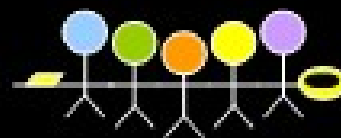


项目管理培训

软件开发中的风险管理



火龙果 • 整理
uml.org.cn

提纲

- 修订的流程和模板
- 实施中的两个问题
- 工具介绍

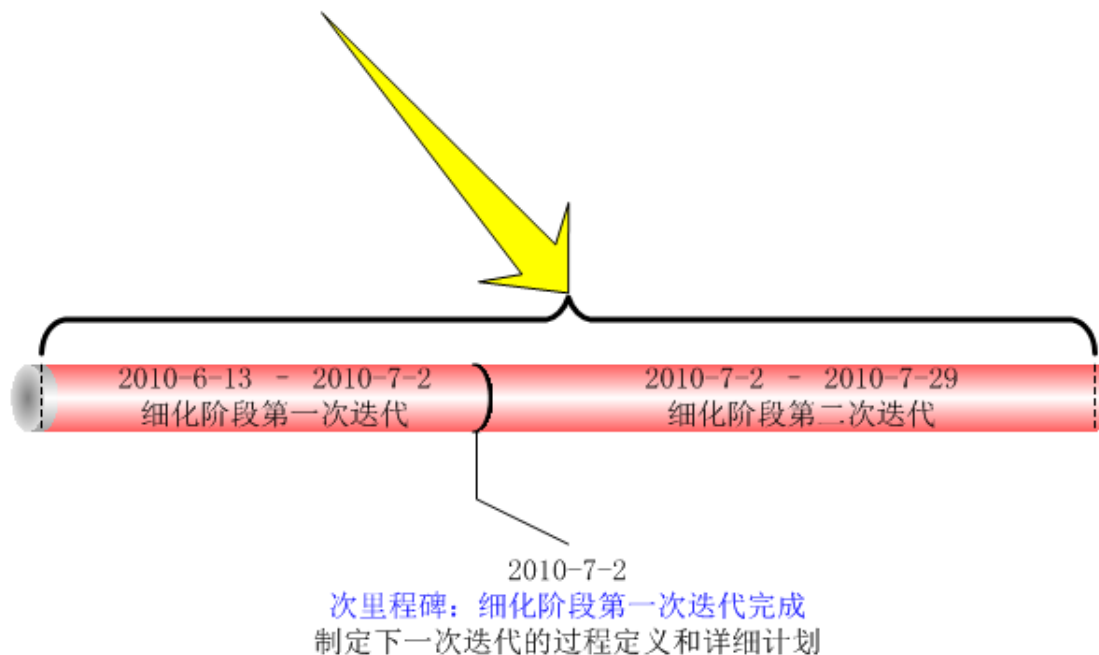
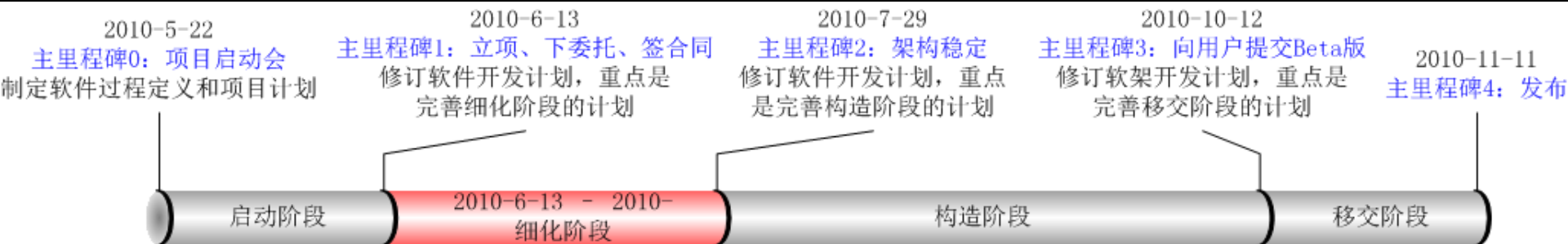
第一部分

