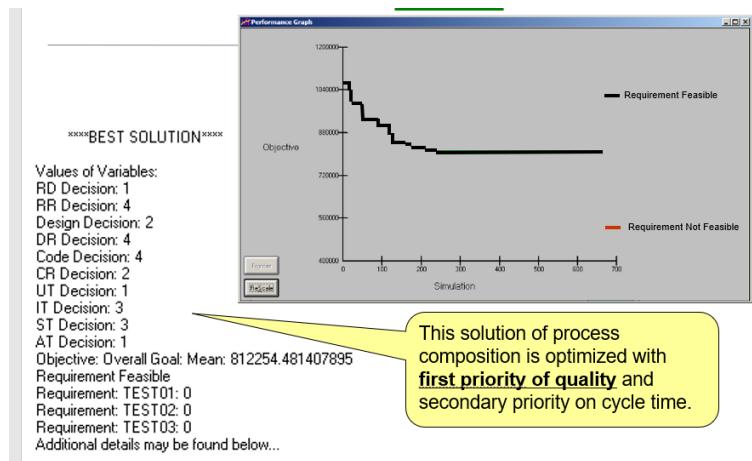
必须要使用蒙地卡罗方法才可以把每个过程“加”起来。

- 水晶球可以帮我们估计质量成本的分布，例如我们需求评审用审查（2），设计评审用走查（2），代码评审也用走查（2），单元测试是手工（1），系统测试估计走 3 轮（2），验收测试预计走 3 轮（2），水晶球就会依据我们刚刚输入的分布用蒙地卡罗预测模型预估质量成本的分布。

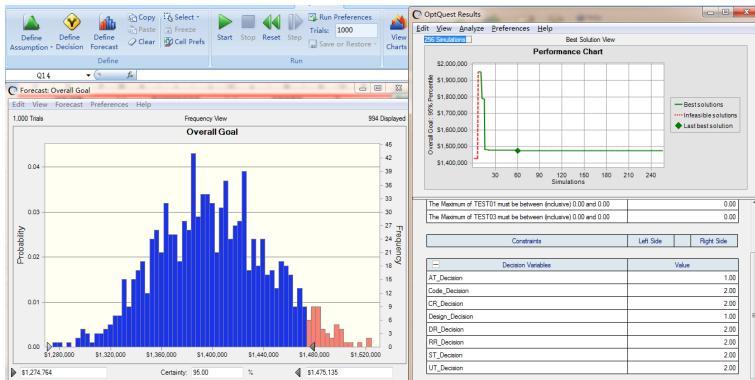


- 比较各种不同的配搭，选出质量成本最低是哪个配搭组合，帮助项目组在策划时选择使用什么方法，得出最佳效果。

下图显示一直水晶球优化的过程，和最终的最“佳”配搭：



在需求评审，设计评审，代码评审，单元测试，系统测试，验收测试都有两种选择时，跑优化，95% 的置信区间得出 Overall goal (Quality + Effort) = 1476773



===== <<< END >>> =====

缺陷分布表			DRE缺陷消除率		
	Req	Design	Coding		
RR	2			RR	2/3 66.67%
DR	0	1		DR	1/(3+3-2) 25%
UT	1	0	4	UT	(1+4)/(12-3) 55.56%
IT	0	0	0	IT	0 0%
SIT	0	0	2	SIT	2/(12-8) 50%
UAT	0	2	0	UAT	2/(12-10) 100%
	3	3	6		

