



软件系统测试

清华大学软件学院 刘强





1

集成测试

2

功能测试

3

性能测试

4

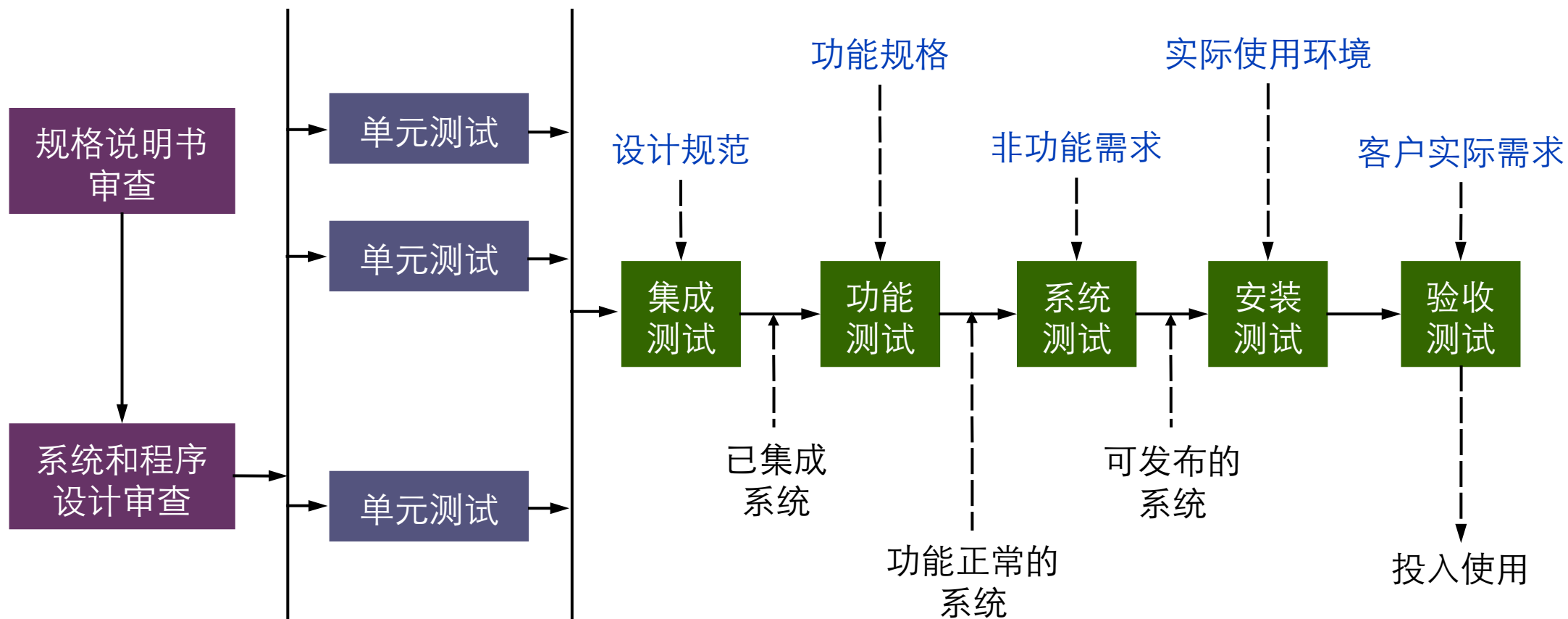
软件缺陷管理

1



- 集成测试概念
- 基于功能分解的集成
- 持续集成
- 集成测试原则

回顾：软件测试阶段



- □ □ □ □ □ □ □ □ □ □ □ □ □ □
- □
- □
- □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

