基于风险的测试设计

郑文强

课程目录

为什么引入风险

穷尽测试不可能

测试时间有限

测试资源有限

课程目录

风险

风险定义

风险可以定义为事件、危险、威胁或情况等发生的可能性以及由此产生不可预料的后果，即一个潜在的问题！

风险级别由出现不确定事件的可能性和出现后所产生的影响（事件引发的不好的结果）两个方面来决定！

管理风险

Kill the managers!

管理风险 - 组织因素

- 技能、培训和人员的不足
- 个人问题
- 政策因素
  - 沟通需求和测试结果方面存在问题
  - 测试和评审中发现的问题未能用于改进
- 对测试的态度或者预期不合理

管理风险 - 技术因素

- 不能定义正确的需求
- 给定现有限制的情况下，没能满足需求的程度
- 测试环境没有及时准备好
- 数据转换、迁移计划，开发和测试数据转换/迁移工具造成的延迟
- 低质量设计、编码、配置数据、测试数据和测试

管理风险 - 供应商因素

- 第三方存在的问题
- 合同方面的问题

质量风险

Kill the customers!

质量风险案例

- 故障频发的软件交付使用
- 软件/硬件对个人或公司造成潜在损害的可能性
- 劣质的软件特性（如功能性、可靠性、易用性和性能）
- 低劣的数据完整性和质量（如数据迁移问题、数据转换问题、数据传输问题、违反数据标准问题）
- 软件没有实现既定的功能

课程目录

1. 为什么引入风险
2. 风险的概念
3. 风险管理过程
4. 基于风险的测试设计
5. 基于风险的测试管理
6. 优点与挑战

风险管理主要特点

- 系统的过程
- 主动的过程
- 持续的过程

风险管理主要活动

- Ø 风险识别
- Ø 风险分析
- Ø 风险应对
- Ø 风险控制

风险识别方法

- 专家咨询
- 独立评估
- 风险模板
- 经验教训（例如：项目评估会议）
- 风险研讨会（例如：失效模式和影响分析）
- 头脑风暴法
- 风险分类（或检查表）
- 过去的经验
- 问卷调查法

案例：风险问卷调查表

项目名称：IGMP 功能测试　　时间：

参与人员：XXX

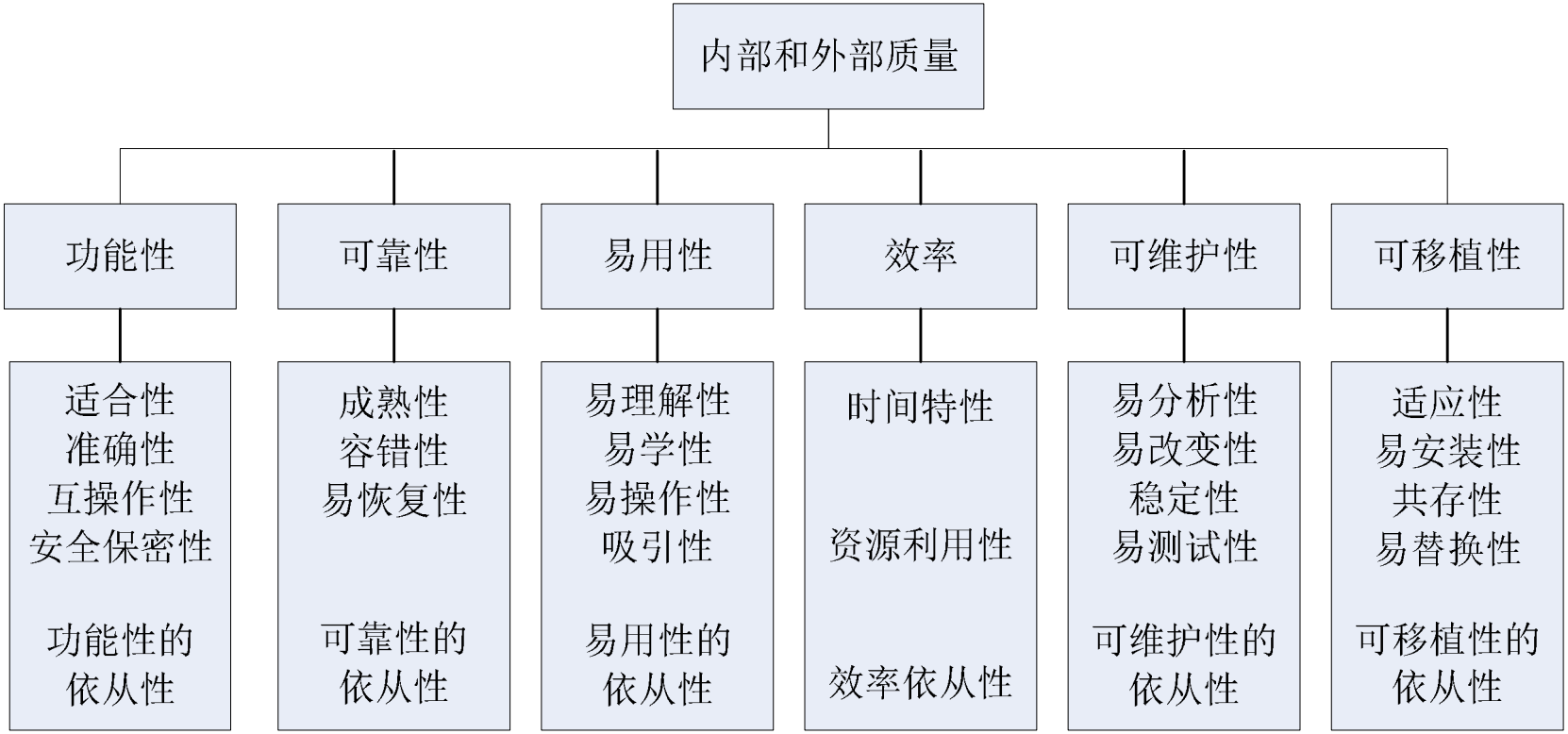
问卷调查表：

序号	问卷问题	是/否	备注
1	需求规格说明是否在开始设计之后还在修改？		
2	需求规格说明中是否存在“未确定”的内容？		
3	需求规格说明中是否包含你认为应该包含的内容？		
4	功能的复杂性是否超出组织目前的经验和技能？		
5	需求或者功能是否难以设计？		
6	需求的实现是否都有相应的解决方案？		
7	功能的内部接口是否清晰地定义？（例如：软件与硬件之间）		
8	是否存在性能瓶颈方面的问题？（例如：用户加入的时间）		
9	功能设计和实现是否做过性能分析？		
10	采用的硬件是否会限制需求的实现？（例如：内存容量）		
11	功能是否需要和商业软件 COTS 进行集成？		
12	功能是否会进行足够的组件测试？（例如：有足够的时间）		
13	假如没有足够时间进行组件测试，组织是否会采用折衷的方法？		
14	设计规格说明是否足够详细地进行代码和测试脚本的开发？		
15	是否有足够的硬件资源进行集成测试和系统测试？		
16	是否能够开发客户实际的场景和数据来验证需求？		
17	需求中定义的性能要求是否可测试？		
18	硬件系统是否能够及时到位？		
19	相关的测试仪表和设备是否能够及时到位？		
20	测试人员是否能够熟练使用测试设备？		
21	需求规格说明中是否有难于验证的需求？		
22	功能开发和测试是否在多个地方进行？		

