



第20章 评审技术

- 何为评审？
 - 技术人员为技术人员召开的会议
 - 对软件工程过程中产生的工作产品进行技术评估

- 一种软件质量保证机制
- 训练场
- 错误与缺陷

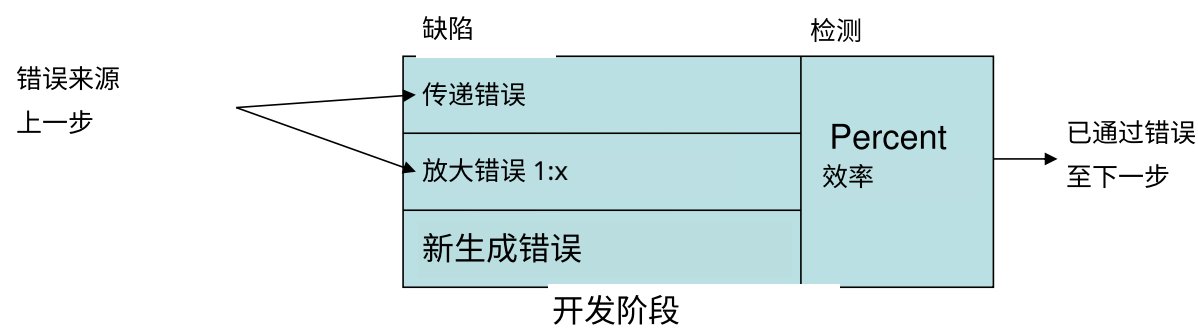
•错误-软件发布给最终用户前发现的质量问题 | 缺陷-软件发布给最终用户后才发现的质量问题 | 注：本书对错误与缺陷的时间划分并非主流观点

1/16



20.2 缺陷放大与消除

- 缺陷放大模型



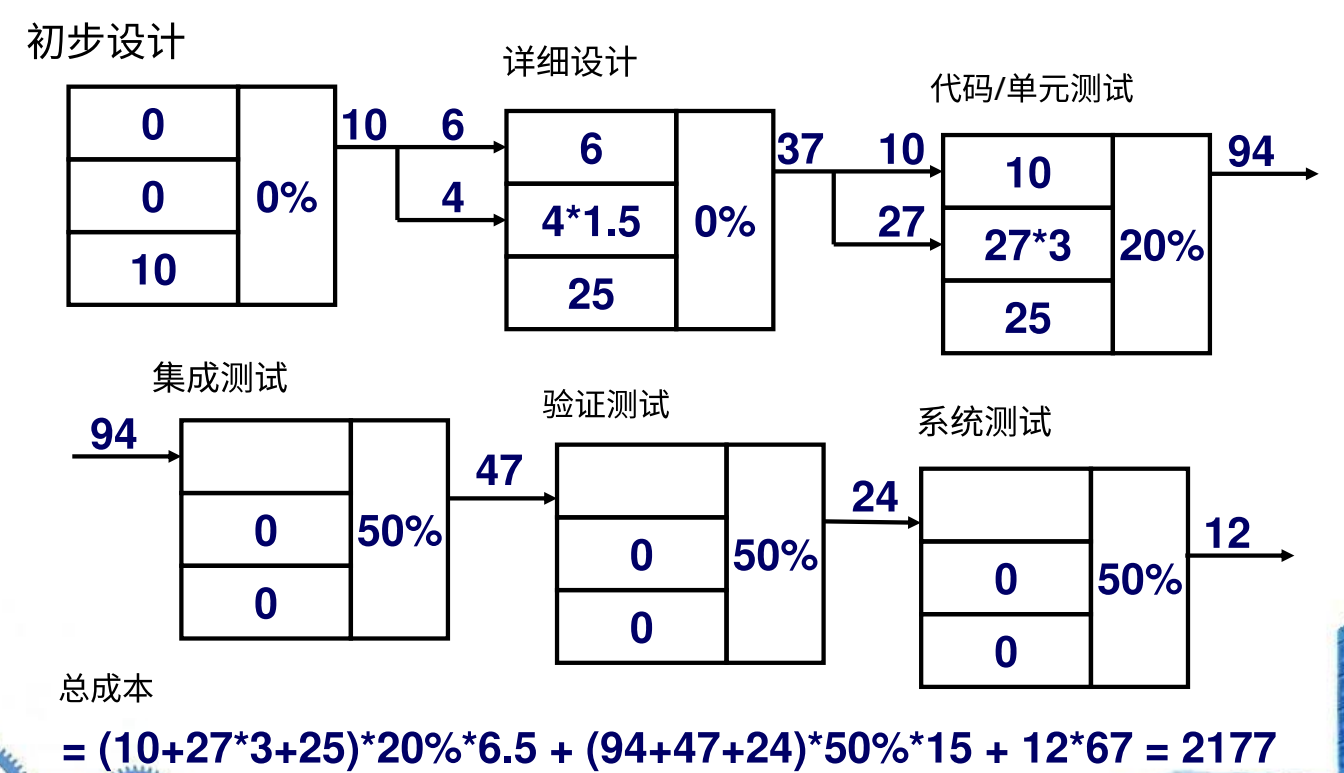
•假设设计阶段发现的错误修正成本为1.5货币单位。相比而言，测试启动前发现的相同错误修正成本升至6.5单位；测试阶段达15单位；发布后则高达67-100单位。- 多项研究表明，软件过程中50%-65%的错误源自设计活动。但形式化评审技术可有效发现75%的设计缺陷。