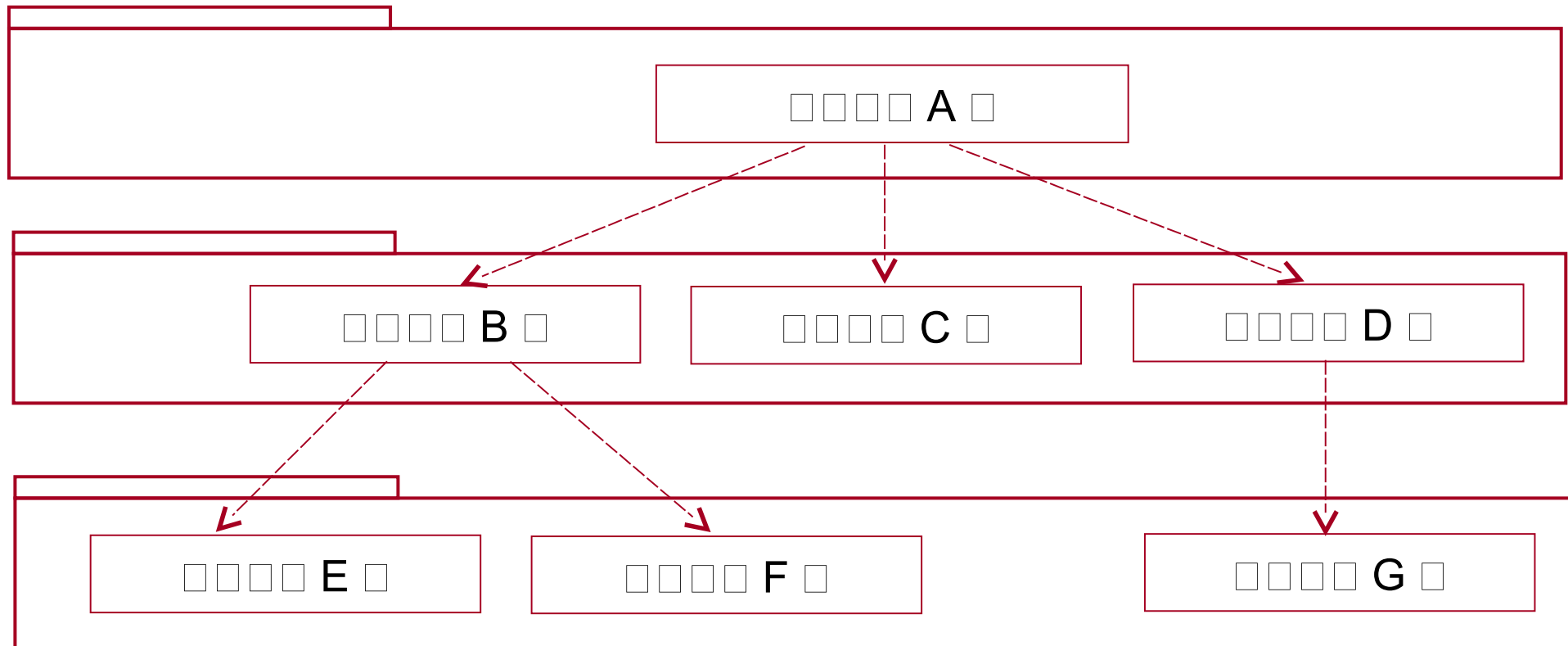


单元测试、集成测试与系统测试

测试类型	对象	目的	测试依据	测试方法
单元测试	模块内部的程序	消除局部模块逻辑和功能上的错误或缺陷	模块逻辑设计 模块外部说明	白盒测试
集成测试	模块间的集成和调用关系	找出和软件设计相关的程序结构、模块调用关系以及模块间接口方面的问题	程序结构	结合使用白盒测试和黑盒测试，较多采用黑盒测试
系统测试	整个系统包括硬件、支持软件、人员等	对整个系统进行一系列整体的、有效性测试	系统结构设计 目标说明书 需求说明书	黑盒测试