案例：风险模板

序号	风险描述	注释
1	**资源**	
1.1	IGMP 测试仪表无法及时到位	
1.2	IGMP 测试硬件平台缺乏	
1.3	参与 IGMP 测试人员的人员目前还在其他项目中，无法及时介入 IGMP 测试工作	
1.4	IGMP 测试人员需要参与其他测试任务，例如：产品的入网测试	
1.5	缺乏 IGMP 测试服务器和 PC	
2	**技术**	
2.1	IGMP 测试范围的变更，由于需求文档中缺少非功能性需求的描述	
2.2	测试团队缺乏 IGMP 测试经验	
2.3	IGMP 测试工作量的偏低估算，有些测试活动没有估算在内	
2.4	用户的 IGMP 需求经常发生变更	
3	**质量**	
3.1	系统需求规格说明和设计规格说明评审过程中发现大量的缺陷	
3.2	测试计划没有经过详细的评审	
3.3	测试设计规格说明没有经过详细的评审	
4	**沟通**	
4.1	需求人员、开发人员和测试人员之间缺乏正规的沟通平台	
4.2	测试人员在测试过程中发现的缺陷，开发人员无法及时到现场进行确认	
5	**第三方外包**	
5.1	IGMP 相关的驱动部分通过外包方式进行开发，可能无法及时提交，或者提交的质量存在较大的不确定性	
6	**法律**	
6.1	某测试工具软件，测试团队只有两个授权（Licenses）。在测试过程中要保证只有两台机器安装该工具软件，避免不必要的版权纠纷。	

案例：风险分类 - 质量属性



案例：风险分类 - 常见风险列表

- 复杂性：规模庞大、复杂以及难于理解的对象。
- 新的：产品功能对项目成员都是全新的，没有任何经验可以借鉴。
- 变更的：变更或者进行了改进的功能对象。
- 关键的：软件系统或者产品的某个部分的失效会导致严重的后果。
- 精确的：软件系统或者产品的某个部分需要达到严格的精度。
- 常用的：客户经常操作或者使用的部分。
- 特殊的：对于组织特别重要的部分，例如：竞争对手不具备而运营商关注的功能。
- 第三方软件：软件系统或者产品中使用的不是该项目团队开发的部分。
- 分布式的：需要集成工作，但是由不同地点或者团队所开发的部分。
- 缺陷集群：已知的缺陷很多的地方。

案例：风险列表 - 领域内的风险列表

风险类别	详细的风险
安装了错误的文件	临时文件没有被清除
	不需要的文件被安装了
	需要的文件没有被安装
	正确的文件被安装在错误的目录
被破坏的安装文件	旧文件替代了新的文件
	在升级过程中用户数据库被破坏了
其他应用程序被破坏了	和其他产品共享的文件被修改了
	属于其他产品的文件被删除了
硬件没有被正确的配置	
安装操作人员替换或者修改了关键文件或者参数	
安装过程时间过长	
安装过程需要有固定的人员监控	
安装过程使人混淆	用户接口容易误用
	安装信息或者指导步骤让人摸不着头脑

风险分析

- 定性分析
- 定量分析

风险级别 = 风险可能性 x 风险严重程度

影响可能性的主要因素

- 技术和团队的复杂性。
- 业务分析人员、设计人员和程序员的技能。
- 团队内部矛盾。
- 与供应商的合同问题。
- 开发团队的地理分布。
- 老方法和新方法对立。
- 工具和技术。
- 不良的管理领导和技术领导。
- 时间、资源和管理压力。
- 缺乏初期的质量保证。
- 高变更率。
- 初期的高缺陷率。
- 接口和集成问题。

测试对象中存在潜在问题的可能性

影响严重程度的主要因素

严重程度指的是当风险发生后，对用户、客户或其他利益相关者影响的严重性

- 使用受影响的特性的频率。
- 对组织形象的损害。
- 商业损失。
- 潜在的金融、生态或社会方面的损失或法律责任。
- 民事或刑事法律制裁。
- 失去经营执照。
- 缺少合理的变通。
- 明显的失效导致负面宣传。

风险应对策略

- 风险减轻
- 风险避免
- 风险转移
- 风险接受
- 应急计划

风险优先级的策略

- 深度优先
- 广度优先

减轻风险的风险应对方法

- Ø 评审开发工作产品
- Ø 评审测试工作产品
- Ø 选择不同的测试级别
- Ø 选择测试设计技术
- Ø 选择经验丰富的测试人员
- Ø 执行再测试
- Ø 执行回归测试

案例：管理风险应对

序号	项目风险列表	产品风险应对和策略	责任人	完成时间
1	IGMP测试仪表没有到位	及时跟踪和催促采购团队		
2	IGMP测试硬件平台缺乏	和开发团队协调，借用部分硬件平台		
3	IGMP测试人员无法及时到位	替换测试人员，或者进行相应的备份		
4	IGMP测试PC无法及时到位	测试部门内协调借用		
5	IGMP估算的测试工作量偏低	加班、增加测试资源		
6	IGMP用户需求变更频繁	测试经理参与变更控制委员会		
7	项目团队缺乏正规的沟通平台	项目定时召开简短沟通会		