2/16



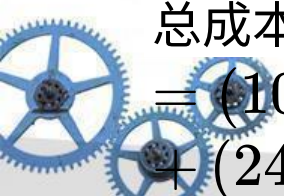
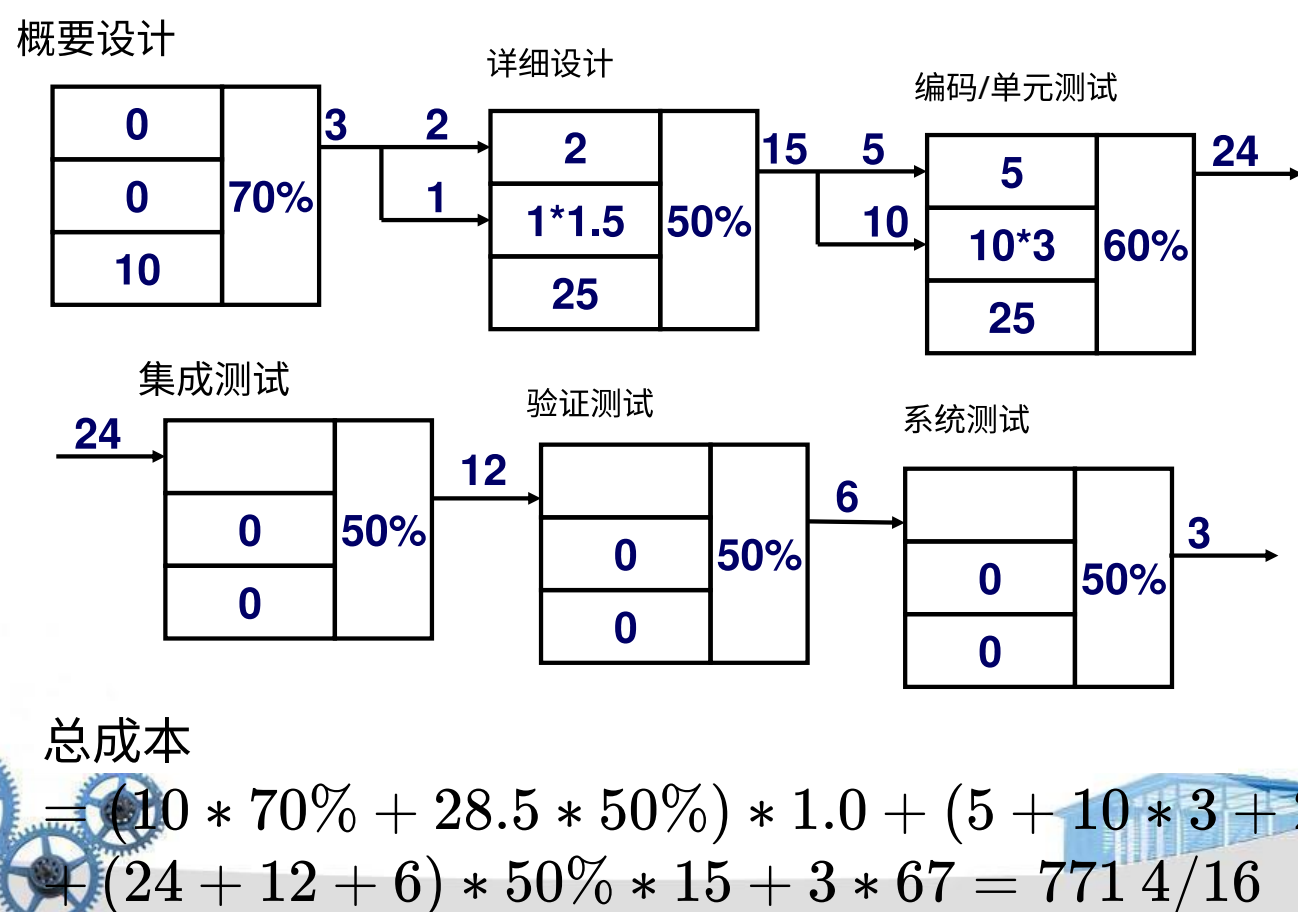
20.2 缺陷放大与消除- 示例：无评审的缺陷放大