- 修订的流程和模板
- 实施中的两个问题
- 工具介绍

什么是风险



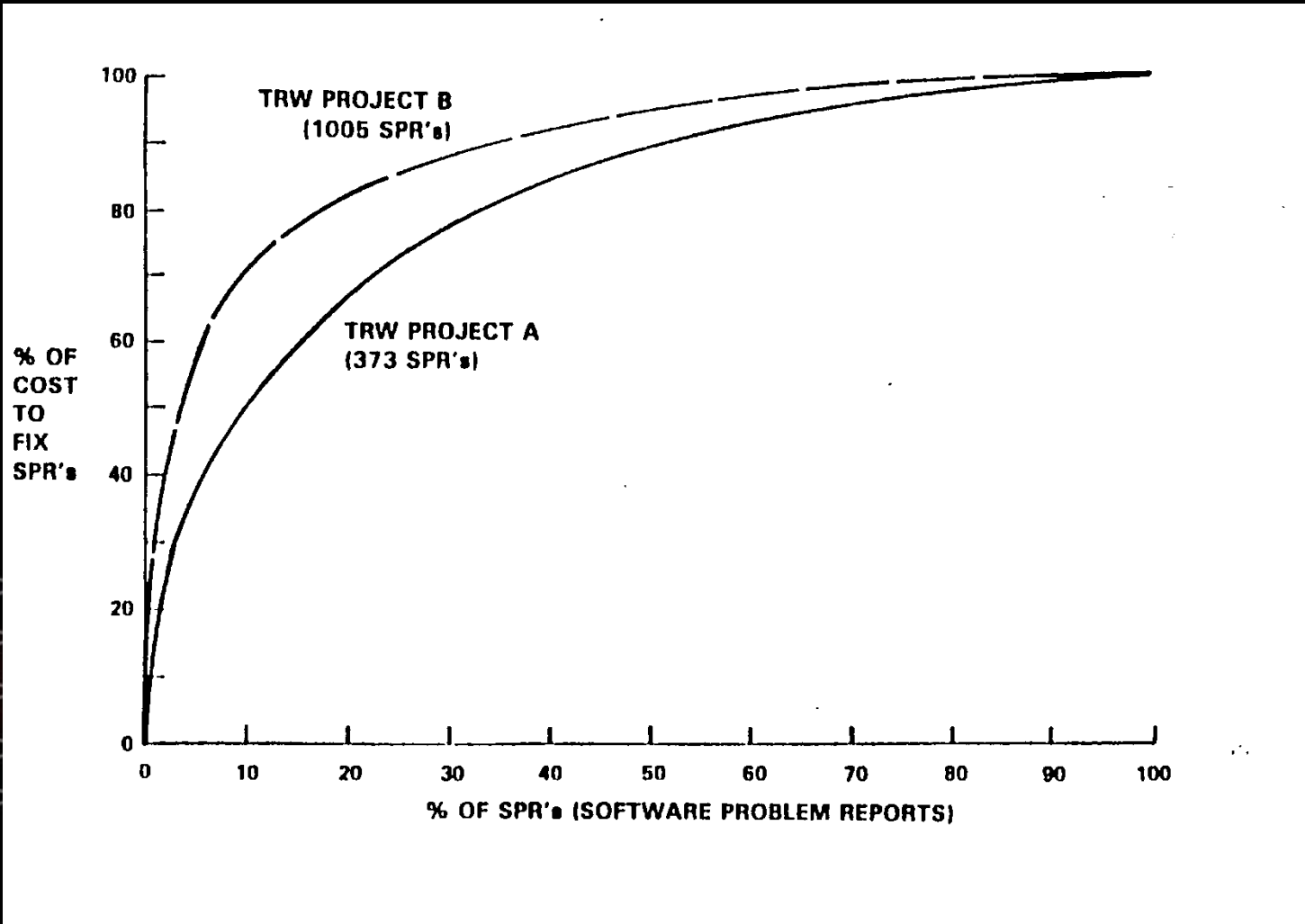
软件研发过程图示



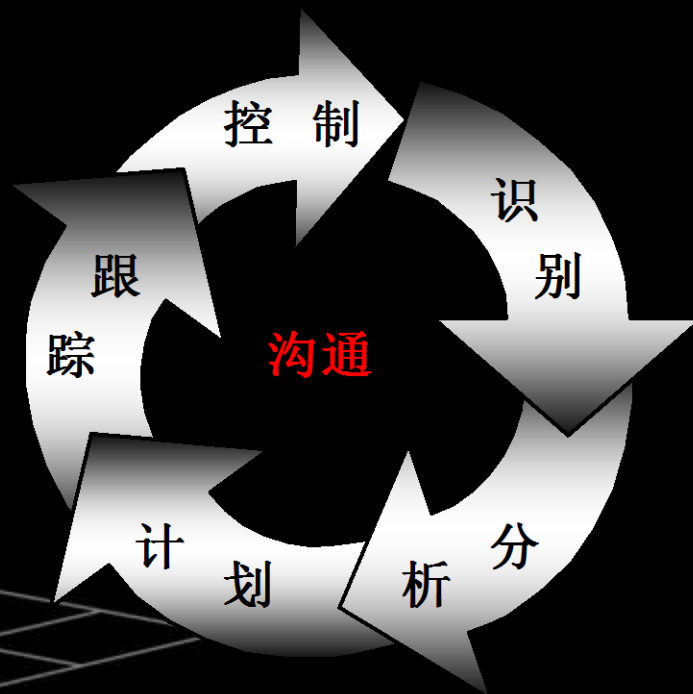
风险管理的重要性

- 明确软件系统的复杂度
- 保证项目的关键风险被关注
- 提供规避风险的技术
- 降低成本——减少返工
— 通常40-50%

返工成本分布



模型



Step1: 识别

- 识别技术
 - 检查表
 - 决策动机分析
 - 假设条件分析
 - 分解、细化

Top 10 Risk Items: 1989 and 1995