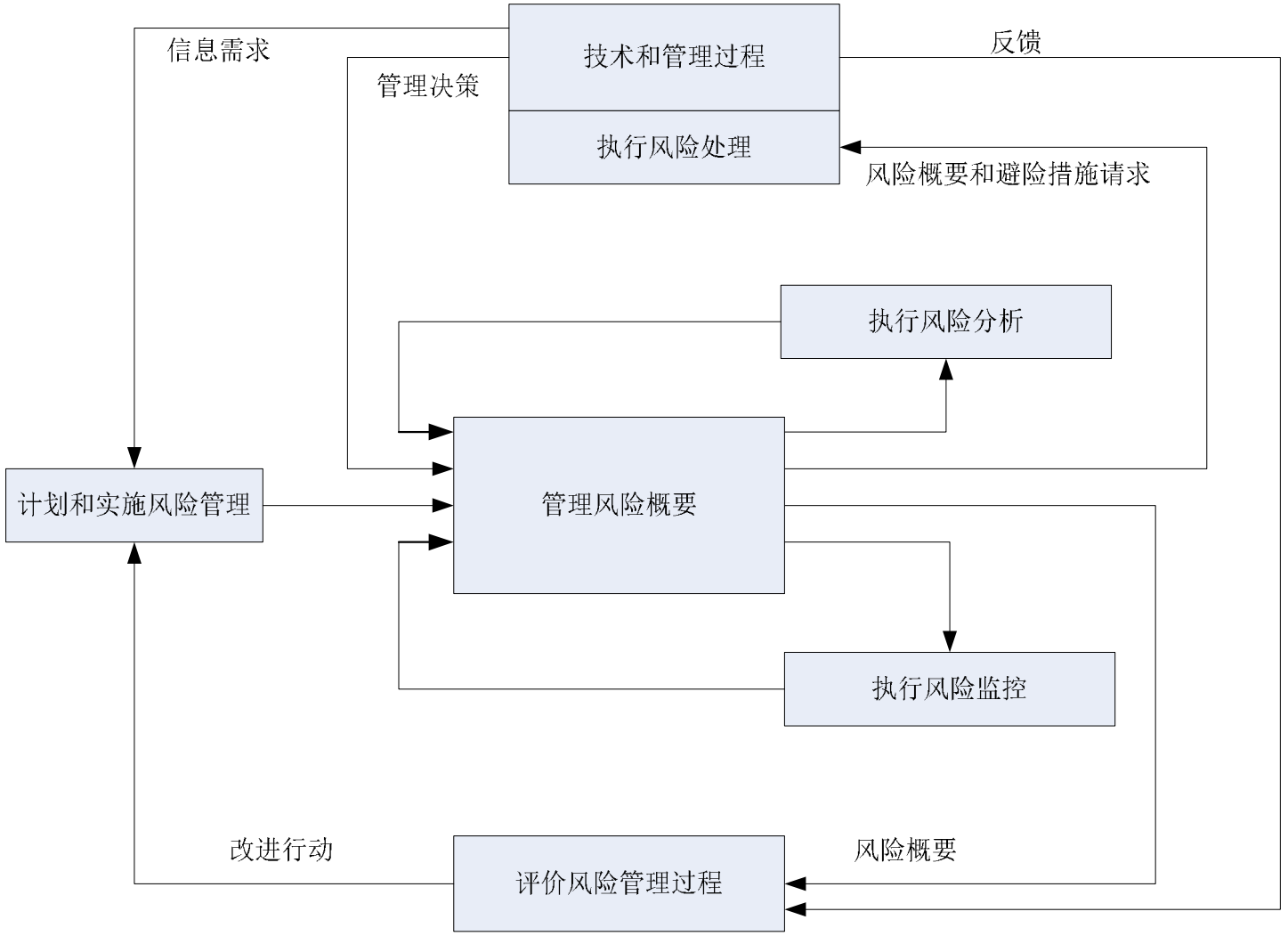
案例：质量风险应对

序号	产品风险	产品风险应对和策略	责任人	完成时间
1	IGMP需求规格说明质量低下	测试人员参与评审工作		
2	IGMP提交的版本质量低下	定义严格的测试入口准则		
3	IGMP的性能达不到用户要求	通过原型的方式对IGMP的架构进行验证		
4	IGMP协议可移植性风险	对移植的代码进行验收测试		

案例：基于风险状态的风险监控

- Ø 新的
- Ø 正在进行的
- Ø 关闭的
- Ø 接受的
- Ø 风险成真的
- Ø 陈旧的
- Ø 转移的

风险管理过程：ISO 16085



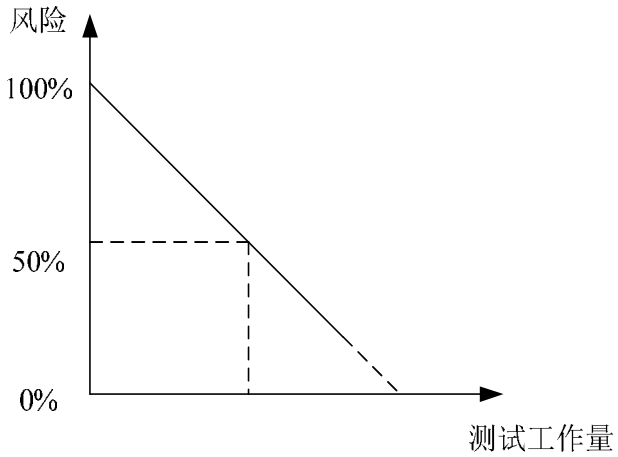
风险管理活动

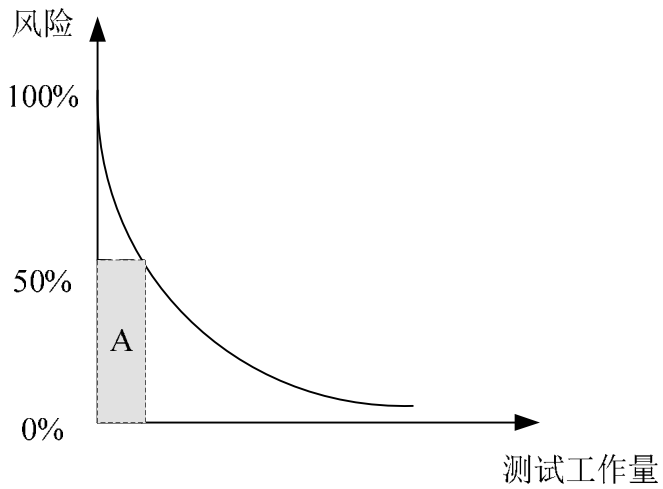
- Ø 计划和实施风险管理
- Ø 管理风险概要
- Ø 执行风险分析
- Ø 执行风险应对
- Ø 执行风险监控；
- Ø 评价风险管理过程