3/16



20.2 缺陷放大与消除-示例：评审环节的缺陷放大效应



4/16

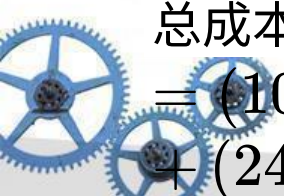


20.3 评审指标及其应用

- 总评审工作量和发现的错误总数定义为：-

$$E_{\text{review}} = E_p + E_a + E_r$$

- $\text{Err}_{\text{tot}} = \text{Err}_{\text{minor}} + \text{Err}_{\text{major}}$ - 缺陷密度表示每单位评审工作产品中发现的错误数。- 缺陷密度 = $\text{Err}_{\text{tot}} / \text{WPS}$ - 准备工作量， E_n - 在实际评审会议前评审工作产品所需投入的工作量（以人时计）- 评估工作量， E_a - 实际评审过程中消耗的工作量- 返工工作量， E_r - 用于纠正评审期间发现的错误所投入的工作量- 工作产品规模，WPS - 已评审工作产品的规模度量（例如UML模型数量或文档页数）- 发现的次要错误， $\text{Err}_{\text{minor}}$ - 可归类为次要的错误数量（纠正所需工作量低于预设阈值）- 发现的主要错误， $\text{Err}_{\text{major}}$ - 可归类为主要错误的数量（纠正所需工作量超过预设阈值）



5/16



评估节省：案例研究-I

- 修正轻微模型错误（评审后立即处理）所需工作量为4人时
- 重大需求错误修正需耗费18人时
- 分析评审数据发现，轻微错误发生率是重大错误的6倍。由此可估算评审阶段发现并修正需求错误的平均工作量约为6人时
- 测试阶段发现的需求类错误，平均需45人时进行定位修正。综合上述平均值可得：
 - 单错误节省工作量 = $E_{\text{testing}} - E_{\text{reviews}}$
 $45 - 6 = 30$ 人时/错误
 - 由于在需求模型评审中发现22个错误，可节省约660人时的测试工作量——这仅是需求相关错误带来的节省。



