



北京航空航天大学
BEIHANG UNIVERSITY

软件过程管理讲座

第一讲：过程性能数据分析

吴超英

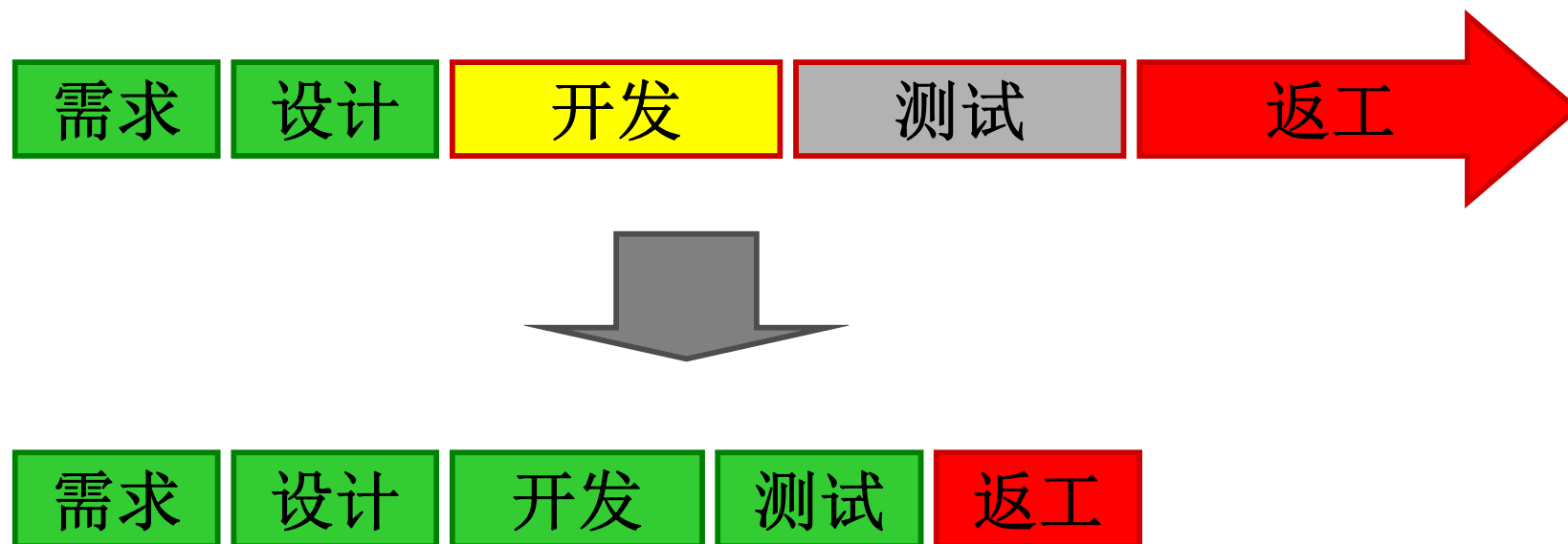
2016年10月

版权所有 请勿翻印



北京航空航天大学
BEIHANG UNIVERSITY

数据为公司了解到哪些重要的信息？



缩短的测试周期

降低的返工成本

提高的生产率

谁应该关注组织级过程性能数据

- 对组织和多个项目整体工作绩效负责的人
 - 高层管理
 - 部门主要负责人
 - 程序经理/产品经理/项目经理
 - 个体工程师
- 对组织的度量系统负责的人
 - 过程组成员
 - 度量组成员

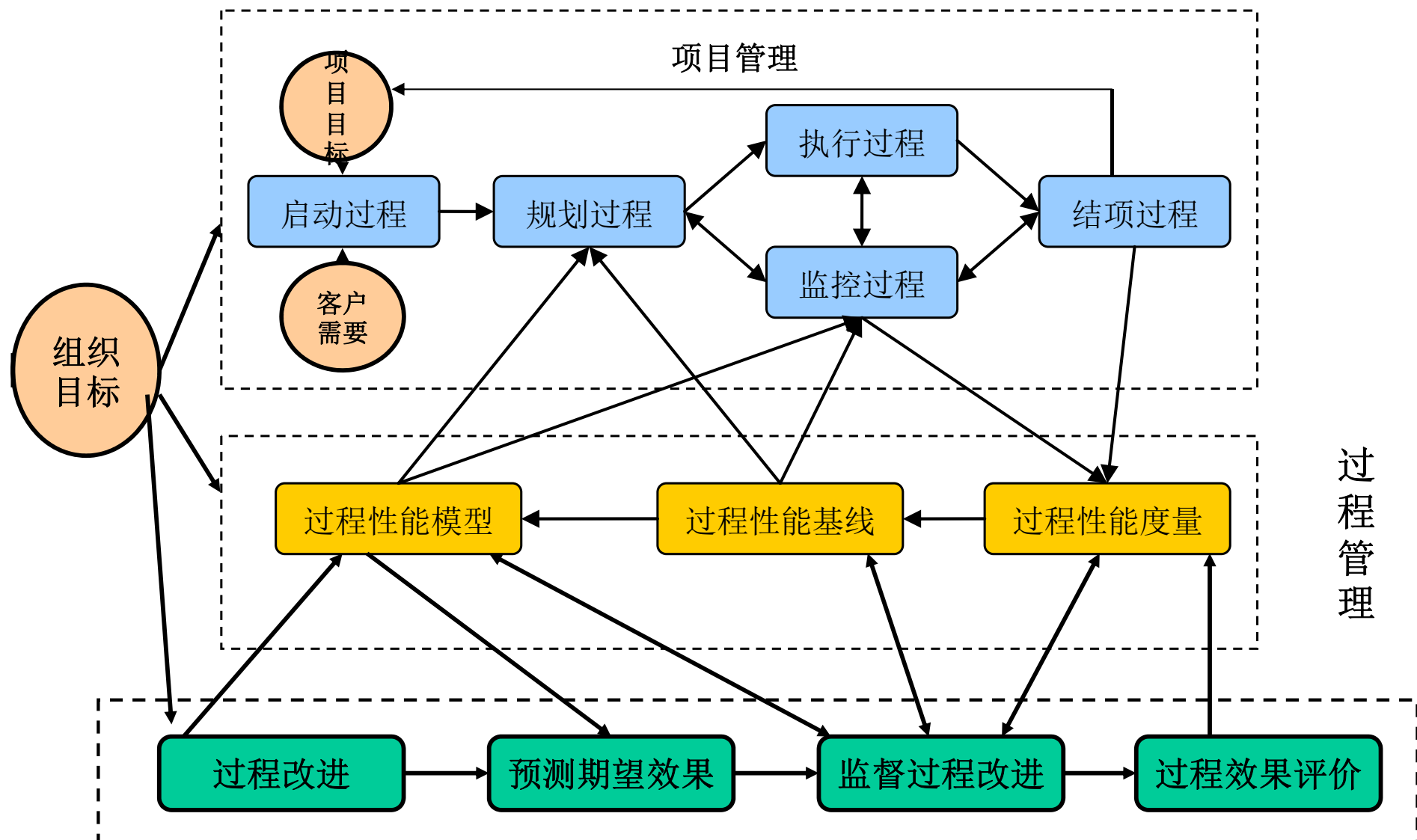


没有数据，无法了解问题的影响因素

- 确定项目的承诺
 - 用历史数据评价这些项目（如业务用例）
 - **75%** 的IT项目管理者失败的原因：
 - 其职责与项目/公司盈利无关
 - 丧失业务目标驱动
- 项目执行的质量
 - 每个项目为组织性能的结果的贡献是多少？
 - 如项目关键成功因素