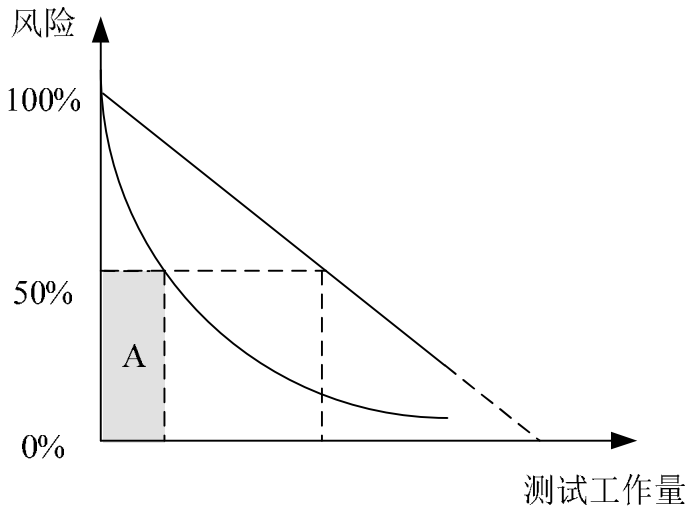
课程目录

1. 为什么引入风险
2. 风险的概念
3. 风险管理过程
4. 基于风险的测试设计
5. 基于风险的测试管理
6. 优点与挑战

风险与测试







测试设计与风险3要素

- 测试对象可能失效的方式
- 测试对象发生失效的可能性
- 测试对象发生失效导致的严重程度

基于风险的测试设计

启发式分析方法

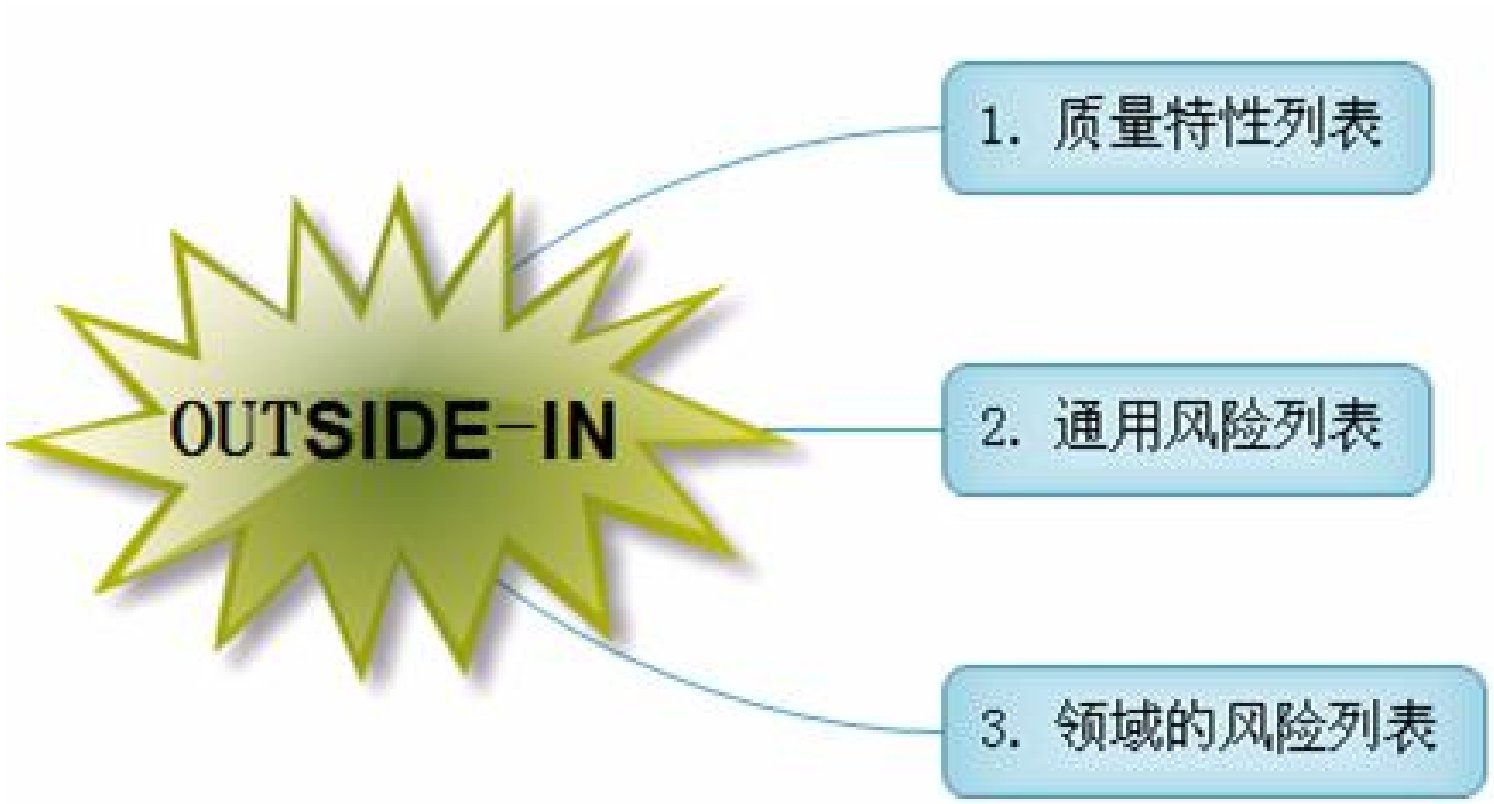
INSIDE-OUT问的3个问题

INSIDE-OUT工作过程

下面是开发人员和测试人员进行从内到外启发式分析方法的一个模拟场景，测试人员提出各种问题，开发人员针对每个问题进行解释或者思考：

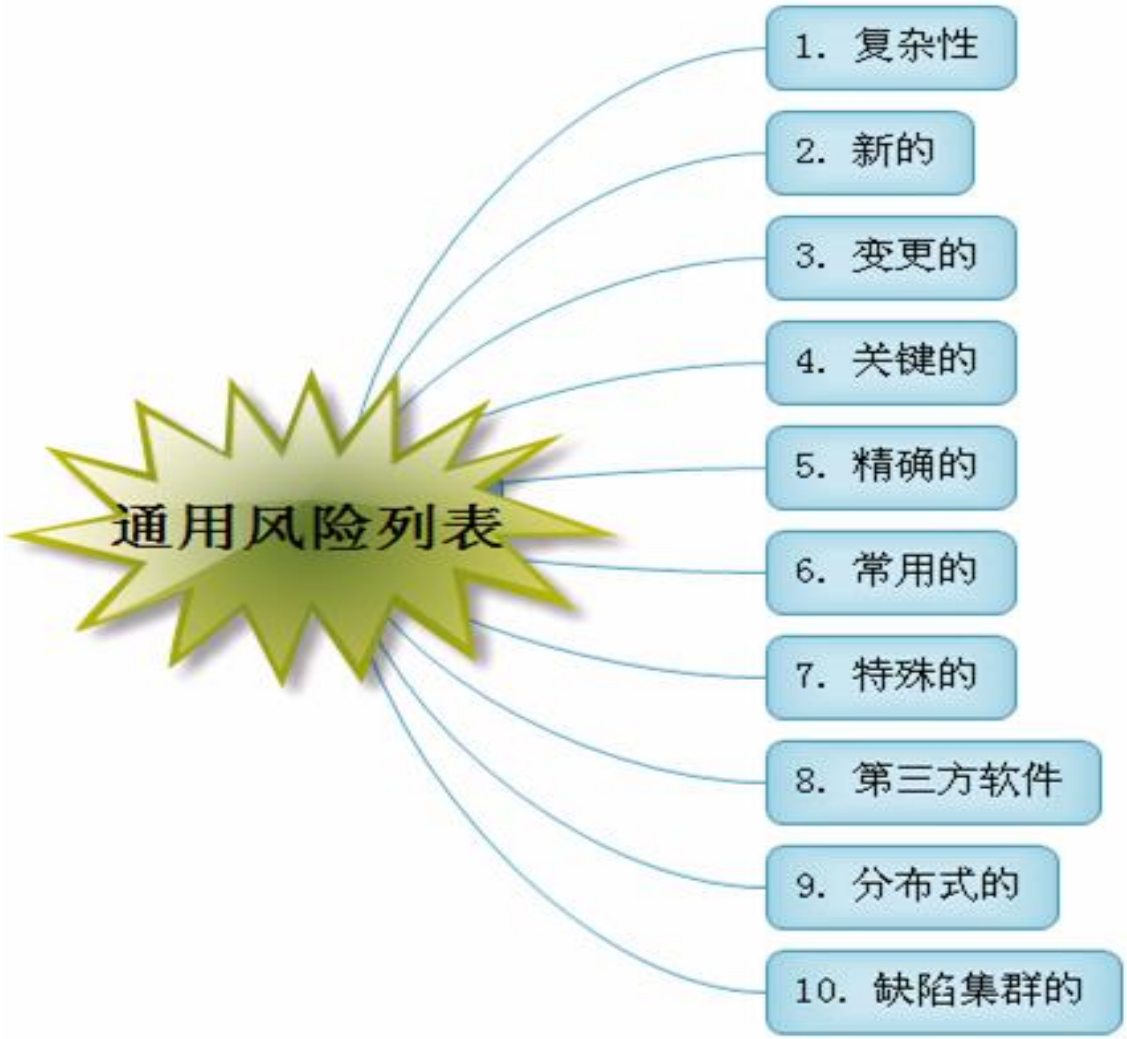
- [指着测试对象原理图中的一个模块] 假如这个功能失效，会发生什么样的现象？
- 这个功能模块会不会在不恰当的时候被调用？
- [指着原理图中的某个部分] 这里有没有相关的错误检查功能？
- [指着原理图中的某个箭头] 该箭头的具体含义是什么？假如这个箭头的通信不通，后果是什么？
- [指着原理图中的某个数据流] 假如这个数据流出现中断，你如何发现这个问题？假如没有发现这个问题，会出现什么后果？
- 这个功能能够处理的最大并发用户数是多少？具体的性能如何？
- 这个功能和其他功能之间是否有交互？
- 开发人员对这个功能最没有把握的部分是什么？从开发人员的角度，应该如何进行测试？