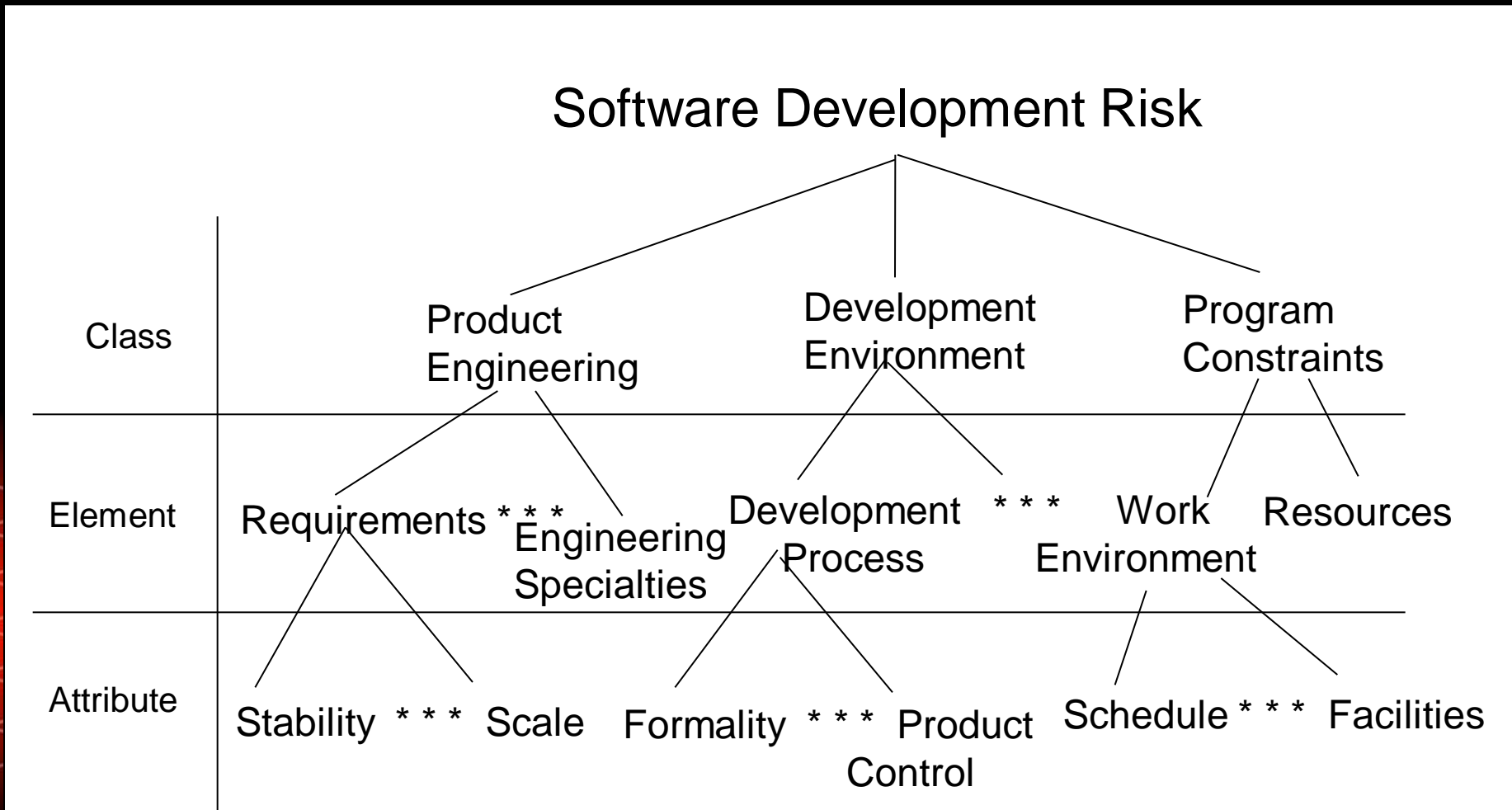
1989

1. Personnel shortfalls
2. Schedules and budgets
3. Wrong software functions
4. Wrong user interface
5. Gold plating
6. Requirements changes
7. Externally-furnished components
8. Externally-performed tasks
9. Real-time performance
10. Straining computer science

1995