高成熟度过程改进概览



关键概念

- 以业务目标为主线梳理组织级过程
- 全面分析影响因素
- 建立客观、有效的过程绩效管理
- 刻化客观因素影响，以去除特殊因素影响
- 求得过程改进的投资回报

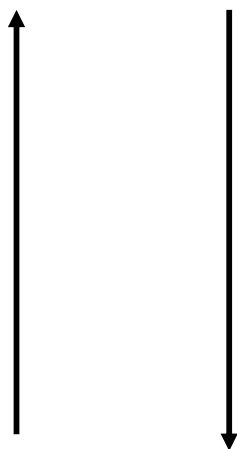
下面讨论

- 几个关键的概念
- 初步实施指导

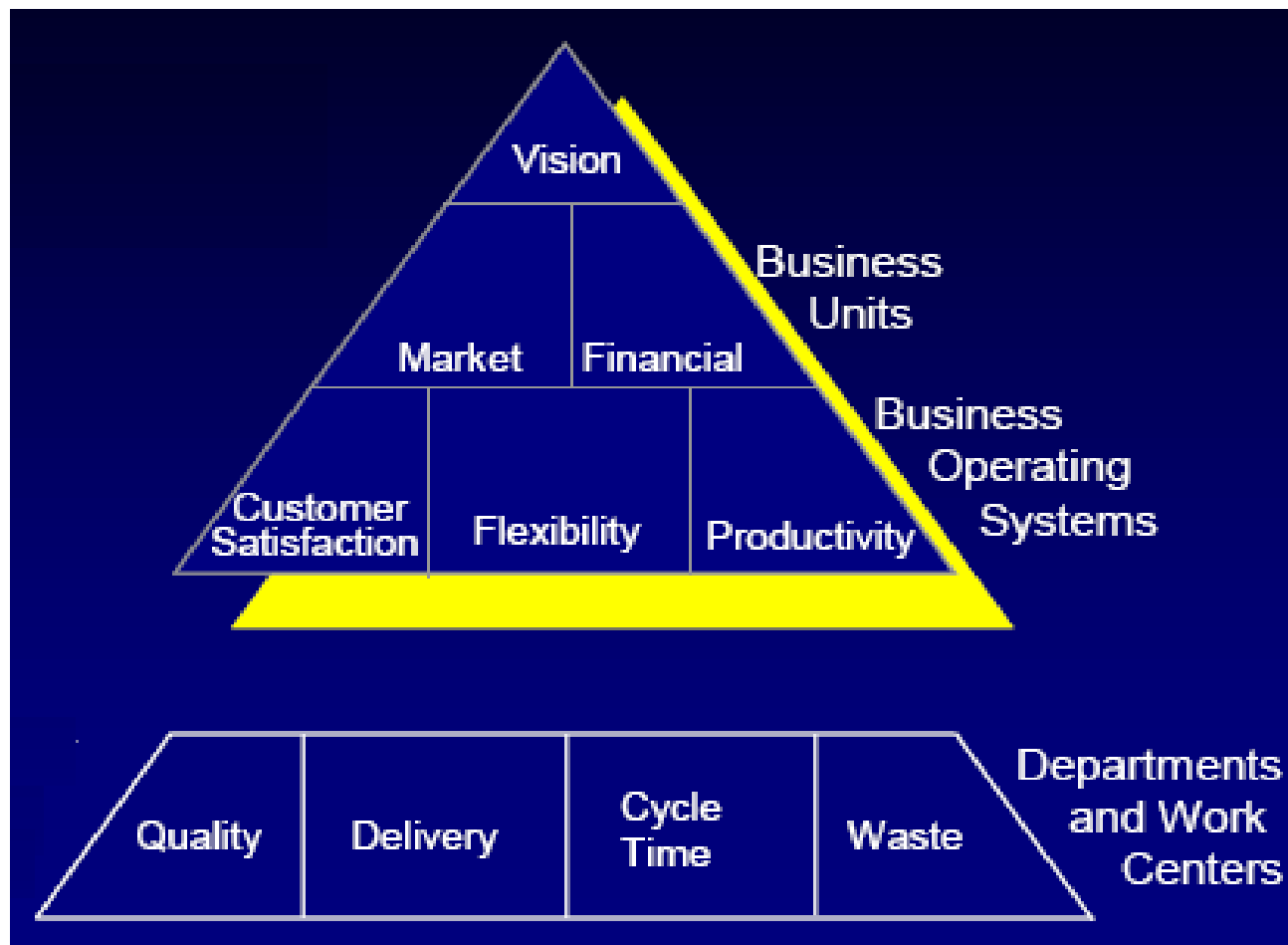


数据与公司业务

信息用于评估性能和
指导改进



由工作过程和事务处理
所生成的数据

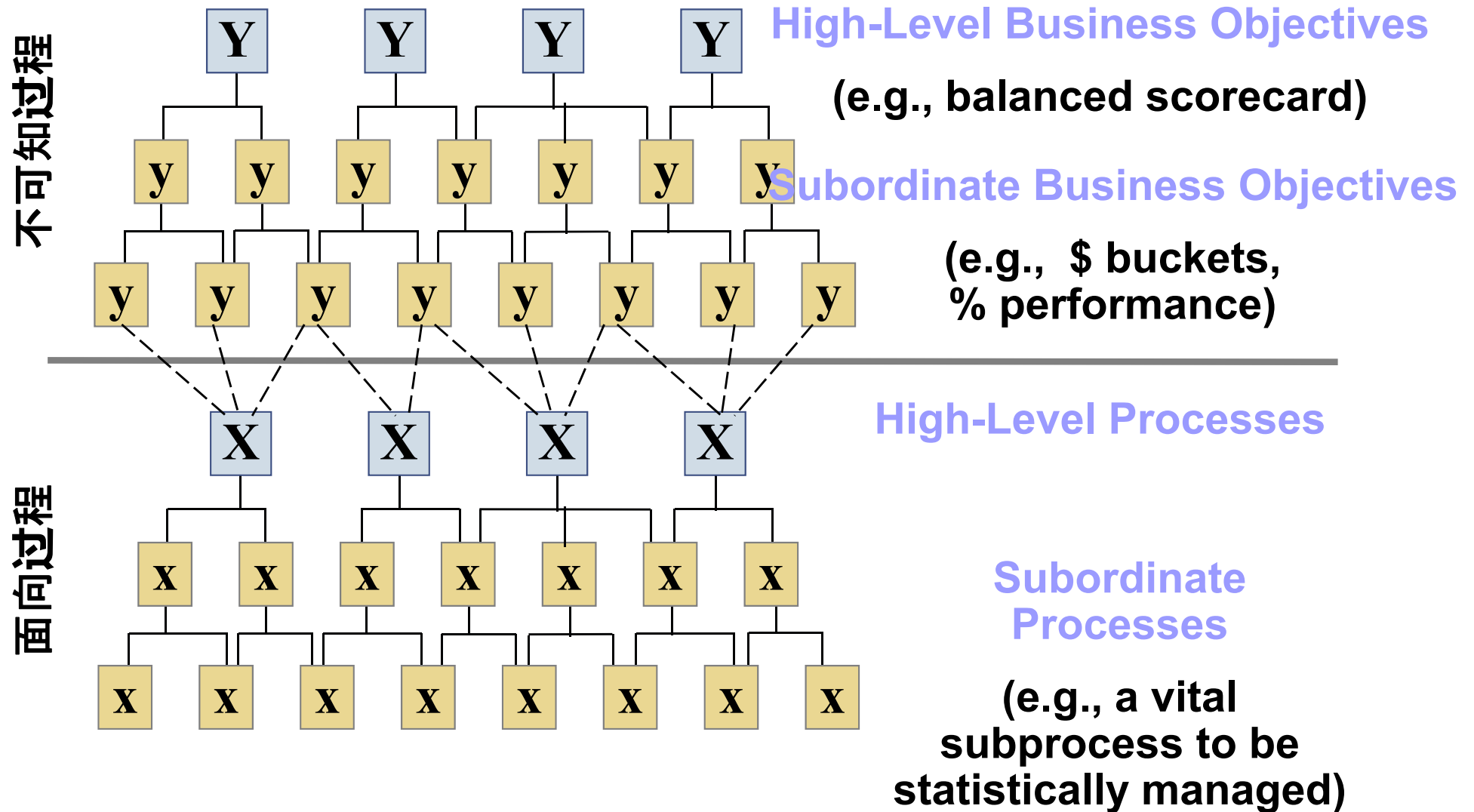


目标分解概念

- 关键的业务和组织目标是什么？
- 组织中关键的过程和子过程是什么？
- 最影响每个目标的前**2-3**个过程和子过程是什么？
- 哪些过程或子过程应该有一个**SMART**项目目标陈述？
- 哪些也应该被统计管理？
- **SMART:**
 - **Specific :** Clear and concise(简明) statement of what will be accomplished
 - **Measurable:** Concrete实体, observable statement of what will be different once the objective is achieved
 - **Attainable:** Feasible in terms of time, cost, and the degree of “stretch” the objective presents (可达到的)
 - **Relevant:** Tied to a specific element of an organization’s objectives



Y to X目标与过程分解方法



如何选择过程和子过程

- 一个项目的生命周期可以被视为一系列的相互连接的过程和子过程。
- 每个过程和子过程均具有其自己的能力、产生其成果、生产率、重要度、缺陷注入率和缺陷排除率等
- 在每个生命周期阶段内，所选择的子过程在达成质量和过程性能方面起着关键（ **critical** ）的作用
- 选择子过程的依据
 - 要覆盖生命周期阶段
 - 具有对达到目标能够提供观察和贡献的能力