OUTSIDE-IN风险列表



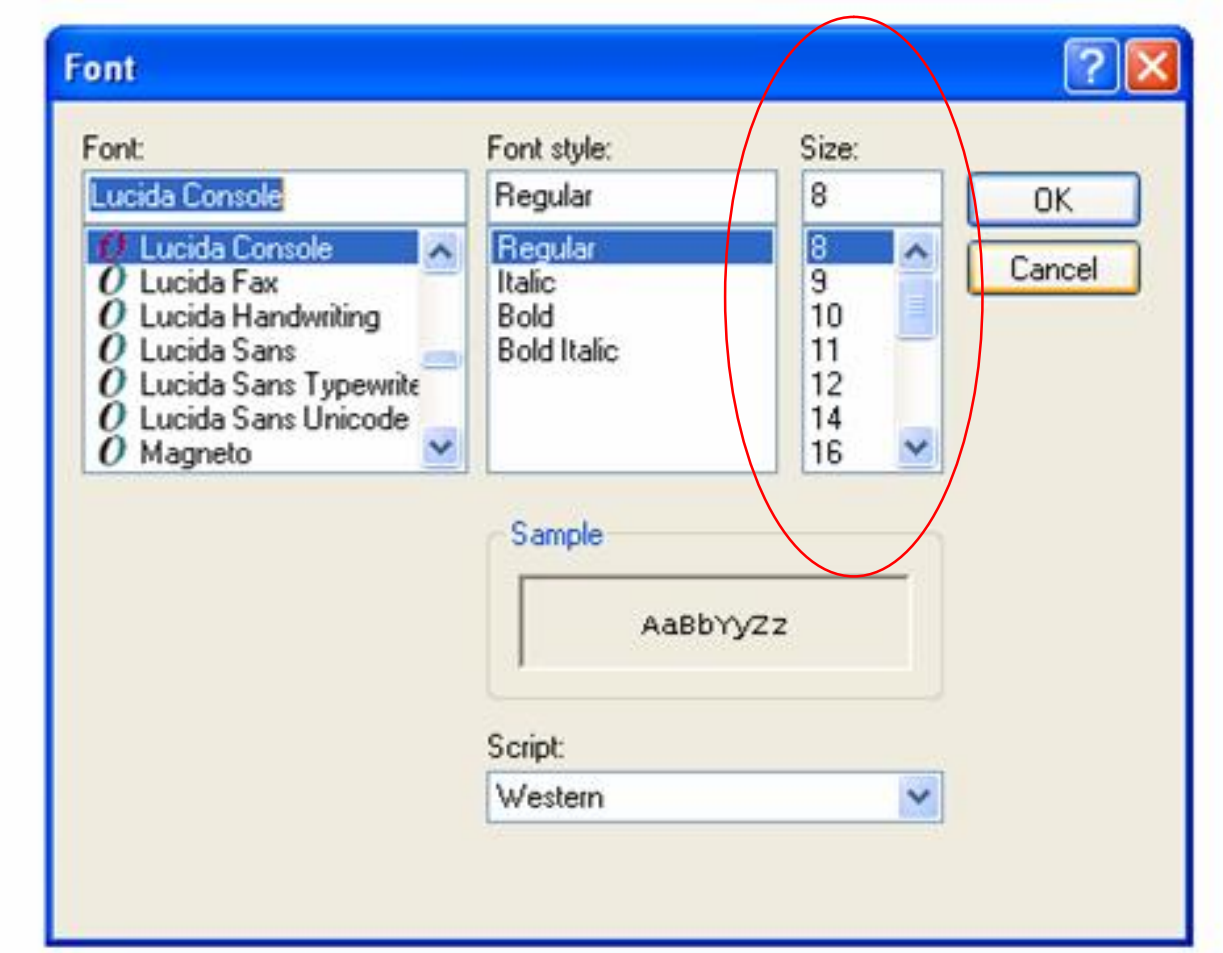
OUTSIDE-IN：质量特性

OUTSIDE-IN：通用风险列表



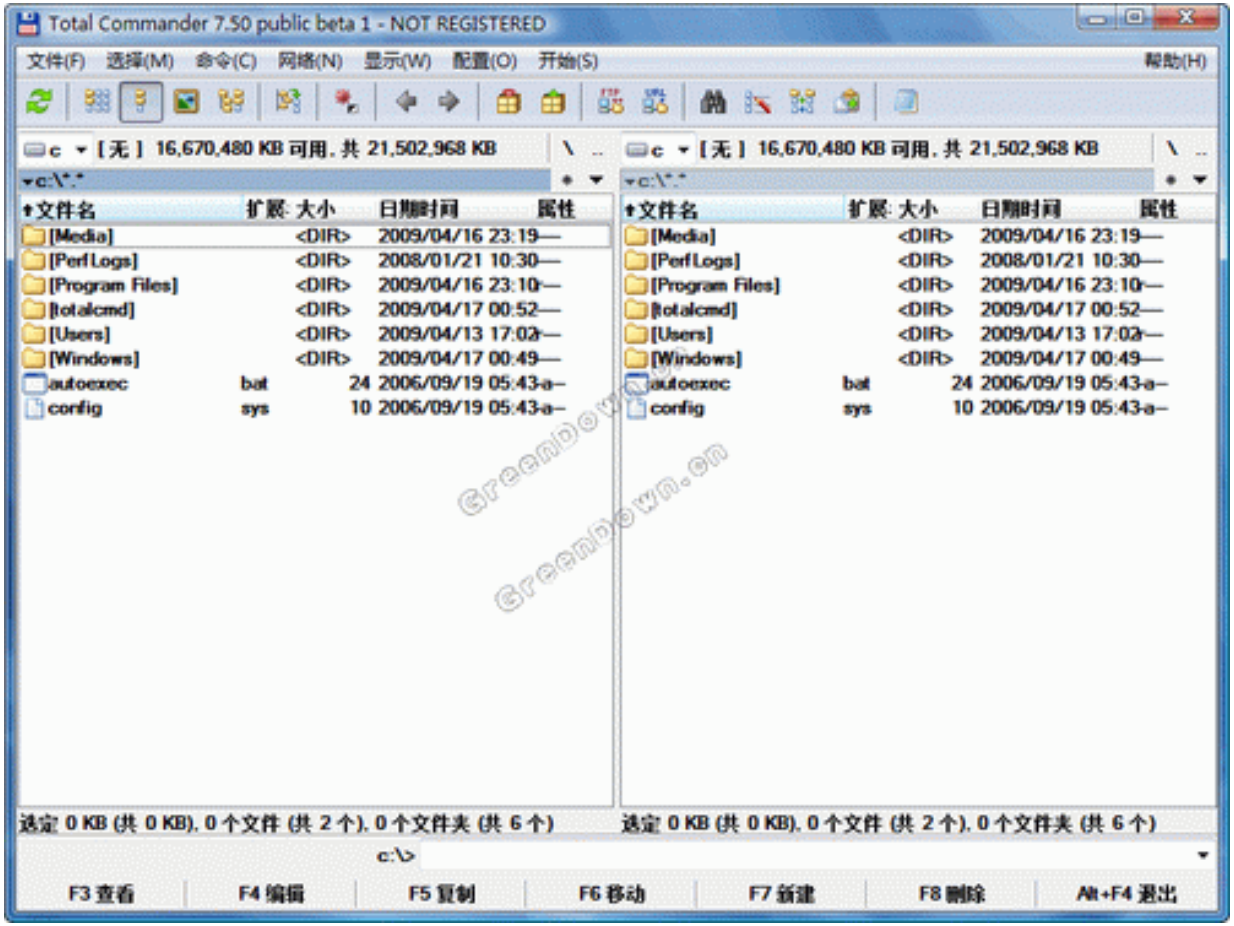
OUTSIDE-IN：领域风险列表

攻击：等价类与边界值



攻击：资源共享冲突

攻击：文件系统攻击



课程目录

1. 为什么引入风险
2. 风险的概念
3. 风险管理过程
4. 基于风险的测试设计
5. 基于风险的测试管理
6. 优点与挑战

基本假设

- 并不是所有的测试都是同等重要的
- 对测试对象的信心来自于执行了重要的功能或者模块的测试用例，发现并解决了其中发现的缺陷，而不是成功执行了成千上万个不是很有价值的测试用例
- 假如将主要的时间和资源等花费在高风险的功能和模块上，而不是那些风险严重程度低、发生可能性小的功能和模块上，那么测试对象可以以更低的成本，更快的速度交付给客户使用