6/16

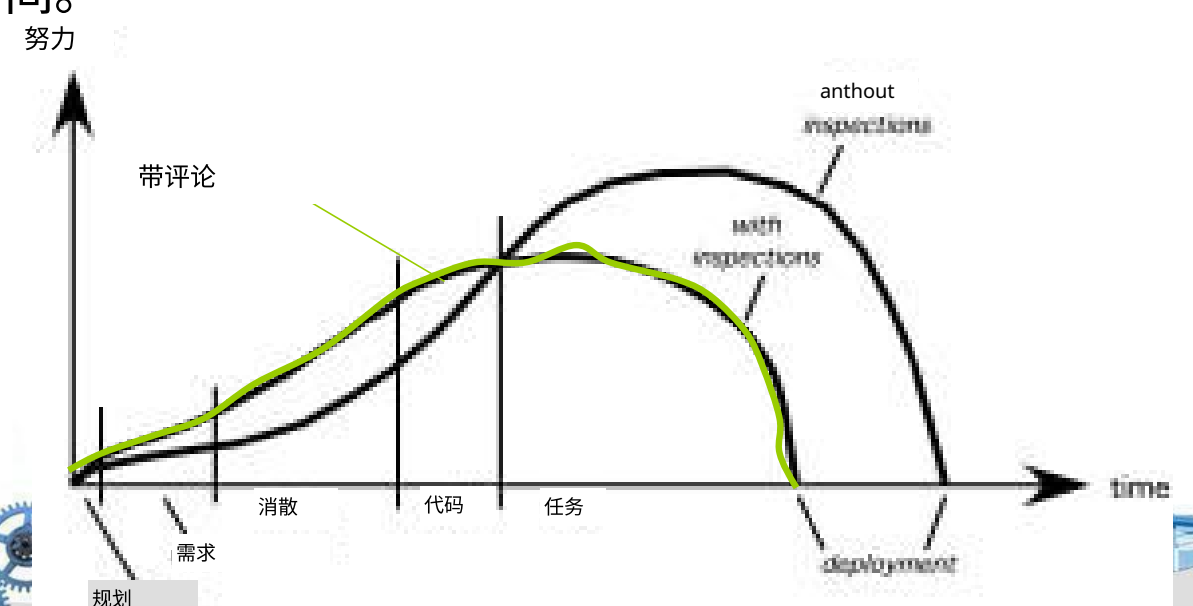


20.3 评审指标

- 实施与不实施评审的投入对比

•采用评审虽会增加前期投入，但因减少测试与修正工作量，这笔早期投资将获得回报。

•采用评审的开发周期比不评审更短。评审不耗时间，而是节省时间。



7/16



工作绩效预测：案例二

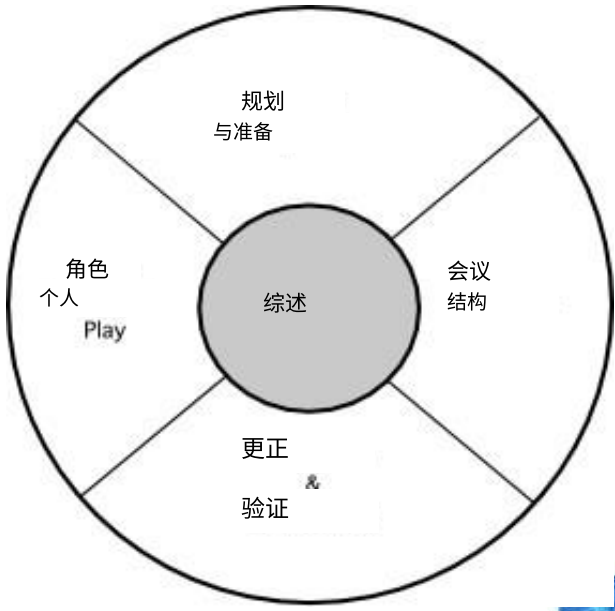
- 若历史数据表明
 - 需求模型的平均缺陷密度为每页0.6个错误，且新需求模型长达32页
- 粗略估算显示您的团队在文档评审中可能发现约19至20个错误。 - 若仅发现6个错误，说明需求模型开发极为成功，或评审方法不够彻底。



20.4 参考模型

- 评审正式性提升的情形：

- 评审员的具体职责已明确定义。
- 评审工作已进行充分规划与准备- 明确定义了评审的具体结构- 评审人员会对所有修改进行跟踪跟进



8/16

9/16



20.5 非正式评审

- 非正式评审包括：
 - 与同事进行的简单桌面检查软件工作产品- 为审查工作产品而举行的非正式会议（超过2人参与） - 结对编程中面向评审的环节- 结对编程能在创建工作产品（设计或代码）时促进持续评审- 其优势在于能及时发现错误，从而提高工作产品质量