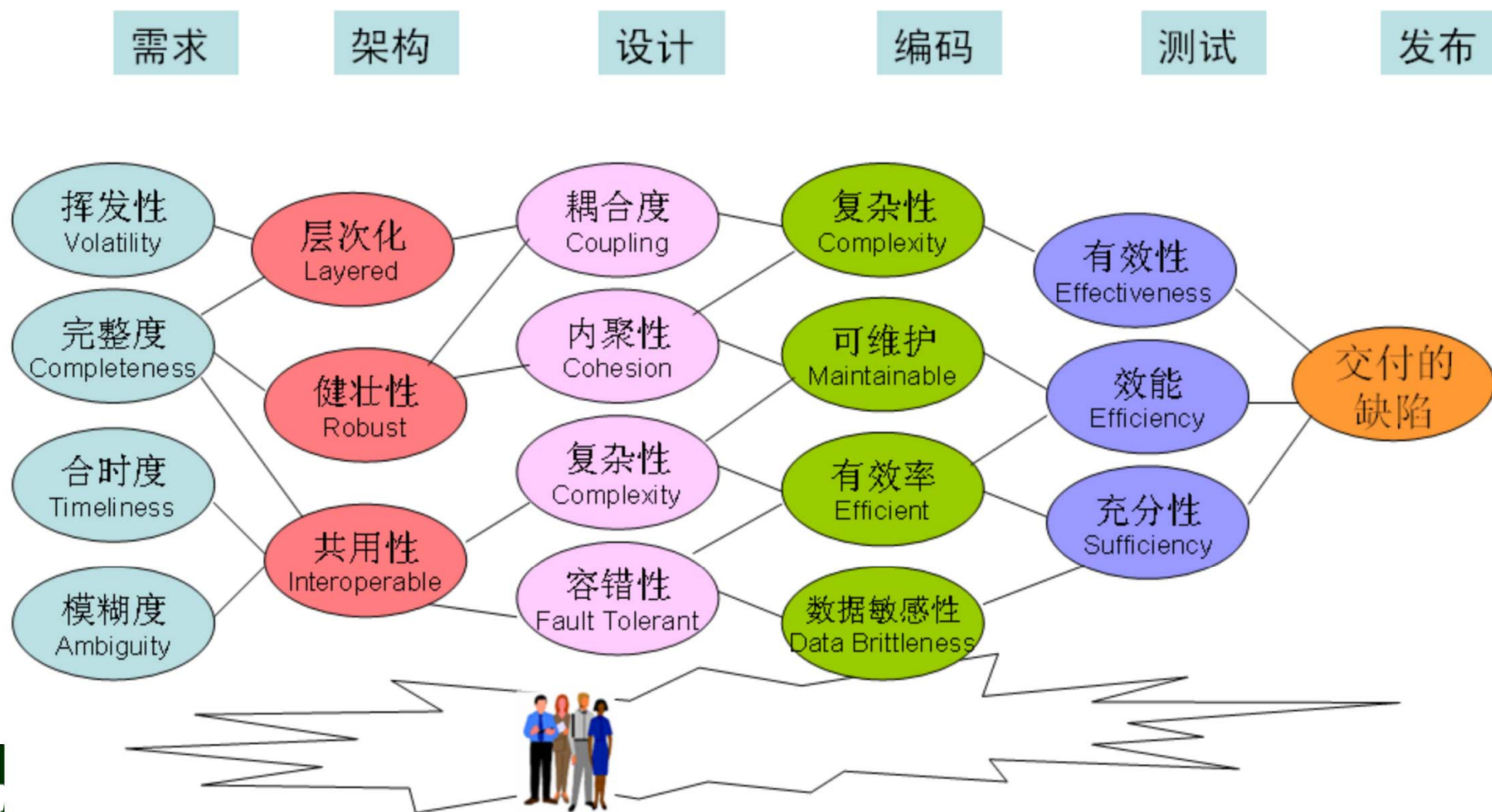
选择过程和子过程误区



子过程不是随意选择，也不是
基于已有数据而选择的

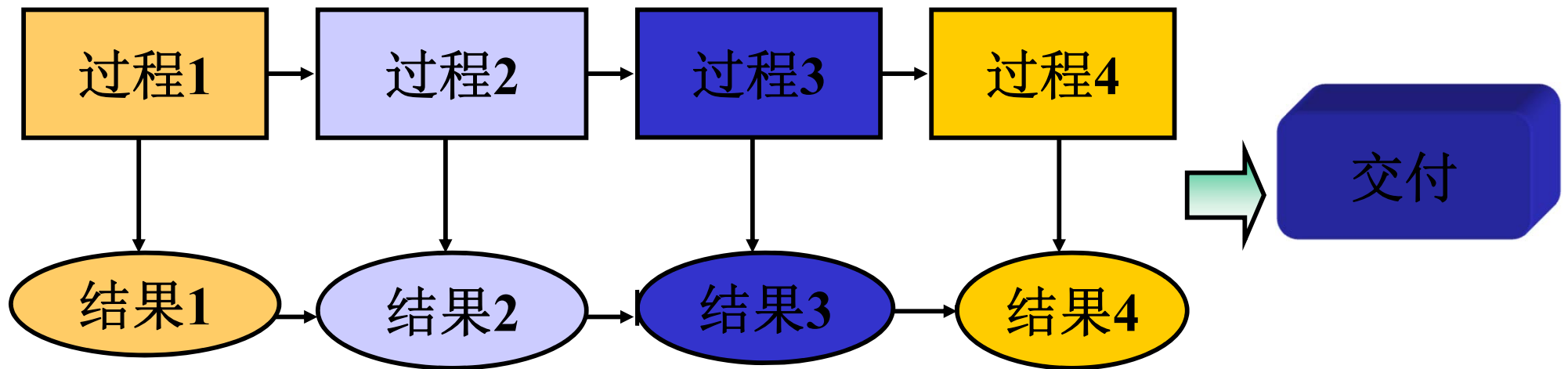
在开发过程中阶段内的关键因素

使用贝叶斯信任网络 (Bayesian Belief Network), 在开发的每个阶段内, 对每个子过程识别关键因素



什么是过程性能的度量

- 过程性能的度量反映了遵循当前过程所达到的实际结果
- 过程性能的度量包括两类度量:
 - 过程度量项（例如工作量、进度周期、缺陷移除率等）
 - 产品度量项（规模、可靠性和缺陷密度等）



在生命周期各阶段识别目标

瀑布模型
示例

阶段	目标1	目标2	目标3	目标4
需求		Y		
设计	Y		Y	Y
实现			Y	
测试	Y		Y	

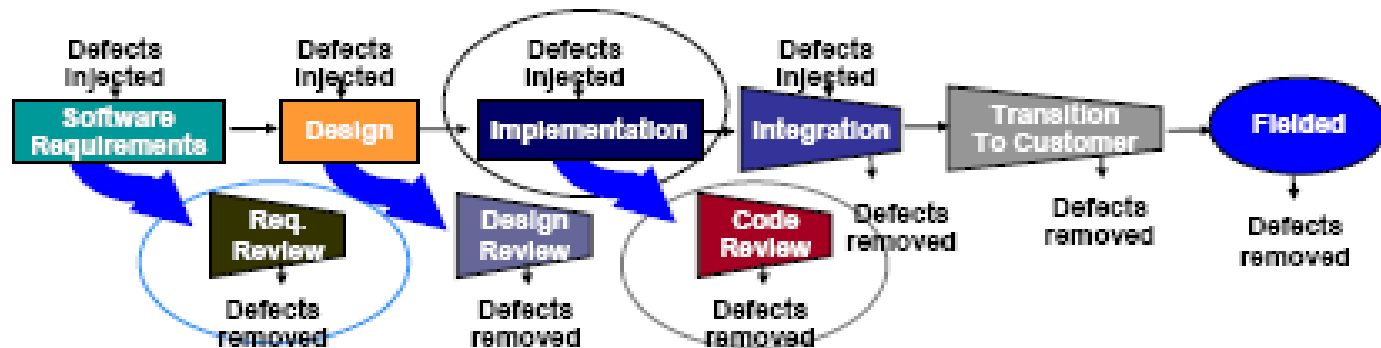
迭代模型
示例

阶段	目标1	目标2	目标3	目标4
需求分析	Y	Y		
迭代开发1	Y		Y	Y
迭代开发2		Y	Y	
迭代开发3		Y		
迭代开发n		Y		
系统测试	Y		Y	

设计过程性能度量示例

- 瀑布模型示例

阶段	过程/子过程	目标1		目标2		目标3		目标4	
		度量	过程性能基线	度量	过程性能基线	度量	过程性能基线	度量	过程性能基线
需求	需求								
设计	设计								
	设计评审								
	原型								
实现	编码								
	代码走查								
	单元测试								
测试	集成测试								
	系统测试								
	验收测试								



考虑对过程性能影响-举例

- 应用领域
 - 航空；金融； ...
- 开发环境
 - **C++;Java;Multi OS;Windows,...**
- 架构类型
 - **Web Service; Embedded; Distributed; Stand Alone,...**
- 客户类型
 - 外部客户：
 - 对日/对美外包
 - 国内开发
 - 内部客户
- 人员技能



定义度量项-举例

度量（需求阶段）	公式	单位	描述
需求规格的规模	—	页	有效的需求文档页数
用例数	—	个	需求规格中包含的用例数
需求挥发性	变更用例数/总用例数*100%	—	概要设计开始后变更用例数占总用例数的百分比
需求开发工作量	—	人时	需求阶段花在需求开发的工作量
需求评审工作量	—	人时	需求阶段花在各次需求评审的工作量的汇总
需求引入缺陷数	—	个	需求开发引入的缺陷数
需求缺陷引入率	需求引入缺陷数/需求开发工作量	Def/人时	需求开发工作中单位人时引入的缺陷数
需求排除缺陷数	Σ 需求评审排除缺陷数	个	需求阶段各次需求评审排除缺陷数的汇总
需求排除缺陷密度	需求排除缺陷数/需求规格的规模	Def/页	需求阶段排除缺陷的密度
需求缺陷排除有效性	需求排除缺陷数/需求引入缺陷数*100%	—	需求阶段收益
需求泄漏缺陷数	需求引入缺陷数-需求排除缺陷数	个	需求阶段泄漏到概要设计阶段的缺陷数

建立需求阶段缺陷引入率过程性能基线

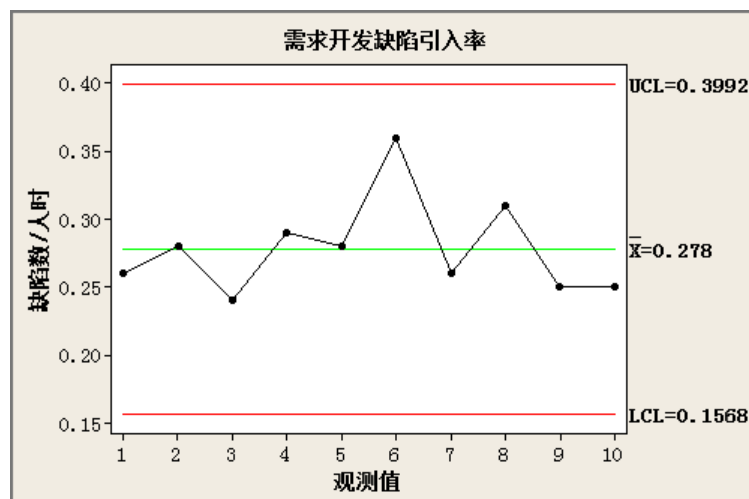
- 某公司需求阶段度量数据

项目编号	需求开发工作量（人时）	需求规格的规模（页）	需求开发引入缺陷数（个）	需求评审排除缺陷数（个）	需求开发缺陷引入率	需求评审缺陷排除有效性
Prj1	172	36	44	28	0.26	0.63
Prj2	198	45	55	42	0.28	0.76
Prj3	169	38	40	32	0.24	0.79
Prj4	106	27	30	21	0.29	0.70
Prj5	127	29	35	25	0.28	0.71
Prj6	129	33	46	35	0.36	0.75
Prj7	137	34	35	23	0.26	0.65
Prj8	182	41	56	44	0.31	0.79
Prj9	137	36	34	25	0.25	0.75
Prj10	196	43	49	33	0.25	0.68



