

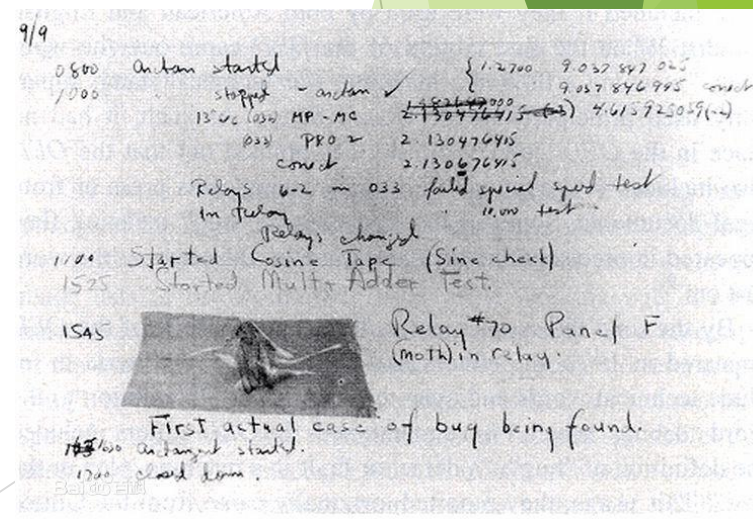
# 软件缺陷的定义 (上)



# BUG

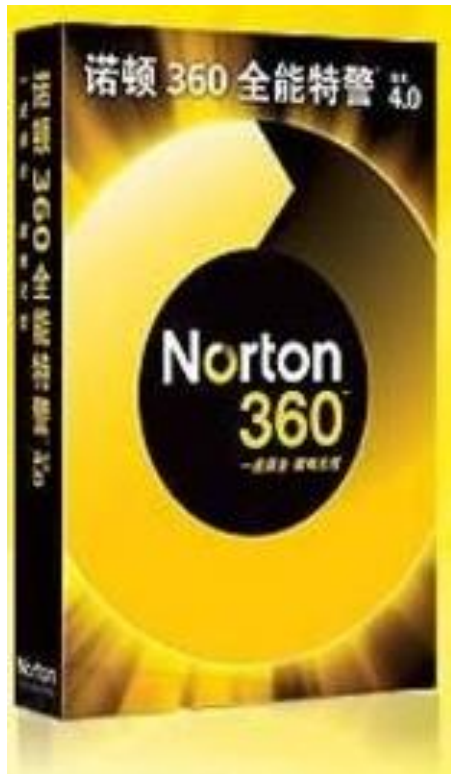
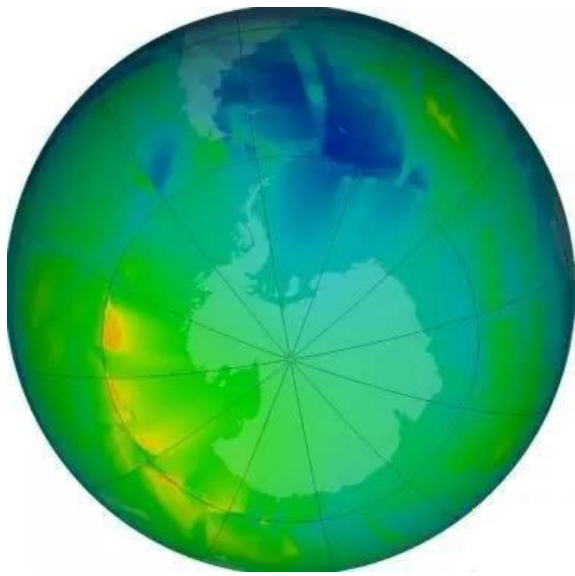


- ▶ Grace Hopper, 计算机软件之母
- ▶ 1945年9月9日
- ▶ “First actual case of bug being found”  
(第一个发现虫子的实例)



# 软件缺陷案例不胜枚举

- ▶ 1978年，NASA臭氧层测绘数据分析软件的缺陷
- ▶ 1982年，西伯利亚天然气管道控制软件缺陷
- ▶ 1990年，AT&T大型交换机软件的缺陷
- ▶ 2007年，赛门铁克公司的诺顿杀毒软件缺陷
- ▶ 2009年，谷歌公司的gmail缺陷
- ▶ 2011年，亚马逊的云计算中心宕机
- ▶ 2012年，骑士资本的部署缺陷