集成测试策略

问题：下面的三层结构系统如何进行集成测试？



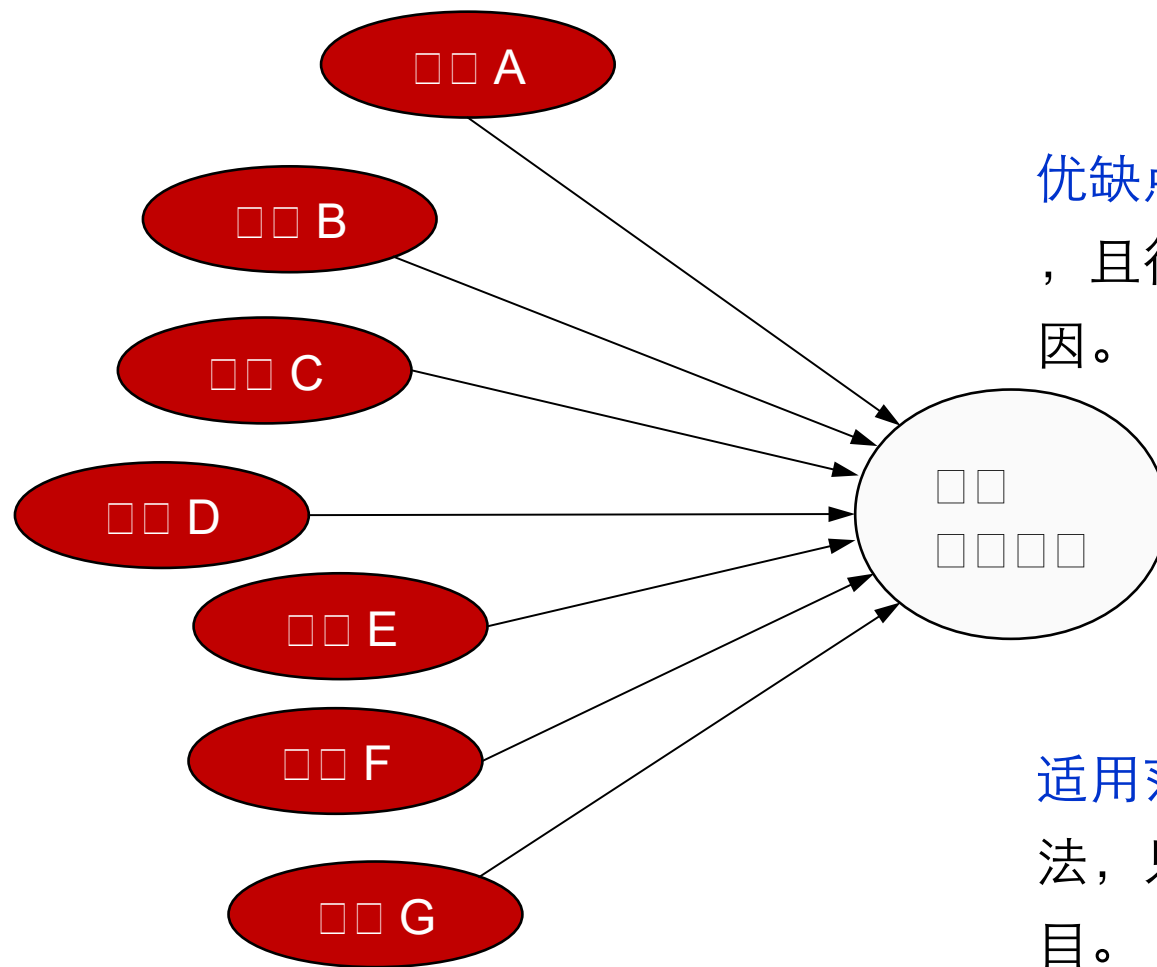
爆炸式集成

- 将所有通过单元测试的模块一次性地集成到一起进行测试，也称为一次性组装或整体拼装。

递增式集成

- 将要测试的模块逐渐集成到已经测试好的模块里面，边集成边测试，测完一部分再连接一部分。

爆炸式集成

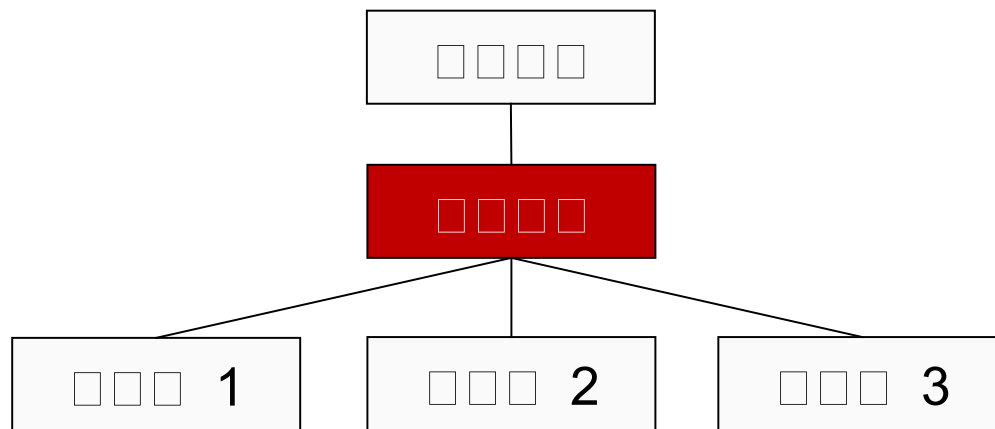