建立需求阶段缺陷引入率过程性能基线

- 需求开发缺陷引入率控制图



- 过程稳定性分析：过程是稳定的，没有发现特殊原因
- 需求开发缺陷引入率的过程性能基线：
 - **{UCL: 0.40, CL: 0.28, LCL: 0.16}**，单位为缺陷数/人时

相关性分析

- 实践中发现带有较强的相似性的数据集
- 相关性是两组变量相关联程度的度量。
 - 如果这两组变量高度相关，并且它们又是相互影响、相互引发的关系，我们可以用一组变量的值来预测和控制另一组变量。
 - 为了得到相关性，变量必须是成对的。也就是说，对每一个 X 的值 X_i 必须有一个对应的 Y 值 Y_i 。



一个相关性的例子

10个PASCAL程序的新建的LOC、修改的LOC和开发时间

Item Number	Total LOC	Development Hours			
n	x	y	x^2	xy	y^2
1	186	15.0	34,596	2,790.0	225.00
2	699	69.9	488,601	48,860.1	4,886.01
3	132	6.5	17,424	858.0	42.25
4	272	22.4	73,984	6,092.8	501.76
5	291	28.4	84,681	8,264.4	806.56
6	331	65.9	109,561	21,812.9	4,342.81
7	199	19.4	39,601	3,860.6	376.36
8	1890	198.7	3,572,100	375,543.0	39,481.69
9	788	38.8	620,944	30,574.4	1,505.44
10	1601	138.2	2,563,201	221,258.2	19,099.24
		603.2	7,604,693	719,914.4	71,267.12



相关性计算

- **x**与**y** 之间的相关性可以用下面的方程计算:

$$r(x,y)=n$$

$$r(x,y) = \frac{\sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{\sqrt{\left[\sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i \right)^2 \right] \left[\sum_{i=1}^n y_i^2 - \left(\sum_{i=1}^n y_i \right)^2 \right]}}$$

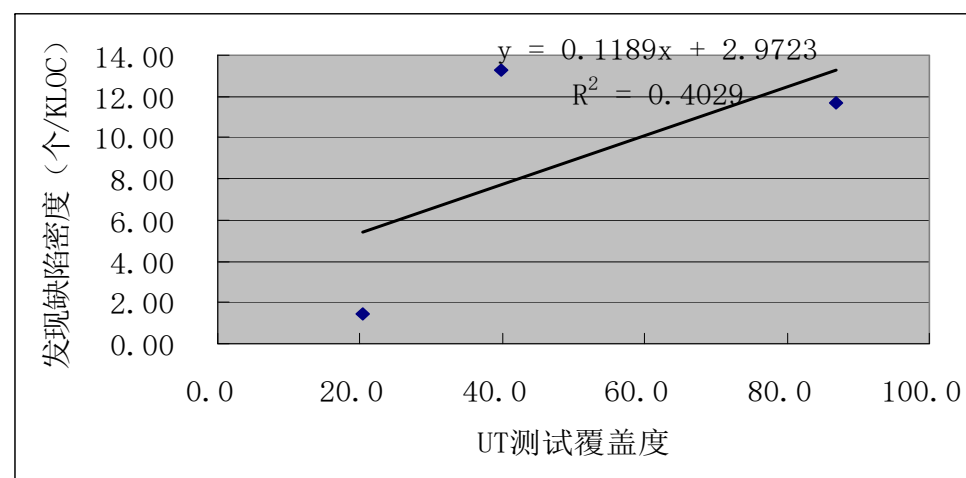
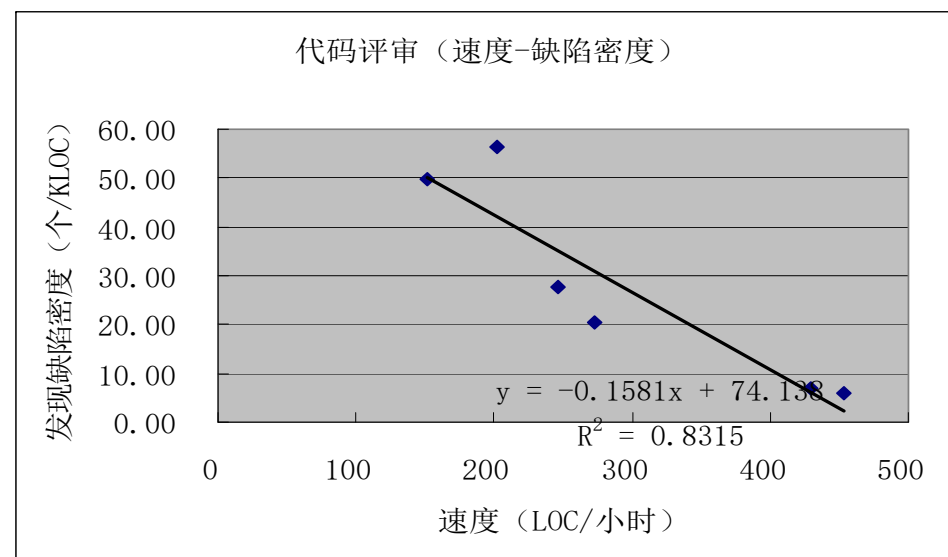
$$r(x,y) = \frac{10 \times 719,914.4 - 6,389 \times 603.2}{\sqrt{(10 \times 7,604,693 - 6389^2) \times (10 \times 71,267.12 - 603.2^2)}} = 0.9543158$$

$$[r(x,y)]^2 = 0.9107$$

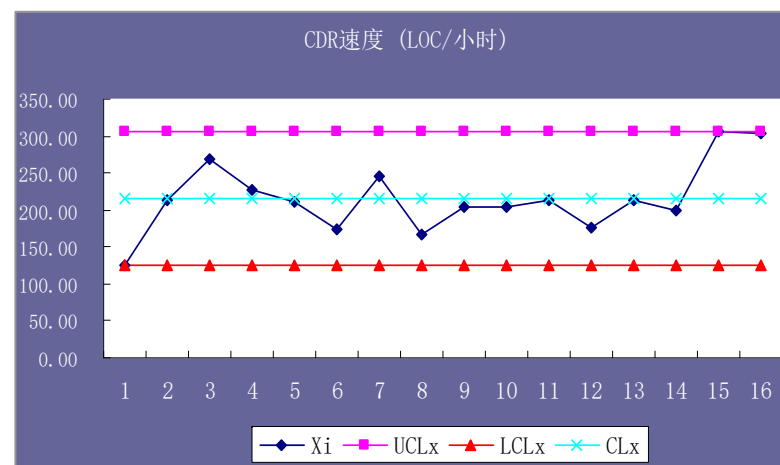
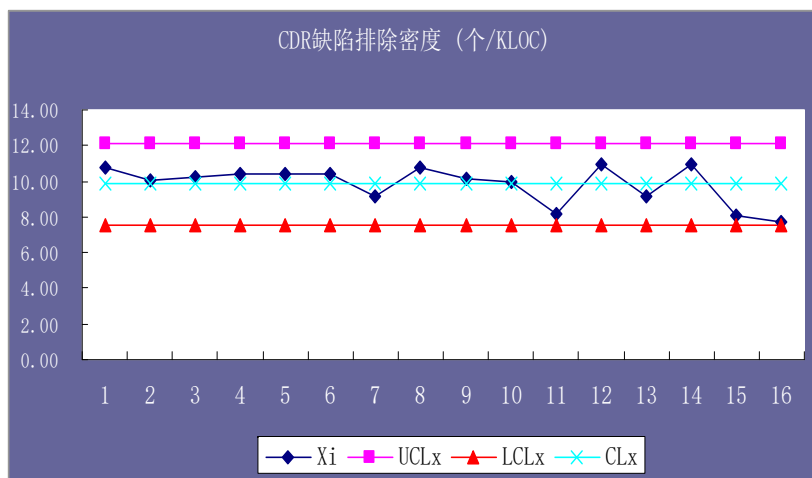
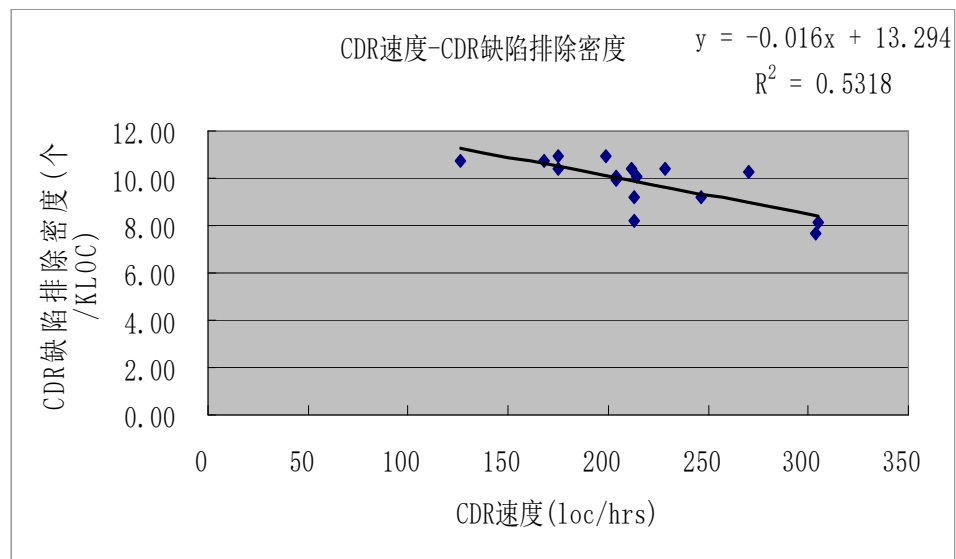
相关性大吗?