2012.12

2012年12月11日，Knight Capital Group Inc. 的一个软件故障导致它在一天内损失了4.4亿美元。

# 软件缺陷的正式定义

- 1. 软件测试员认为软件难以理解、不易使用、运行速度缓慢，或者最终用户认为不好，则是缺陷。

有BUG



开发的产品



VS



用户要的产品



站在用户角度



# 一个例子：QQ圈子

- ▶ 2012年3月推出
- ▶ 按真实生活中的关系自动分圈
- ▶ 最贴合“不忘老朋友、结识新朋友”的需求
- ▶ 2014年11月下线



# 问题1：需求模糊不清

► 对需求进行书面约定：需求规格说明



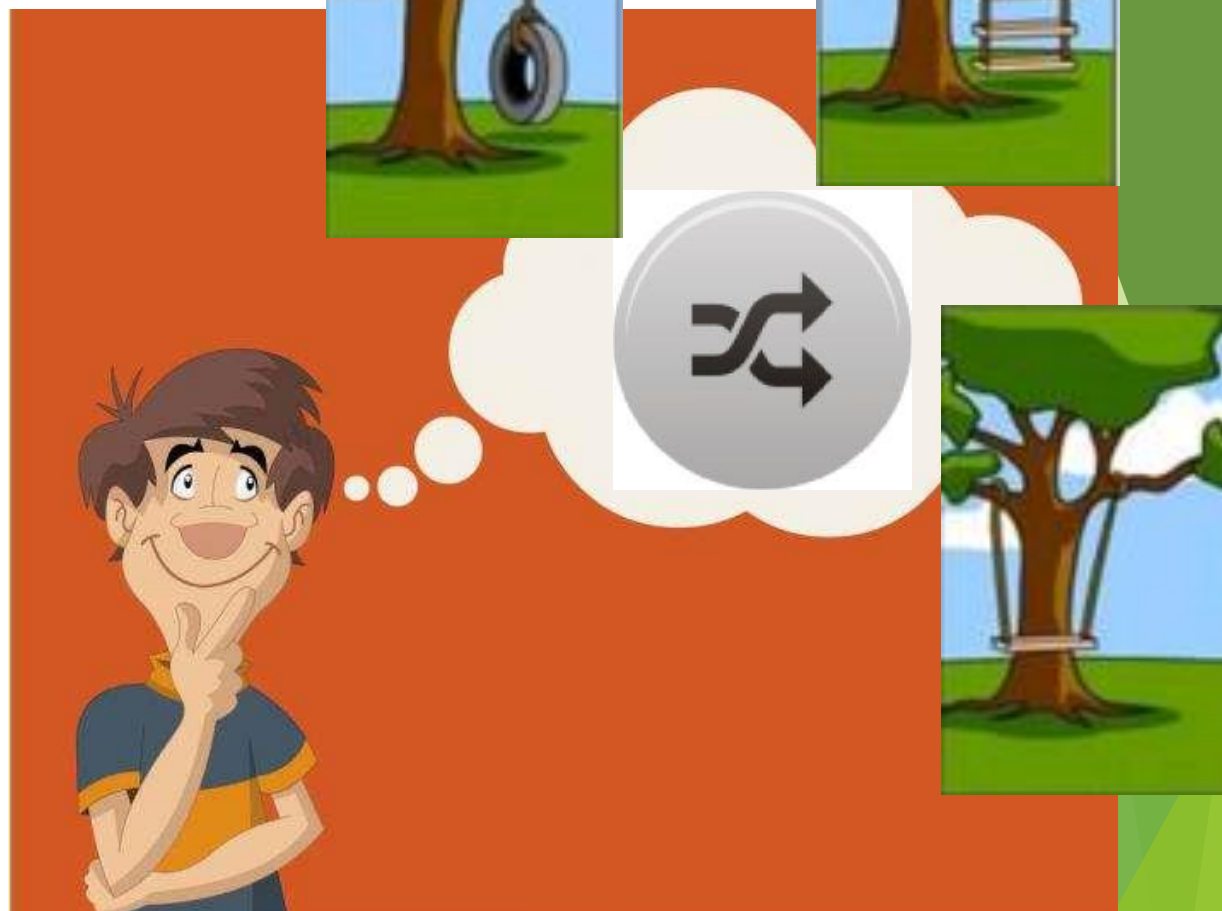


## 问题2：需求变化无常

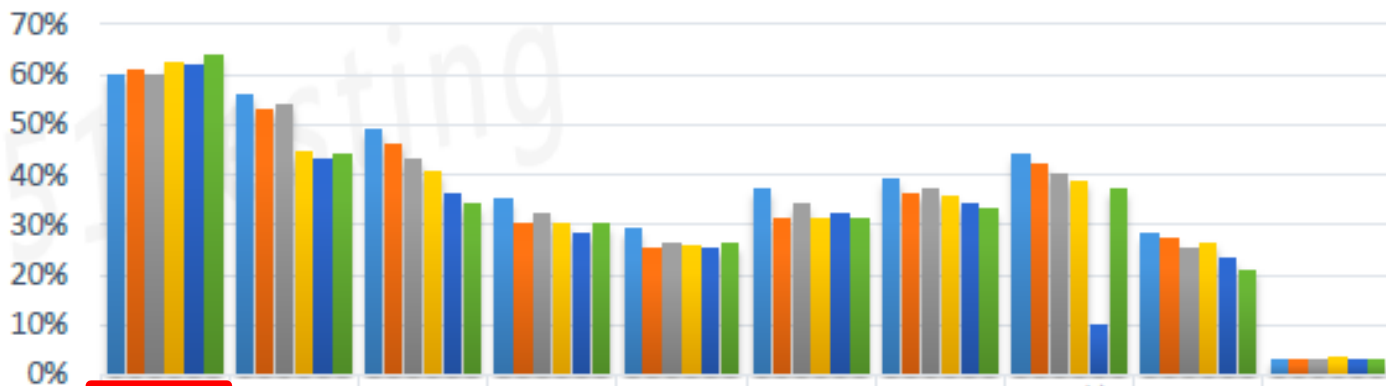


## 问题2：需求变化无常

► 必须确保需求的稳定



## 历年软件测试从业人员在测试工作中存在的障碍分布



A: 测试人员获得需求不够完整、不清晰、不规范

B: 需求管理不好, 导致测试资产和需求无法保持一致

C: 缺少前端测试分析、测试设计过程, 导致测试覆盖不充分

D: 缺少单元测试自动化手段, 导致无法有效开展单元测试

E: 缺少接口测试自动化手段, 导致无法有效开展模块测试

F: 缺少系统测试自动化测试手段, 导致系统测试执行效率低