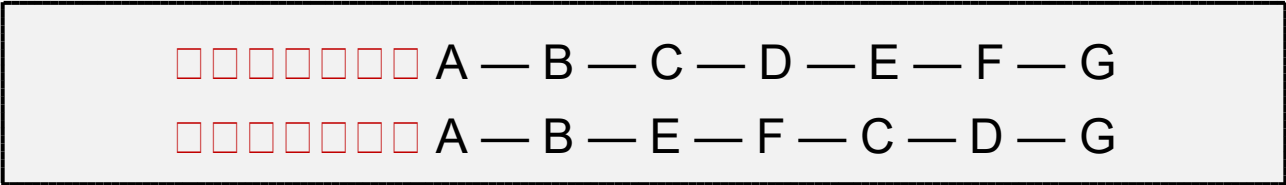
优缺点：操作简单，但发现错误比较晚，且很难确定出错的真正位置和错误原因。

适用范围：一般情况下不建议使用该方法，只适合功能简单、规模小的小型项目。

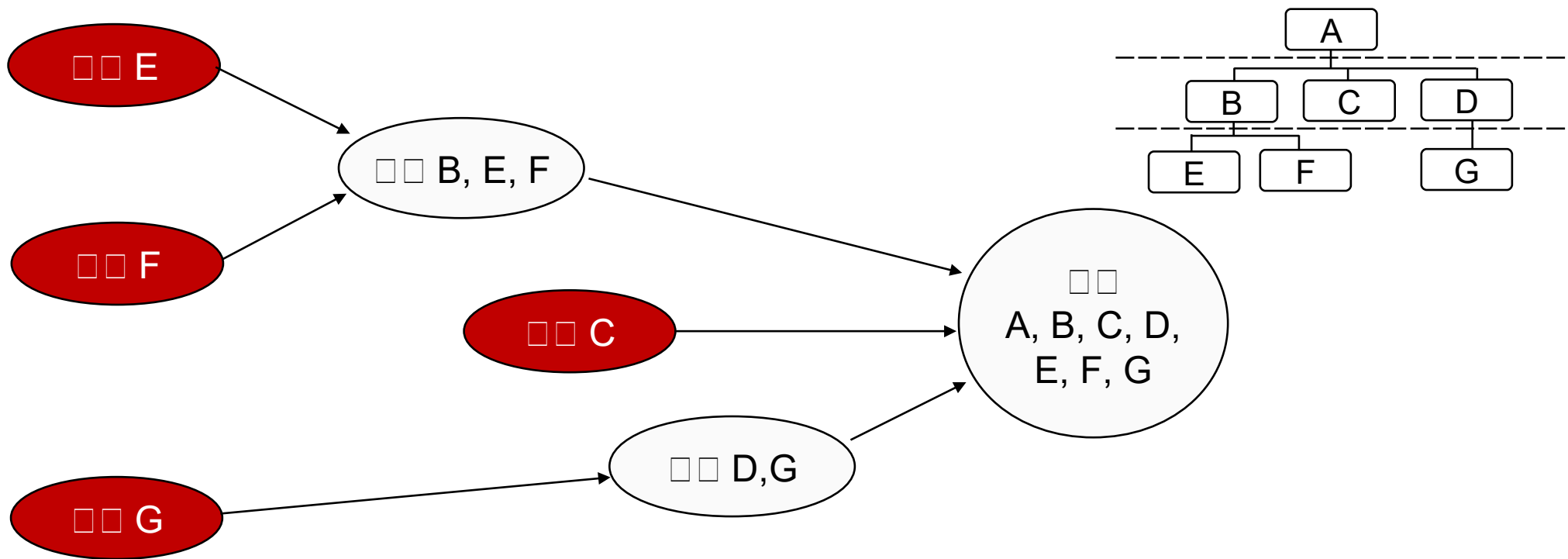
基于功能分解的集成

- **自顶向下集成：**从顶层开始采用同设计顺序一样的思路对被测系统进行测试，一般集中于顶层的组件，然后逐步测试处于底层的组件，被上层单元调用的下层单元以桩出现。