编码阶段的因素分析

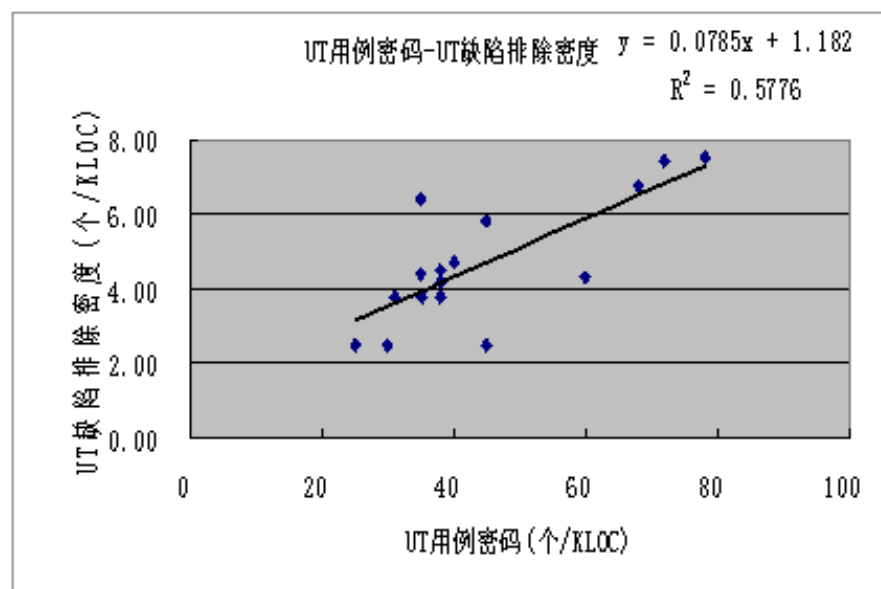
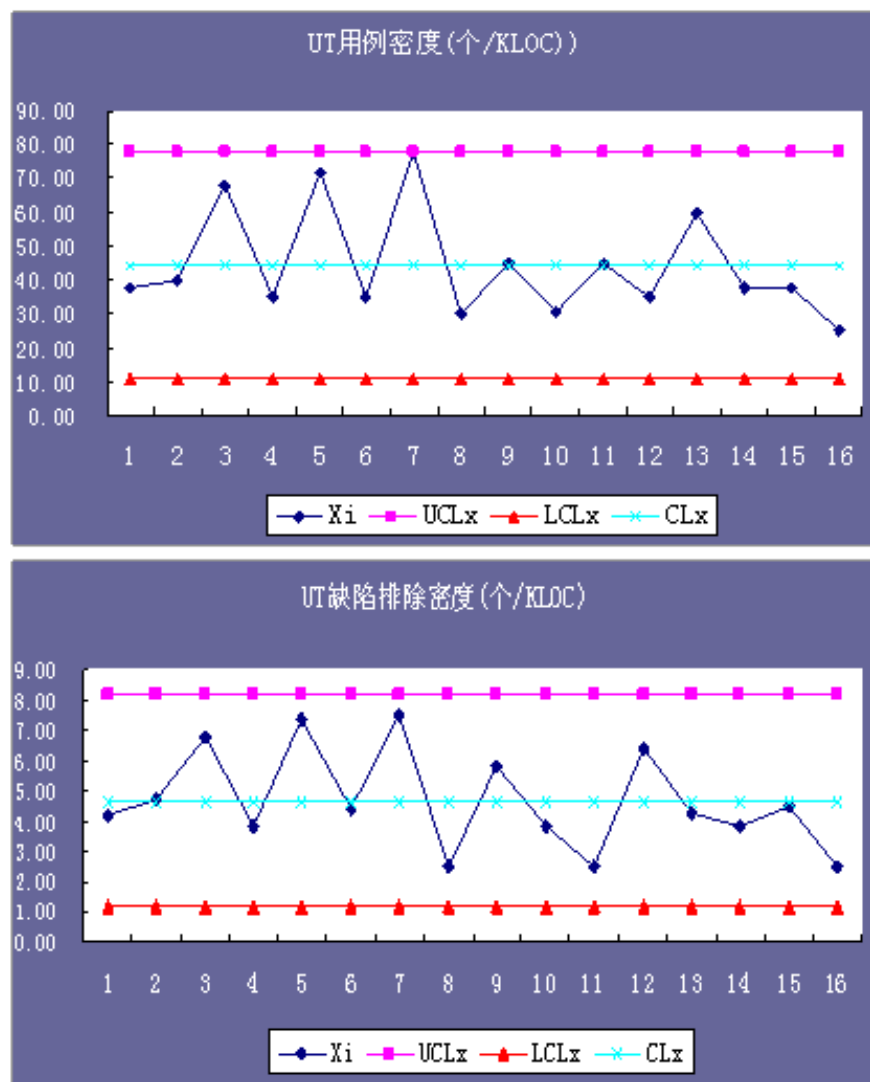
- 代码评审
 - 代码评审的速度/发现缺陷
 - 评审准备时间/发现缺陷
- 单元测试
 - **UT**分支覆盖度/发现缺陷
 - 测试用例密度/发现缺陷



建立过程性能基线-代码评审实例



建立过程性能基线-UT实例



回归分析模型举例

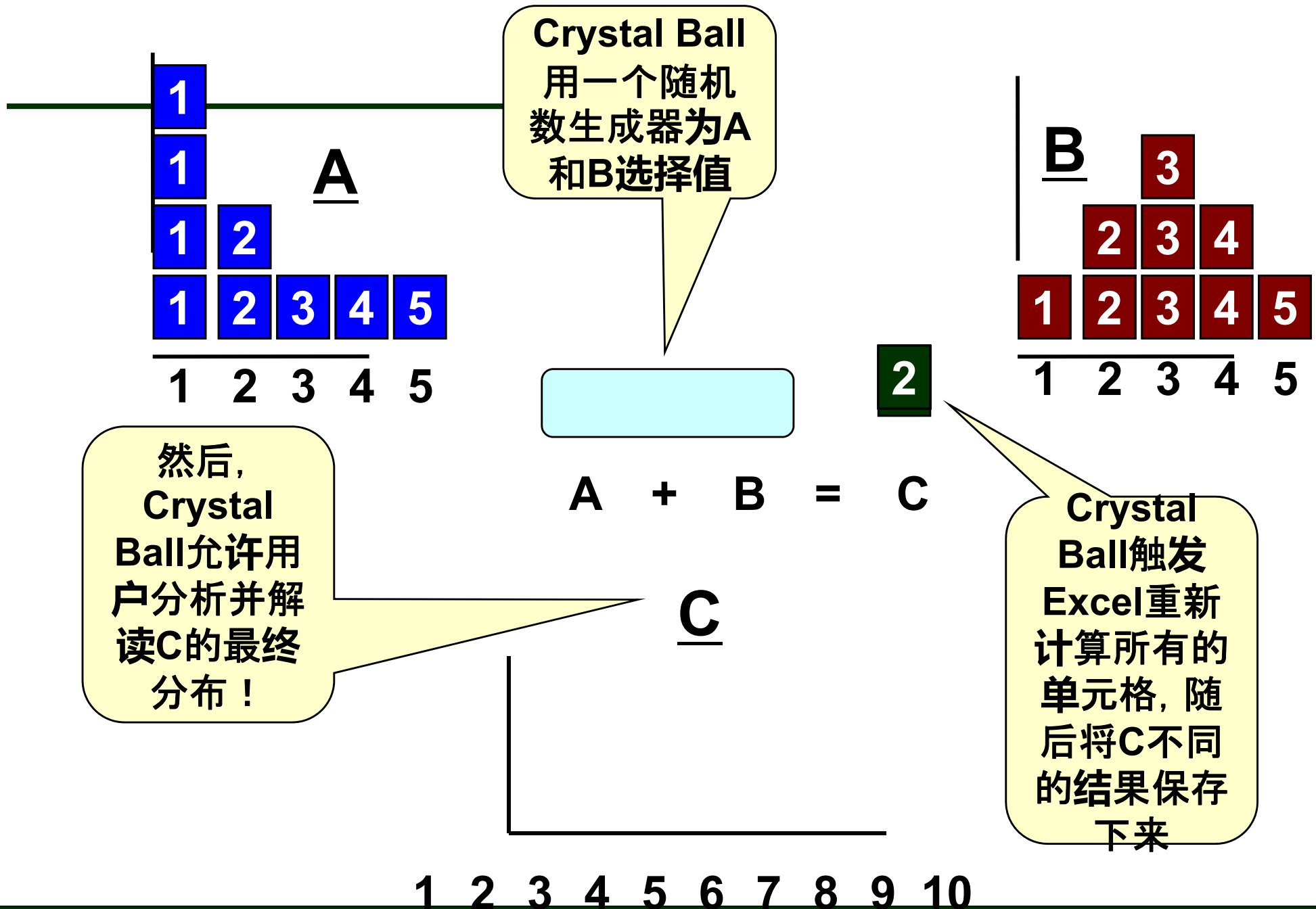
使用这些因素	预测结果
需求挥发性；设计和编码的复杂度；测试覆盖度；缺陷泄漏率	交付缺陷密度
人员更换率；应用领域经验的年数；任务转换率	生产率
测试设备可使用率；需求挥发性；复杂度；人员周更换率	成本和进度变化差异
任务周期；人员可用性；未定义的需求百分比；缺陷发现率	周期时间或面试时间



蒙特卡罗模拟

- 可以对不确定的变量建模（例如：输入大量的数值而不是单个值）
- 灵敏度分析更加准确
- 分析许多不同的不确定性变量的同步影响（例如：更加现实逼真）
- 使受众更加容易投入并接受建模，因为分析中包含了不确定变量的值
- 建立结果的置信度（例如：支持风险管理）





性能基线和模型的用途

- 在项目层面：
 - 在计划期间
 - 新的生命周期阶段开始
 - 重新计划时
 - 监督项目进展时
- 在组织层面
 - 在面对目标分析组织标准过程集中的过程的过程性能
 - 评估过程改进活动的投资回报，或可能的投资回报



过程改进成功案例

- 介绍



跟踪生命周期过程性能基线(示例)