G: 缺少自动回归测试手段, 导致版本频繁时, 无法进行有效测试

H: 缺少有效度量、缺陷分析手段, 无法有针对性的对测试工作进行管理和改进

I: 业务知识复杂

J: 其他

	2010年	2011年	2012年	2013年	2014年	2015年
A: 测试人员获得需求不够完整、不清晰、不规范	60%	61%	60%	62%	62%	64%
B: 需求管理不好, 导致测试资产和需求无法保持一致	56%	53%	54%	44%	43%	44%
C: 缺少前端测试分析、测试设计过程, 导致测试覆盖不充分	49%	46%	43%	41%	36%	34%
D: 缺少单元测试自动化手段, 导致无法有效开展单元测试	35%	30%	32%	30%	28%	30%
E: 缺少接口测试自动化手段, 导致无法有效开展模块测试	29%	25%	26%	26%	25%	26%
F: 缺少系统测试自动化测试手段, 导致系统测试执行效率低	37%	31%	34%	31%	32%	31%
G: 缺少自动回归测试手段, 导致版本频繁时, 无法进行有效测试	39%	36%	37%	36%	34%	33%
H: 缺少有效度量、缺陷分析手段, 无法有针对性的对测试工作进行管理和改进	44%	42%	40%	39%	10%	37%
I: 业务知识复杂	28%	27%	25%	26%	23%	21%
J: 其他	3%	3%	3%	3%	3%	3%





# 测试不是银弹，而是要进行软件质量控制



测试人员为什么没有发现这个bug?