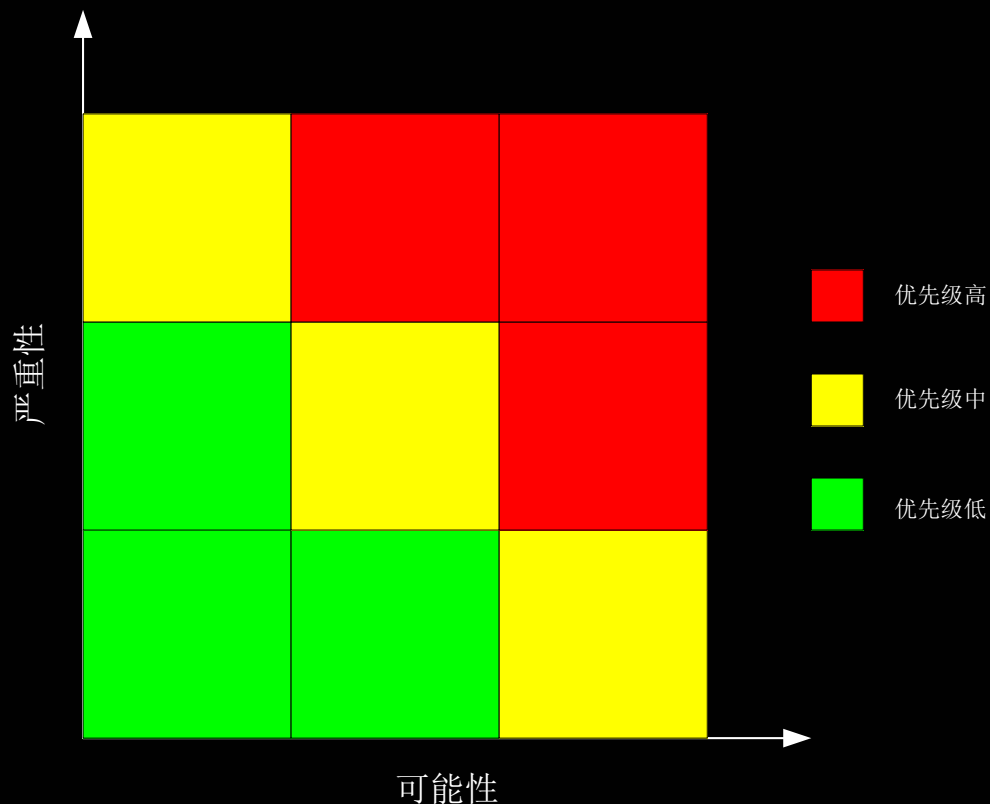
1. Personnel shortfalls
2. Schedules, budgets, process
3. COTS, external components
4. Requirements mismatch
5. User interface mismatch
6. Architecture, performance, quality
7. Requirements changes
8. Legacy software
9. Externally-performed tasks
10. Straining computer science

