

可用性

- 可用性与系统故障和故障导致的结果相关
- 大部分IT系统的关键需求
- 通过可用时间的比例来度量
 - 工作时间100%可用
 - 每周计划停机时间不超过两小时
 - 24x7x52 (100%可用)
- 与应用程序的可靠性相关
 - 不可靠的应用程序可用性很差

可用性

- 可用性的损失周期:
 - 检测失败的时间
 - 纠正失败的时间
 - 重启应用的时间
- 高可用性策略
 - 消除单点故障
 - 复制与故障转移
 - 自动检测与重启
 - 在应用程序/系统故障后重新建立性能水平并恢复受影响数据的能力。
- 可恢复性

可用性

- 关注点:
 - 系统故障如何被检测到
 - 系统故障发生的频率
 - 系统故障时会发生什么
 - 允许系统停止运行多长时间
 - 什么时候故障会安全发生
 - 故障如何被避免
 - 故障发生时需要什么类型的通知

通用可用性场景表

场景组成部分	可能取值
激励源	系统内部; 系统外部
激励	故障: 遗漏, 崩溃, 定时, 响应
工件	系统的处理器, 通信通道, 持久存储过程
环境	正常操作模式, 降级模式 (即更少的功能, 备用解决方案)
响应	系统应当检测事件并能执行以下一项或几项操作: <ul style="list-style-type: none">记录通知相关方, 包括用户和其他系统禁用导致错误或失败的事件源在预先设定的时间间隔内不可用继续以正常或降级模式运行
响应度量	系统必须可用的时间区间; 可用时间; 系统处于降级模式的时间区间; 修复时间

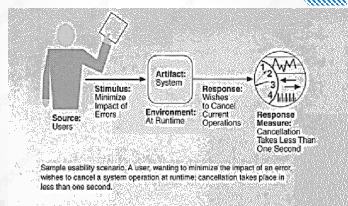
易用性

- 易用性涉及用户完成任务的容易程度以及所提供的用户支持类型。
- 易用性:
 - 学习系统功能
 - 有效使用系统
 - 最小化错误影响
 - 系统适应用户需求
 - 提高用户信心和满意度

通用易用性场景表

场景组成部分	可能的取值
激励源	用户
激励	学习系统功能; 有效使用系统; 最小化错误影响; 适应系统; 感觉舒服
工件	系统
环境	运行时或配置时
响应	系统提供以下一个或多个响应: <ul style="list-style-type: none">支持“学习系统功能”——帮助系统对上下文敏感; 用户熟悉界面; 界面在不熟悉的情况下使用支持“有效使用系统”——数据/命令的集合; 重用已经输入的数据/命令; 支持屏幕内的离屏导航; 具有一致操作的清晰视图; 全面的搜索; 最小化错误的影响; 取消、取消、从系统故障恢复; 识别并纠正用户错误; 检查忘记密码; 验证系统资源支持“适应系统”——定制个性化支持“感觉舒服”——显示系统状态; 按照用户的步骤工作
响应度量	任务时间; 错误数量; 解决问题的数量; 用户满意度; 用户知识的获取; 成功操作数占总操作数的比率; 时间/数据丢失量

易用性场景例子



设计权衡

- » 质量属性是很少正交的，它们相互作用并相互影响。
- » 软件质量属性是系统非功能性需求的一部分
- » 解决方案要进行合理的折衷：
 - 不可能完全满足所有相互竞争的需求
 - 一定要满足所有涉众的需求