风险的作用

- 帮助确定采用的测试技术和方法
- 帮助确定测试范围
- 帮助确定测试用例的优先级
- 决定是否可以通过一些非测试的活动减少风险，如测试人员培训

基于风险的测试策略

质量风险列表案例

序号	质量风险	质量风险应对和策略
1	IGMP需求规格说明质量低下	
2	IGMP提交的版本质量低下	
3	IGMP的性能达不到用户要求	
4	IGMP协议可移植性风险	

管理风险列表案例

序号	管理风险	管理风险应对和策略
1	IGMP测试仪表没有到位	
2	IGMP测试硬件平台缺乏	
3	IGMP测试人员无法及时到位	
4	IGMP测试PC无法及时到位	
5	IGMP估算的测试工作量偏低	
6	IGMP用户需求变更频繁	
7	项目团队缺乏正规的沟通平台	

测试结果和状态报告案例

状态	测试用例数目	百分比
通过	155	85.9%
失败	10	5.5%
被阻塞	14	7.7%
未执行	2	1.2%
总计	181	100%

风险级别	测试用例数目	通过	失败	被阻塞	未执行
1	7	6	1	0	0
2	31	27	2	2	0
3	37	30	2	5	0
4	57	50	3	4	0
5	49	42	2	3	2
总计	**181**	**155**	**10**	**14**	**2**

基于风险选择测试重点

	严重程度的影响因素		发生可能性的影响因素			总的风险级别
功能模块	使用频率	失效的可视性	复杂性	时间压力	地理分散程度	
权重	*3*	*10*	*3*	*10*	*1*	
功能模块A	5	3	2	4	5	
功能模块B	5	3	5	4	5	
功能模块C	2	1	2	2	5	
功能模块D	4	4	3	2	1	
功能模块E	5	1	4	2	1	

尽早开展风险活动

- 可以根据已知的产品风险的级别，分配测试工作量、选择测试技术、安排测试活动的顺序以及修正缺陷
- 可以根据已知的项目风险计划和管理测试工作，从而实现风险的减轻和缓解
- 可以根据剩余的风险报告测试结果和项目状态，帮助项目管理人员作出合适的决定。如果风险在可接受范围之内，可尽快发布软件版本；如果风险过高，则需要继续进行测试和相关的风险对应活动。