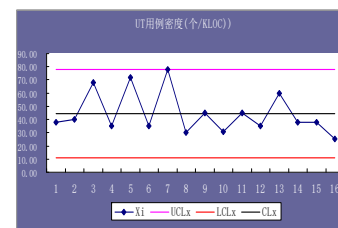
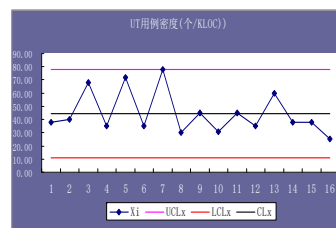
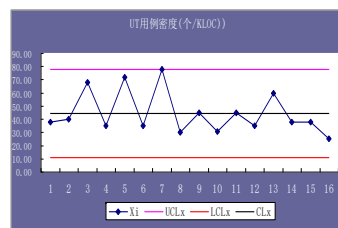
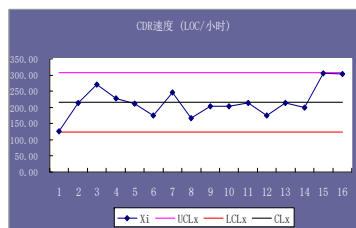
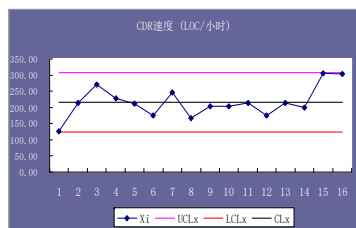
需求沟通时间

审查速度

单元测试覆盖度

测试覆盖度

客户使用时间



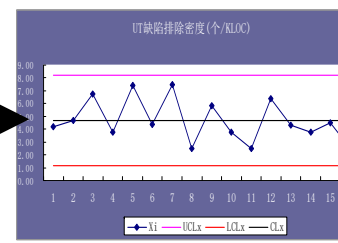
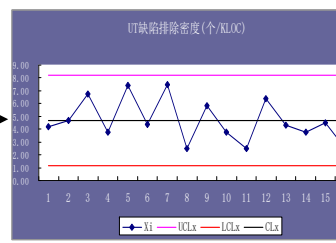
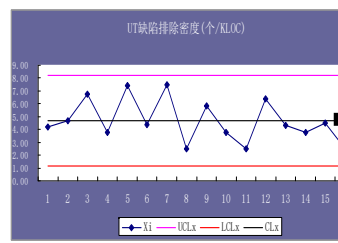
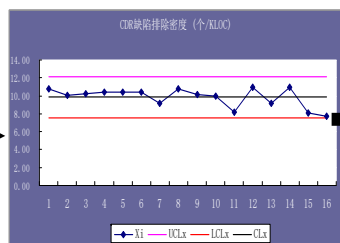
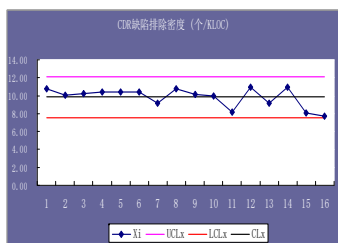
发现需求缺陷

发现设计缺陷

发现代码缺陷

发现缺陷

故障率



需求

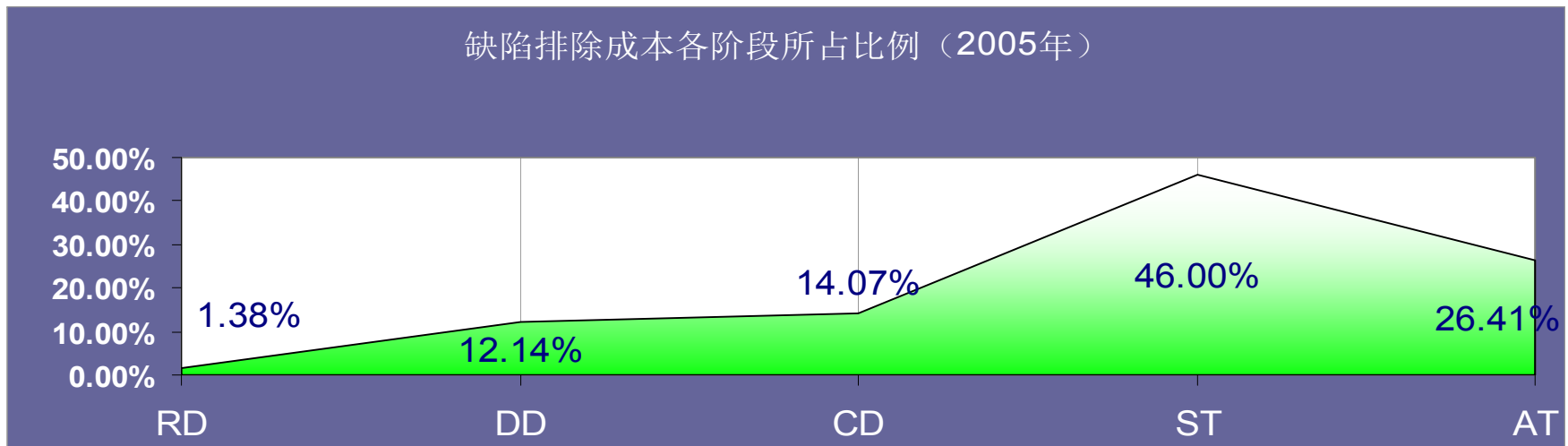
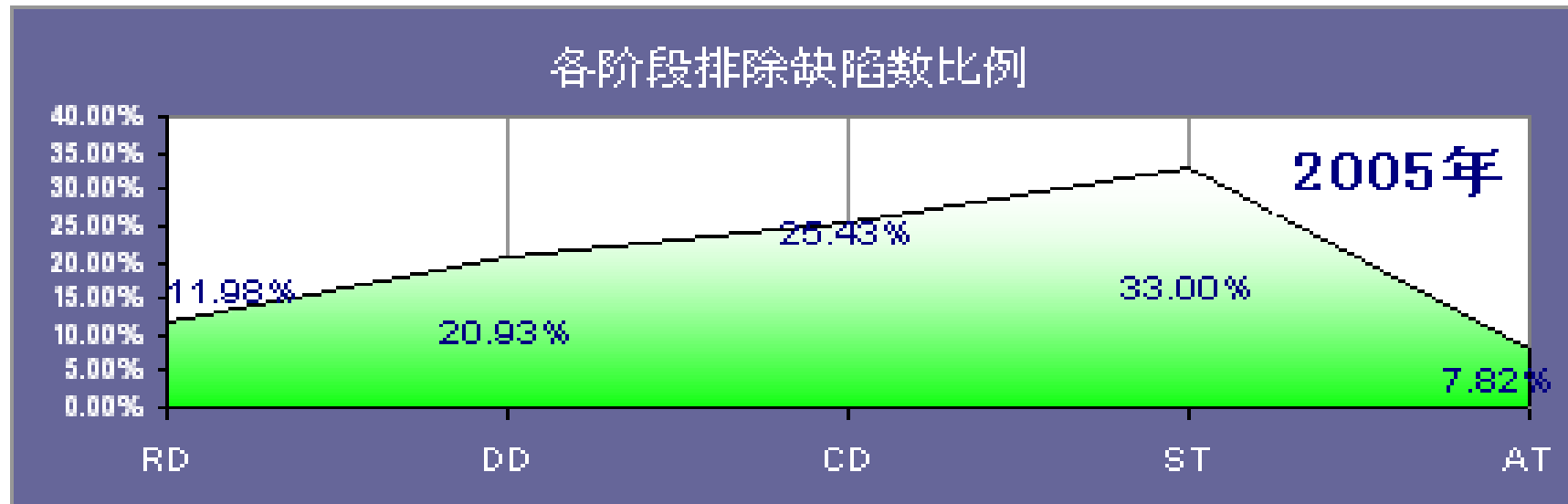
设计