20.6 正式技术评审

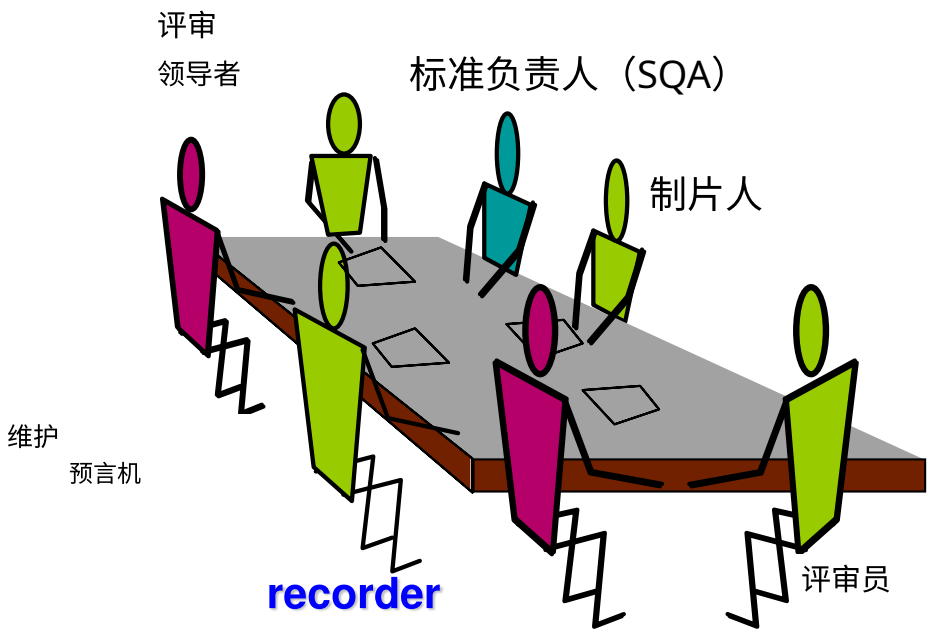
- 正式技术评审的目标是：
 - 发现软件任何表现形式中的功能、逻辑或实现错误- 确认被评审软件符合需求要求- 确保软件表现形式遵循预定义标准- 实现开发过程标准化- 提升项目可管理性- 正式技术评审实际包含走查和审查等多种评审类型



20.6.1 评审会议

- 通常需要3至5人参与评审
- 需提前准备，但每人工作量不超过2小时
- 评审会议时长应少于2小时
- 聚焦于工作成果（如需求模型片段、详细组件设计、组件源代码）

参与者



12/16

13/16



流程

- 参与者角色
 - 生产者——工作成果的开发者
 - 通知项目负责人工作产品已完成且需进行评审必需
 - 评审负责人-评估产品就绪状态，生成副本将产品材料分发给两至三位评审人员用于预先准备。
 - 审稿人预计耗时1至2小时审阅产品、做笔记并逐步熟悉该工作内容。
 - 记录员-评审员：负责书面记录评审过程中提出的所有重要问题。

流程

- 准备阶段：生产者->评审组长->评审员->问题列表
- 执行阶段：生产者介绍->评审员提出问题->记录员
- 跟踪阶段：结论，SQA报告；



20.6.3 评审指南

- 评审产品而非生产者-设定议程并保持-限制争论与反驳
- 阐明问题领域，但无需解决所有指出的问题。
- 做好书面记录。
- 限制参与者数量并坚持预先准备。 - 为每个可能评审的产品制定检查清单。 - 分配资源并安排正式技术评审时间。 - 为所有评审人员提供实质性培训。回顾早期评审结果。

14/16

15/16



20.6.4 样本驱动评审(SDRs)

•SDR旨在量化那些需要完整正式技术评审(FTR)的主要工作产物
为实现此目标...

•抽查每个软件工作产物的 a_i 比例样本 i ，记录在 a_i 中发现的缺陷数量 f_i

•通过将 f_i 乘以 $1/a_i$ ，推算工作产物 i 中的缺陷总量

•根据各工作产物的缺陷总量估算值进行降序排序

•将有限的评审资源集中于预估缺陷数量最多的工作成果上。