风险活动的迭代

- 存在新的或者有较大变化的产品风险
- 测试过程中发现不稳定的功能或者易产生缺陷的区域
- 修复缺陷带来的风险
- 潜在的未测试的区域

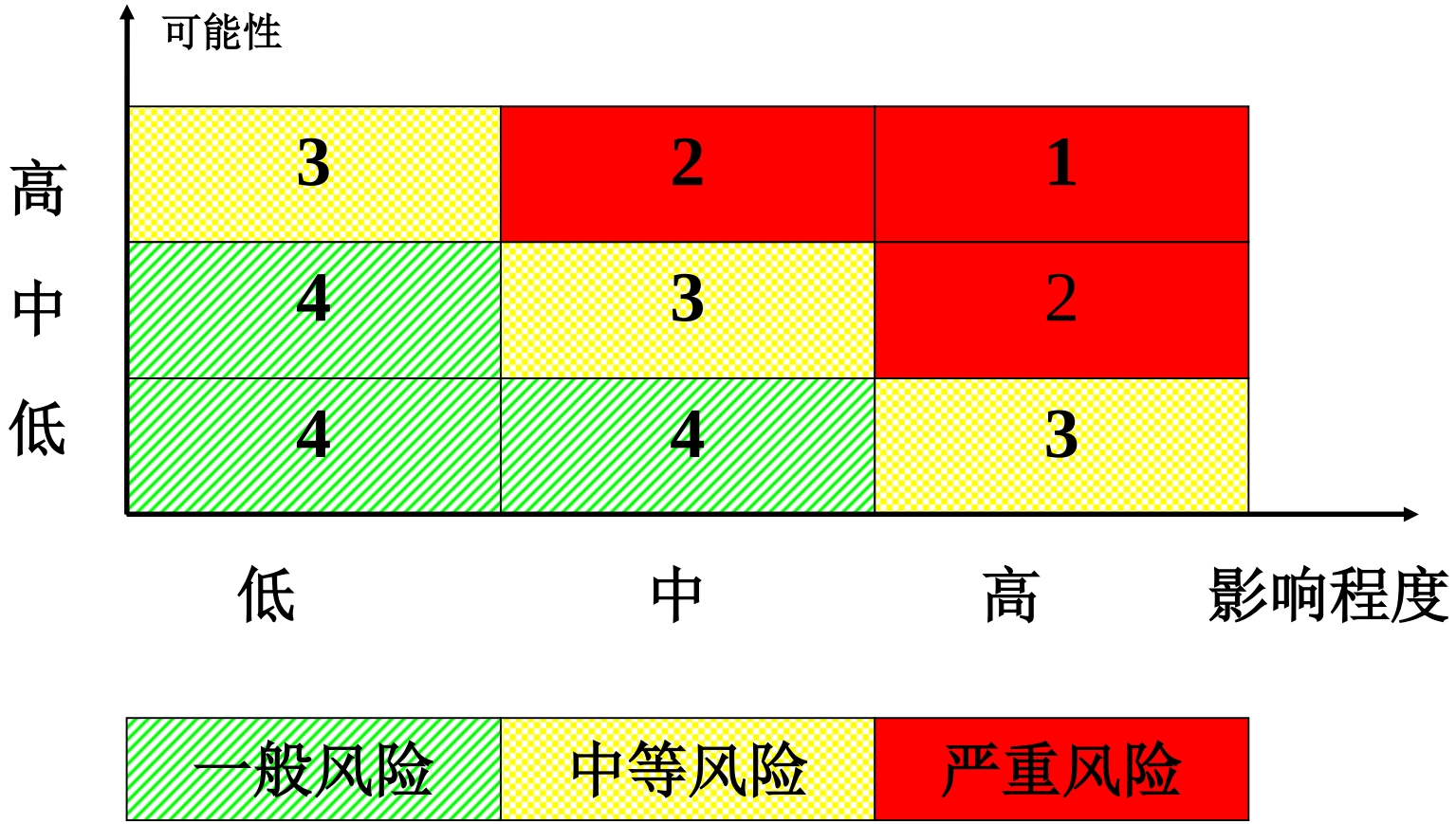
课程目录

1. 为什么引入风险
2. 风险的概念
3. 风险管理过程
4. 基于风险的测试设计
5. 基于风险的测试管理
6. 优点与挑战

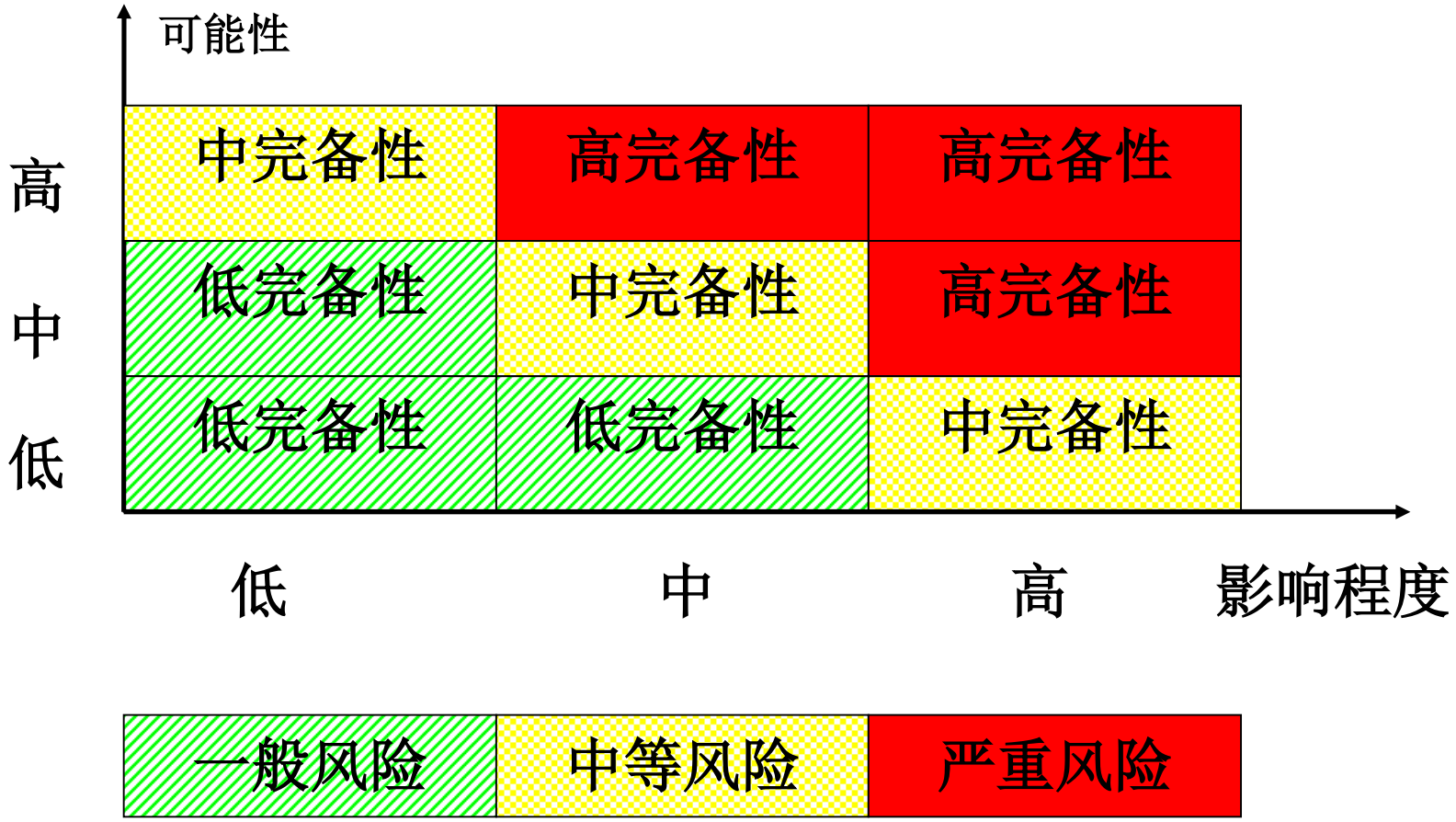
基于风险的测试的优点

- Ø 确定测试优先级
- Ø 确定测试完备性
- Ø 确定测试资源分配
- Ø 易于测试进度监控
- Ø 加速测试信心提升