编码

测试

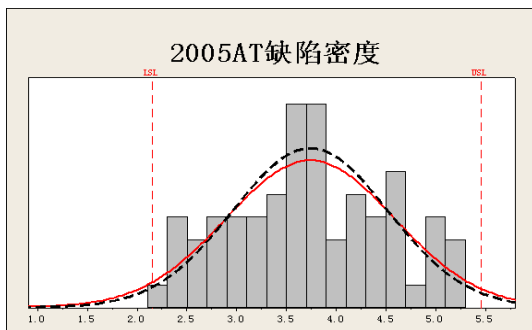
交付

某企业改进前

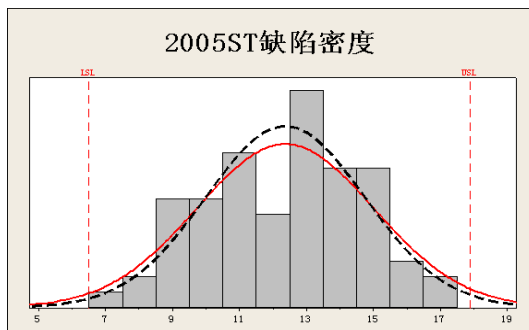


各阶段缺陷排除密度的变化（2005—2007）

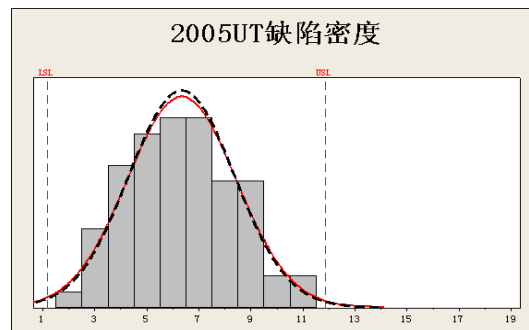
验收缺陷密度



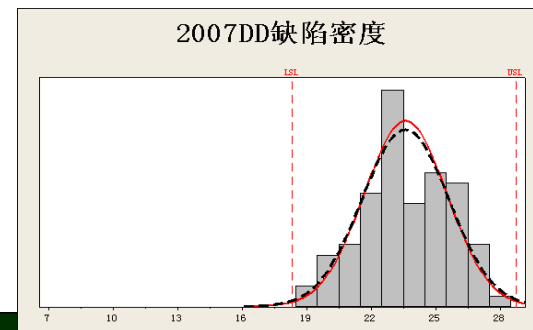
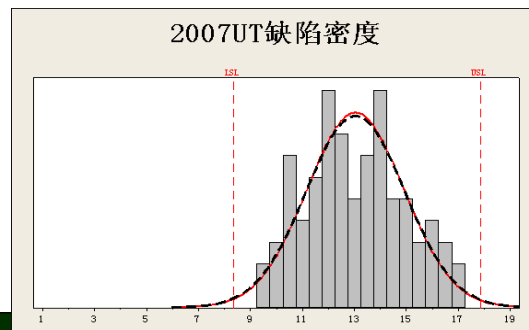
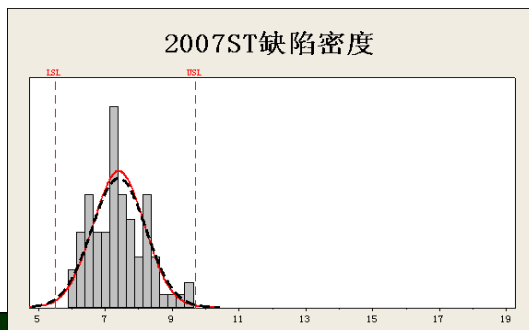
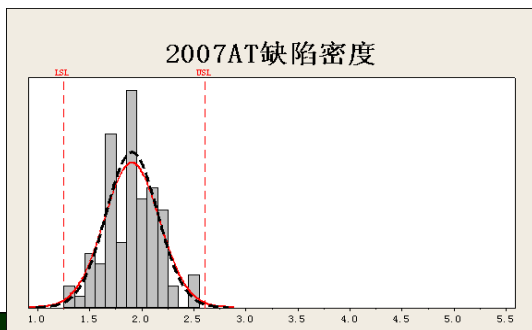
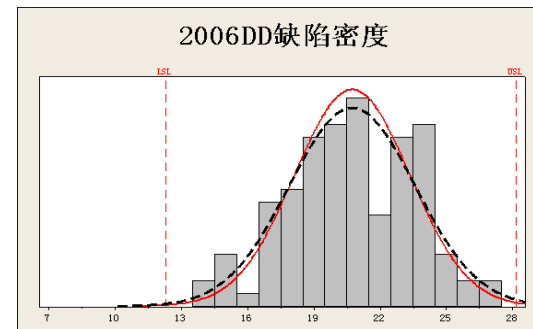
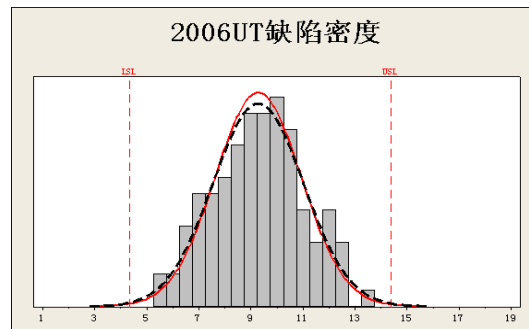
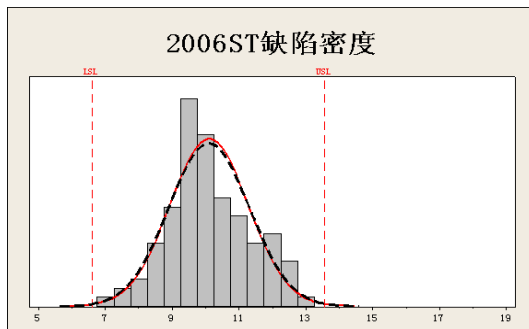
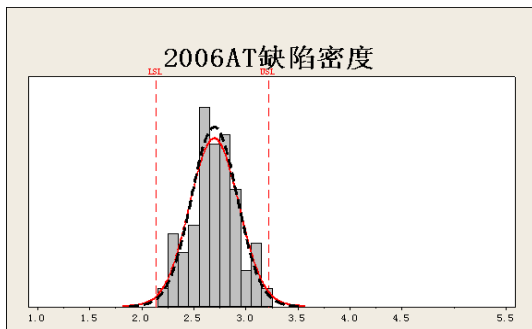
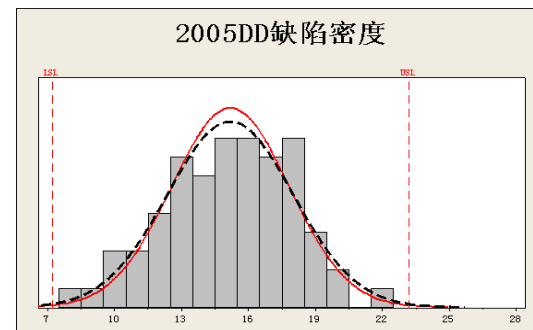
系统测试缺陷密度

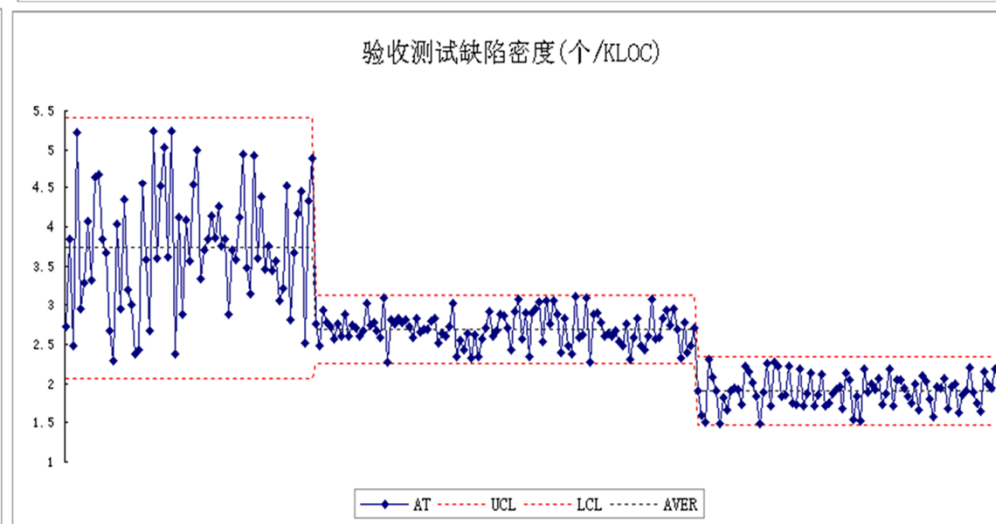
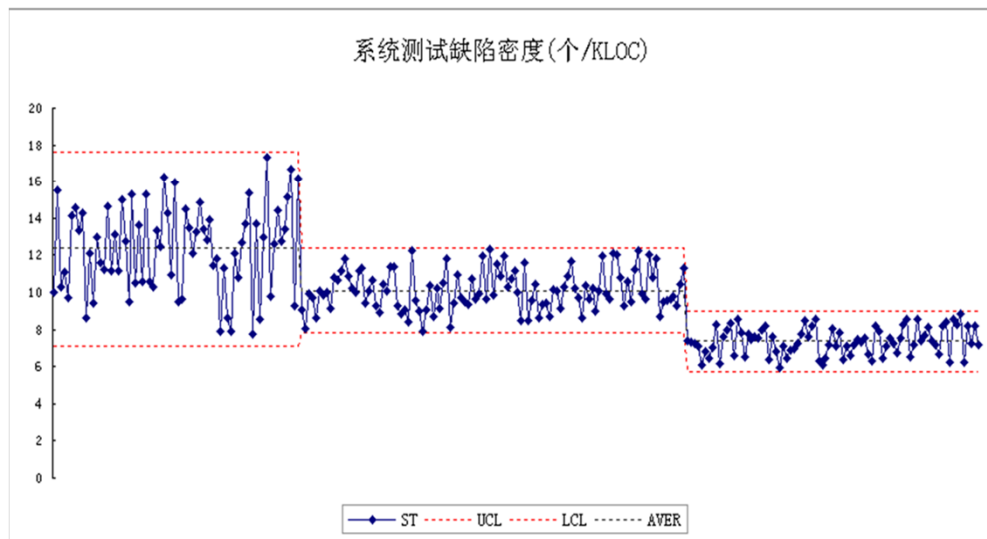
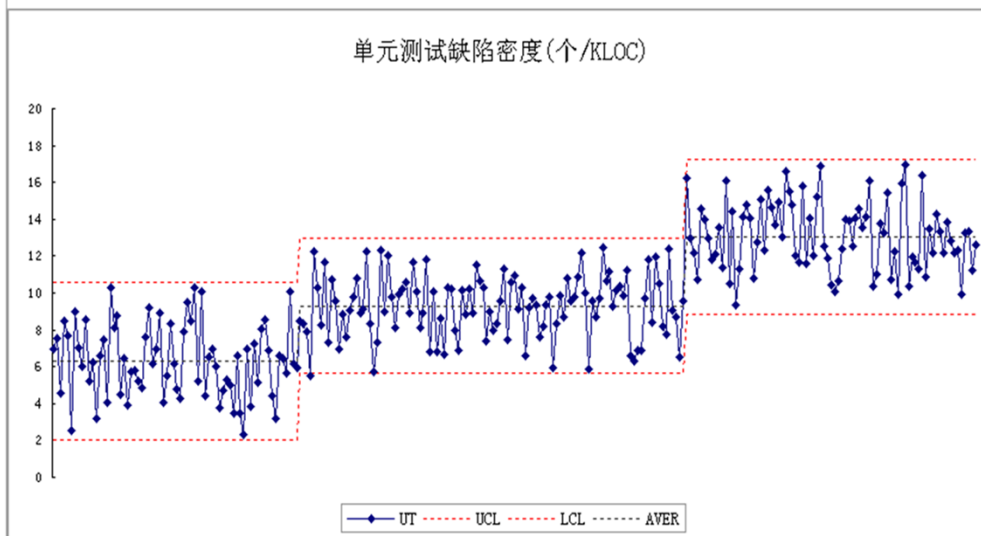
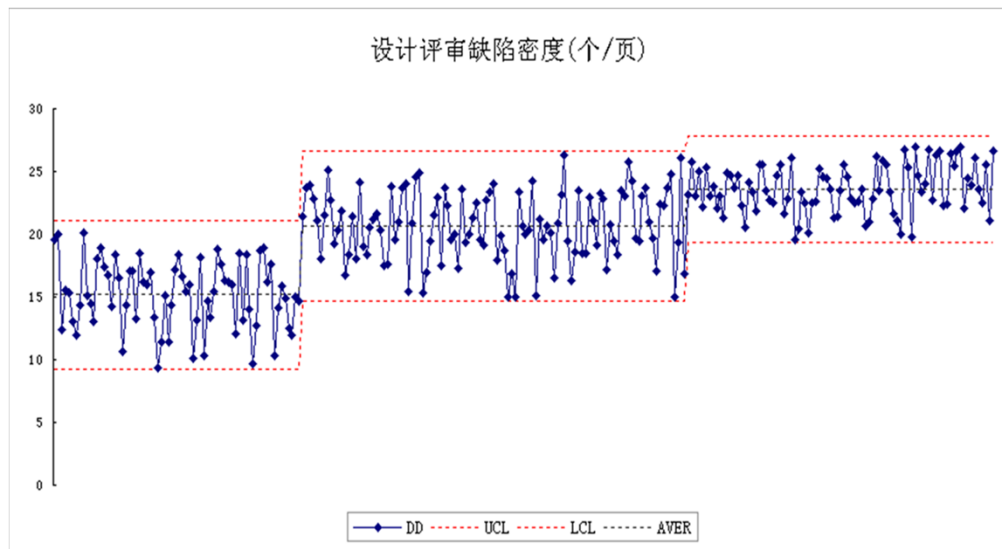