SEI 风险分类表



Step2: 分析

- 风险指数 = 严重性 × 可能性
- 严重性：
 - 高、中、低
- 可能性：
 - 高、中、低



Step3: 计划

- 策略决策
- 规避计划
 - 风险发生前的一些行动
- 应急计划
 - 风险发生后的一些补救措施

风险应对技术

- 风险规避/消除
- 风险转移
- 风险局部化
- 风险最小化
- 风险投保



选择正确的风险技术

- 高风险：风险转移、风险规避，同时准备应急计划
- 中等风险：规避计划或者应急计划
- 低风险：跟踪，可以不事先做出计划，风险如果发生才思考并采取措施

选择正确的风险技术

- 按类型
 - 两类风险
 - > 不确定性
 - > 能力的限制
 - 相应的风险应对措施

	不确定性	能力的限制
风险转移		X
风险规避/消除		X
风险最小化		X
风险投保	X	X
风险局部化	X	
风险监控	X	

风险管理计划

1. Why?

风险的重要性，与项目目标的关系

2. What, When?

消除风险的步骤、里程碑、工作产品

3. Who, Where?

责任人，所属的组织

4. How?

方法 (原型法、问卷调查, ...)

5. How Much?

资源 (预算、进度、关键人员)

风险管理计划：容错原形

1. 目标 (“Why”)

- 确定会引起不符合性能要求的容错特性，并改善

2. 工作产品和里程碑 (“What”， “When”)

- 第三周结束时

1. 候选容错特性评估报告
2. 可重用组件评价报告
3. 流量特征草稿
4. 原型评价计划
5. 原型描述

- 第七周结束时

6. 包括关键容错特性的可操作原型
7. 流量模拟
8. 性能报告
9. 容错特性描述和改善计划草稿

- 第十周结束时

10. 改进后的原型
11. 修改后的容错特性描述

风险管理计划：容错原形

- 职责 (“Who”, “Where”)
 - 系统工程事: A
 - 任务 1, 3, 4, 9, 11, 协助完成 5, 10
 - 开发负责人: B
 - 任务 5, 6, 7, 10, 协助完成 1, 3
 - 开发人员: C
 - 任务 2, 8, 协助完成 5, 6, 7, 10
- 方法 (“How”)
 - 用假设的容错性能要求来检验
 - 使用实时操作系统, 加入带有容错特性的原型
 - 用典型的流量评价性能
 - 根据观测的结果优化原型
- 资源 (“How Much”)
 - \$60K – 全职人员 (10周)*(3人)*(\$2K/人周)
 - \$0K – 3台工作站 (已有)
 - \$0K – 2个处理器 (已有)
 - \$0K – 1个测试协同处理器 (已有)
 - \$10K – 备用
 - \$70K – 总额

Step4: 跟踪

- 执行规避风险的计划
- 监控风险变化，及时启动应急计划
- 更新风险的状态，评估规避计划和应急计划的效果

Step5: 控制

- 根据规避和应急计划的效果调整风险应对的措施
- 根据项目的实际情况调整风险的优先级
- 增加识别的新风险和应对措施
- 在软件开发计划的任务安排中加入定期更新风险管理计划的任务

模板和案例

- 风险管理计划
 - 风险列表
 - 规避及应急计划
- 举例

第二部分

- 修订的流程和模板
- 实施中的两个问题
 - 风险识别
 - 策略决策
- 工具介绍

风险识别

- SEI风险分类
 - **Product Engineering.** The technical aspects of the work to be accomplished.
 - **Development Environment.** The methods, procedures, and tools used to produce the product.
 - **Program Constraints.** The contractual, organizational, and operational factors within which the software is developed but which are generally outside of the direct control of the local management.

SEI-TBQ

A. Product Engineering

2. Design

d. Performance

[Are there stringent response time or throughput requirements?]

[22] Are there any problems with performance?

- Throughput
- Scheduling asynchronous real-time events
- Real-time response
- Recovery timelines
- Response time
- Database response, contention, or access

[23] Has a performance analysis been done?

(Yes) (23.a) What is your level of confidence in the performance analysis?

(Yes) (23.b) Do you have a model to track performance through design and implementation?