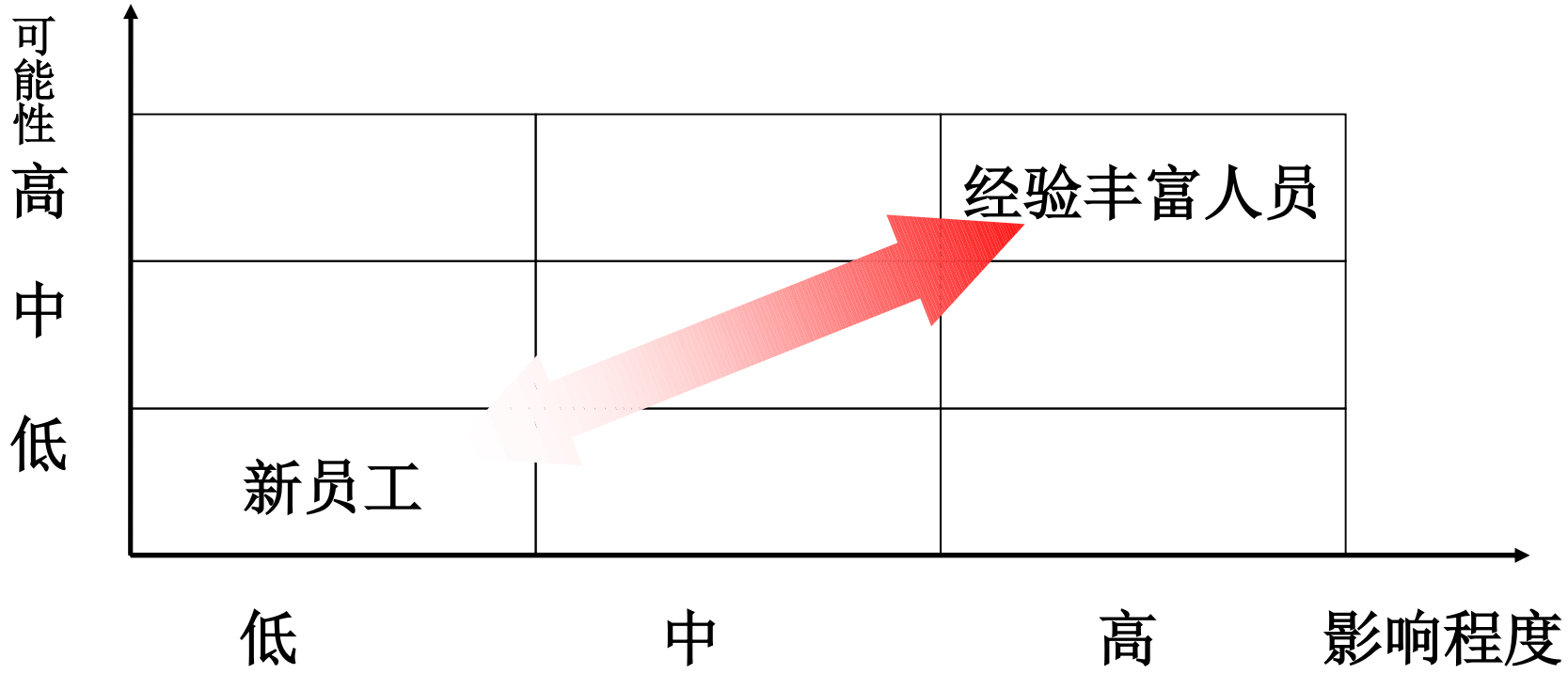
确定测试优先级



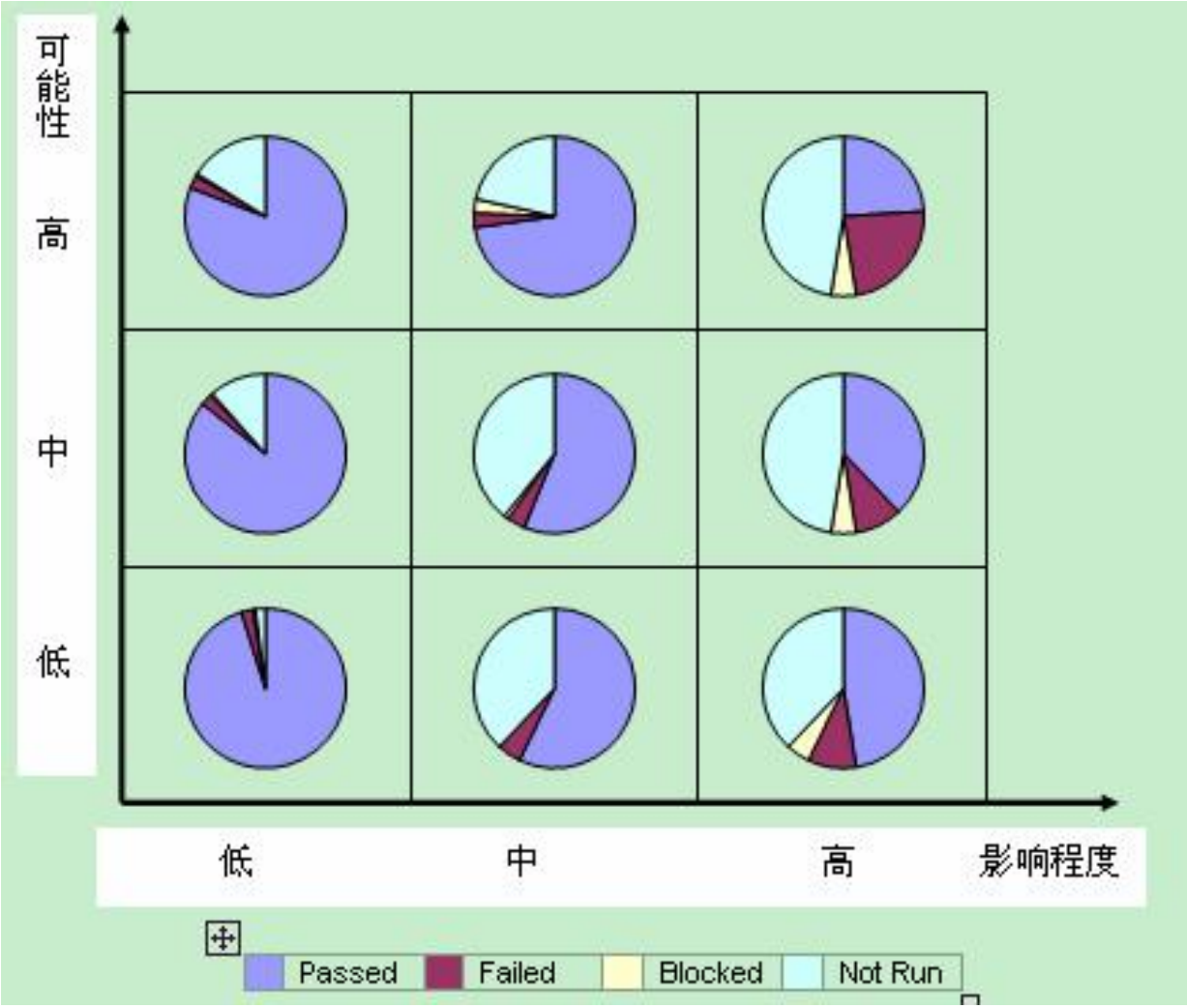
确定测试完备性



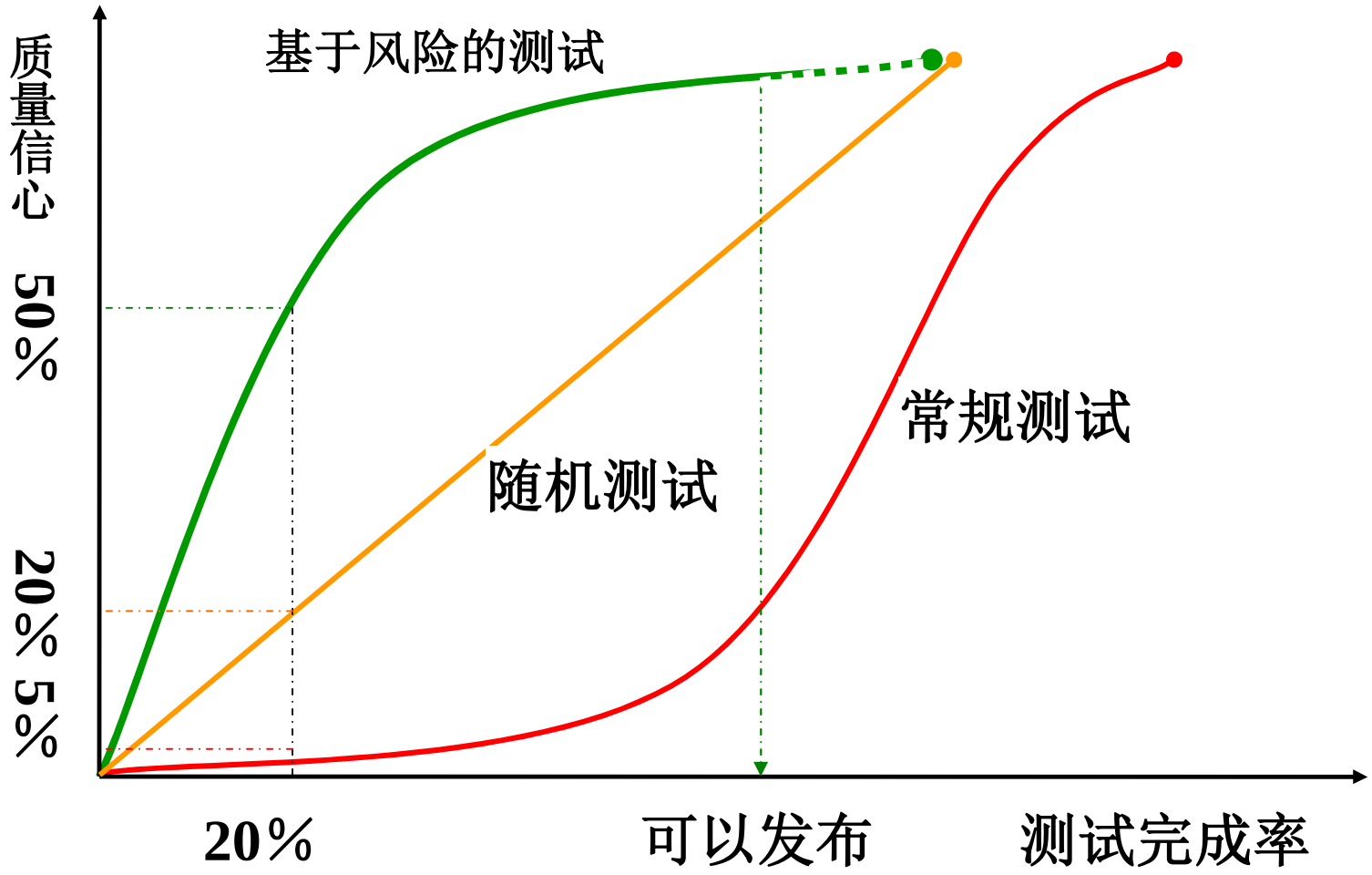
确定测试资源分配



易于测试进度监控



加速测试信心提升



注意点和误区

- Ø 测试风险分析自身的风险
- Ø 完成测试之后的风险
- Ø 基于风险的测试是管理人员的事情
- Ø 风险列表不清晰
- Ø 项目成员不愿意谈论风险

小结

1. 为什么引入风险
2. 风险的概念
3. 风险管理过程
4. 基于风险的测试设计
5. 基于风险的测试管理
6. 优点与挑战


Q & A