单元测试缺陷密度

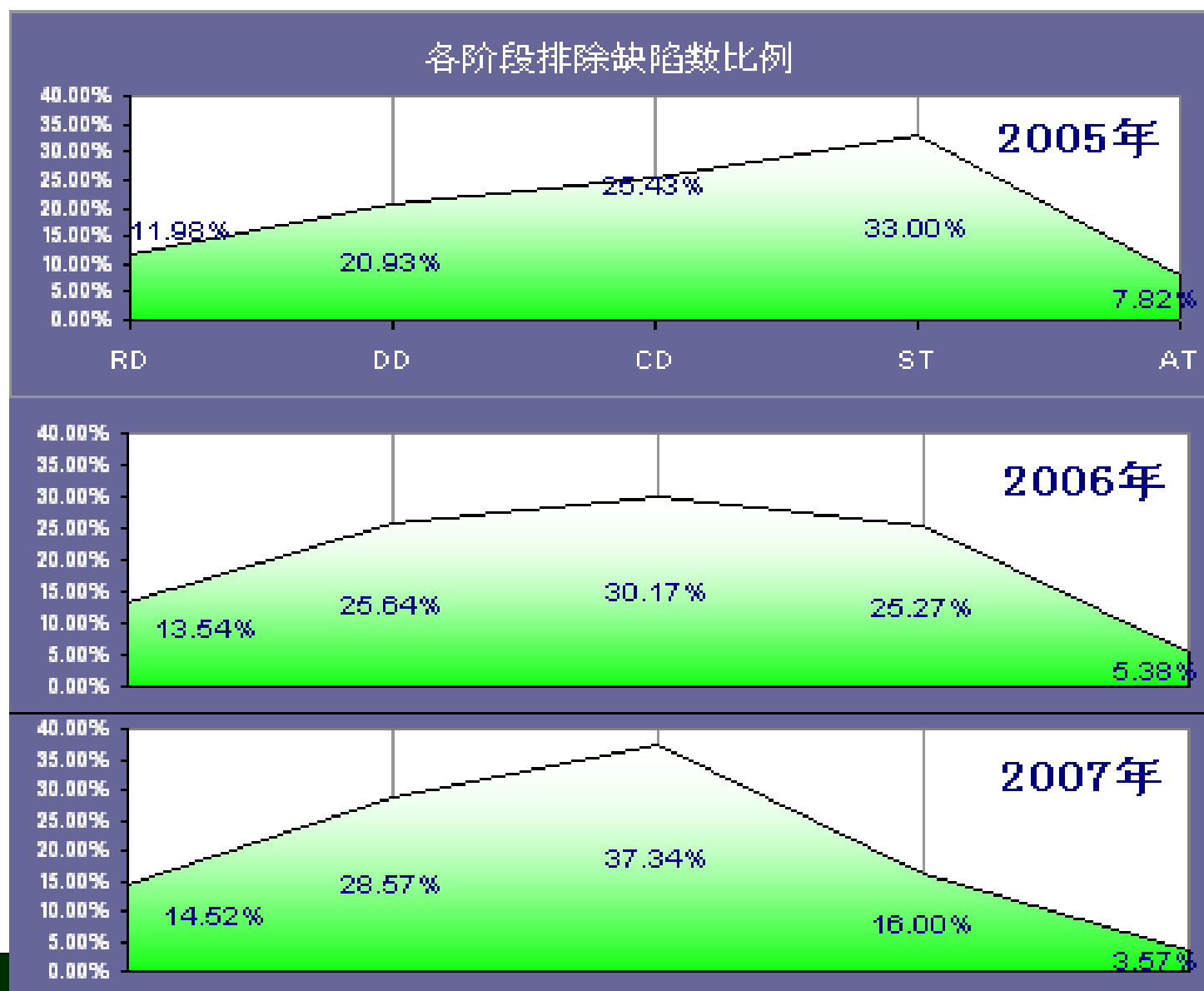


详细设计缺陷密度

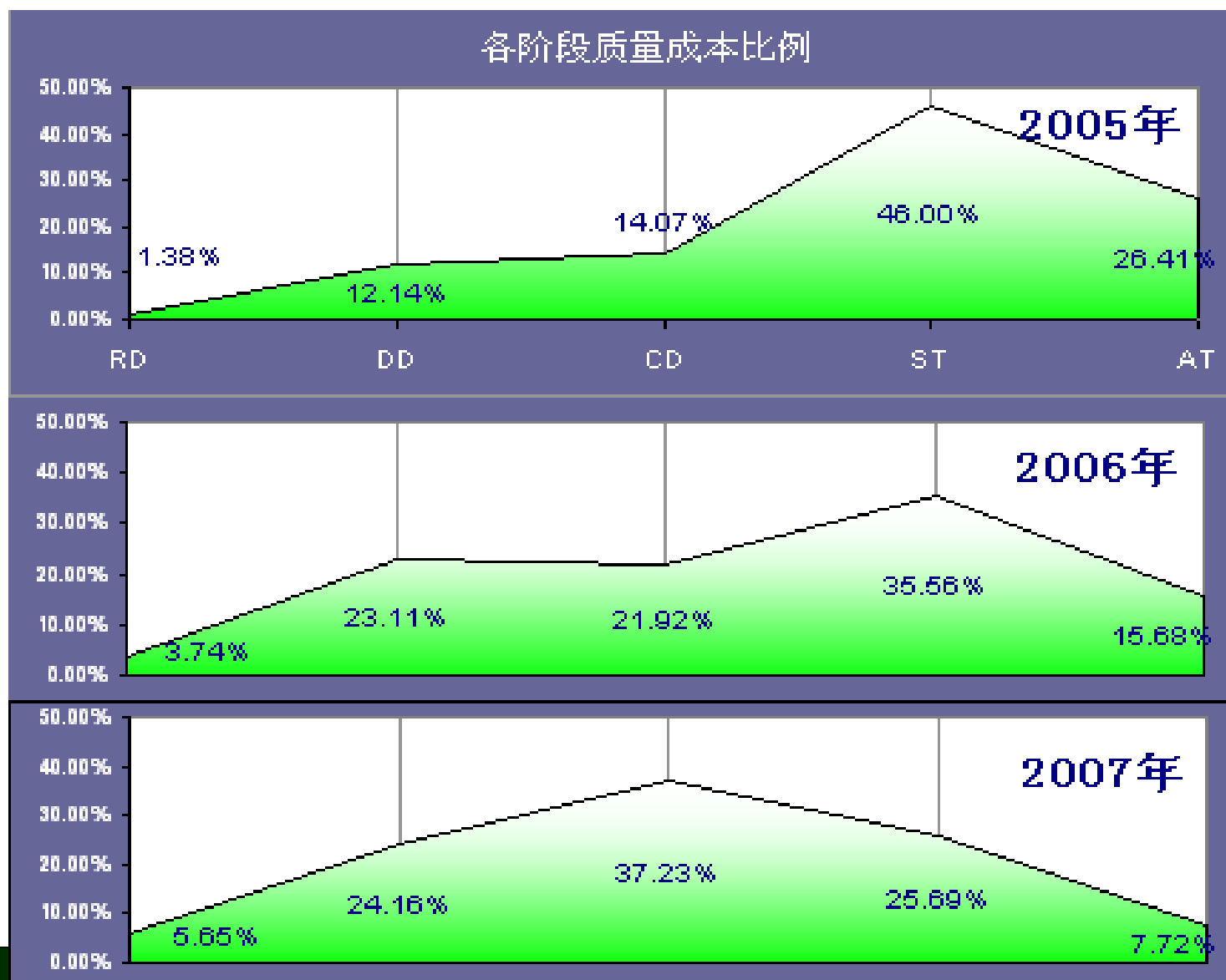




各阶段缺陷排除数占全生命周期比例的变化 (2005—2007)



各阶段质量成本的变化（2005-2007）



Questions?