我们整理的数据库

- 常见风险
- 各类型的项目风险管理举例

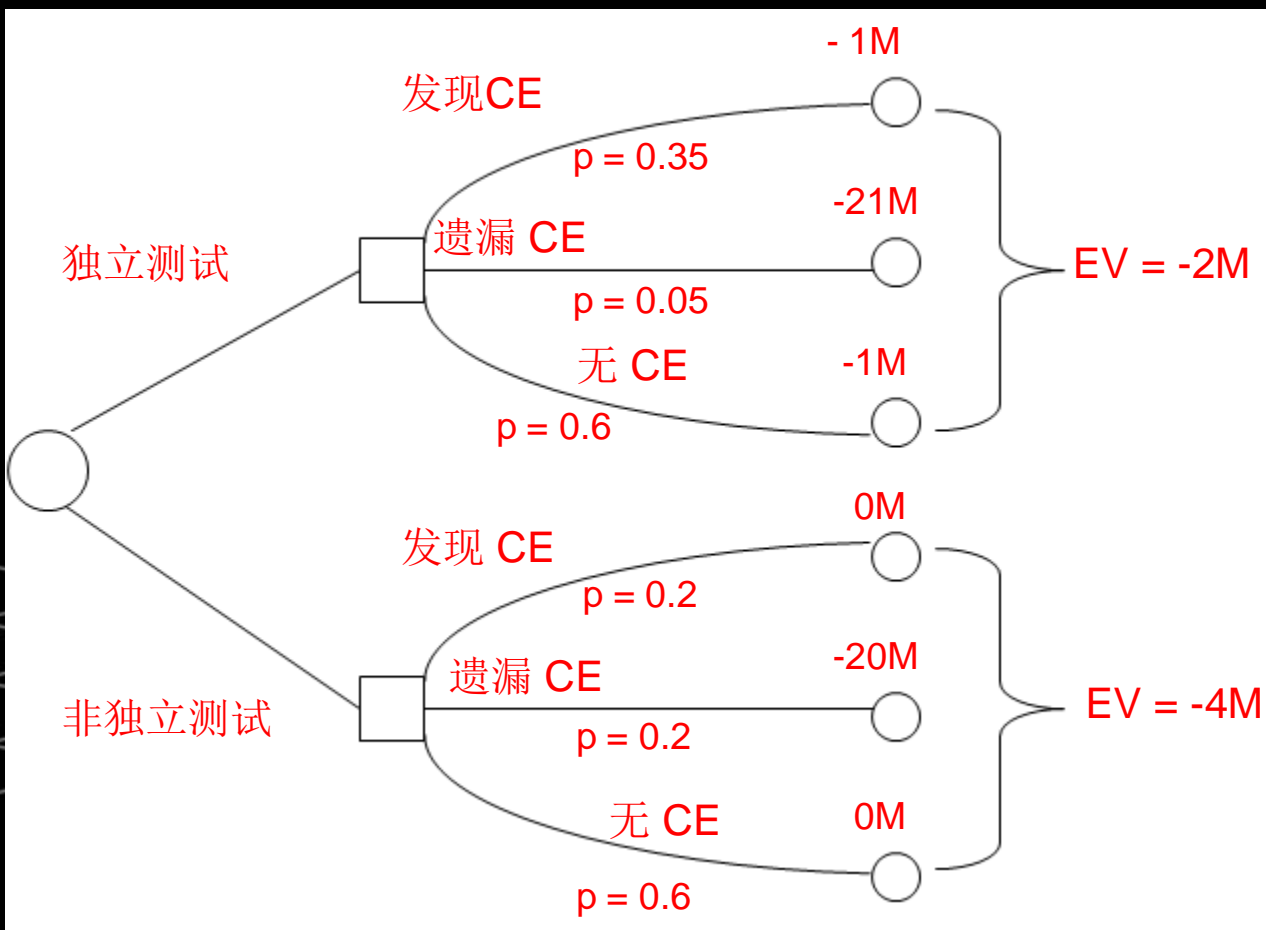
策略决策

- 风险管理是花费成本的
- 不同的策略成本收益也不同
- 决策数技术
 - 通过风险指数比较优劣
 - 比较全面的考虑

决策分析的例子

- 卫星控制软件
- 概率 (含有致命错误) = 0.4
- 发生致命错误的损失 = \$20M
- 如果做独立的测试:
 - 概率 (致命错误遗留) = 0.05
 - 增加的费用 = \$1M
- 不做独立的测试:
 - 概率 (致命错误遗留) = 0.2
 - 增加的成本 = \$0

决策分析图

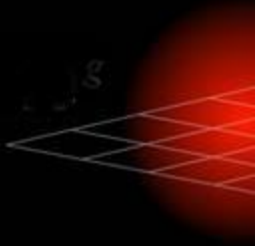


第三部分

- 修订的流程和模板
- 实施中的两个问题
- 工具介绍
 - 只是介绍别人的方法
 - 不是推广使用
 - 参考

工具背景

- Risk Radar
- Software Program Managers Network
- ~~Access2000~~
- Risk Tracking



THANK YOU