- **自底向上集成：**从最底层组件开始，按照分解树的结构，逐层向上集成，调用下层单元的上层单元以驱动出现。
- **三明治式集成：**这是一种混合增殖式测试策略，它将自顶向下和自底向上的集成方法有机地结合起来，从顶层和底层向中间集成，可以减少桩模块和驱动模块的开发。



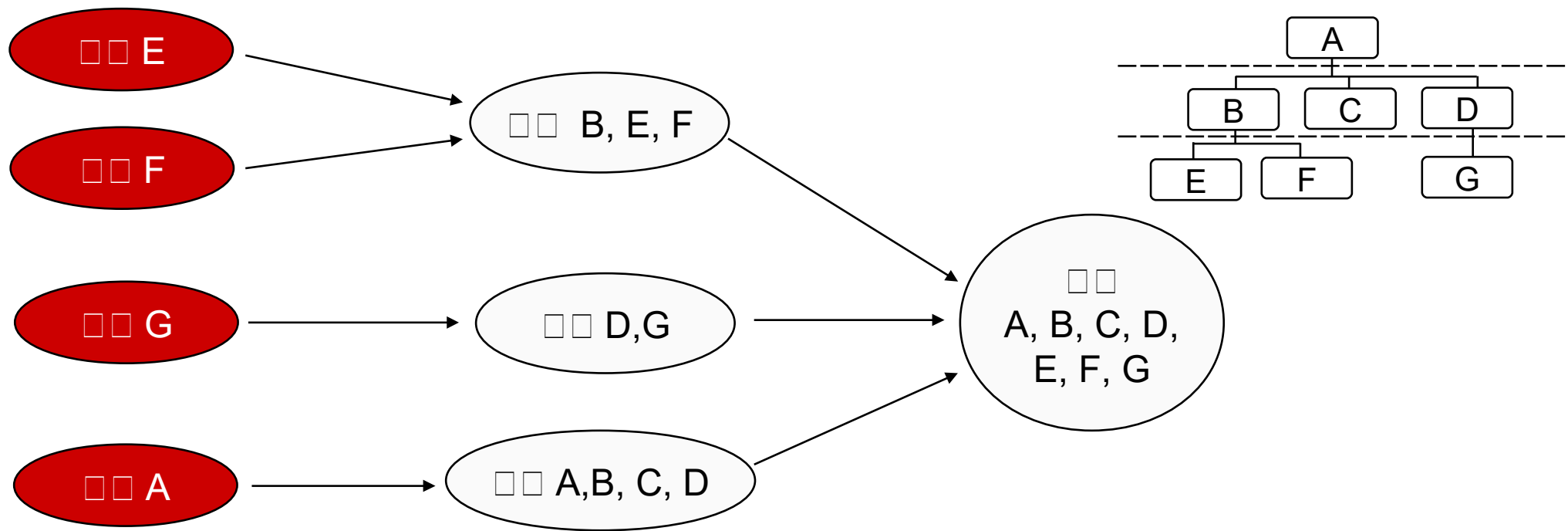
[illegible]

自底向上集成



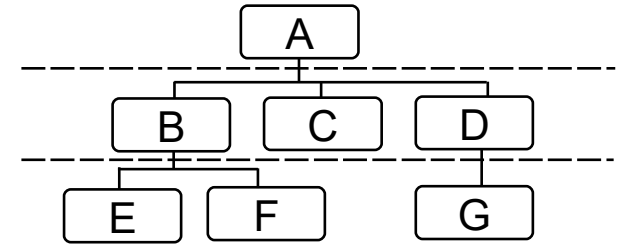
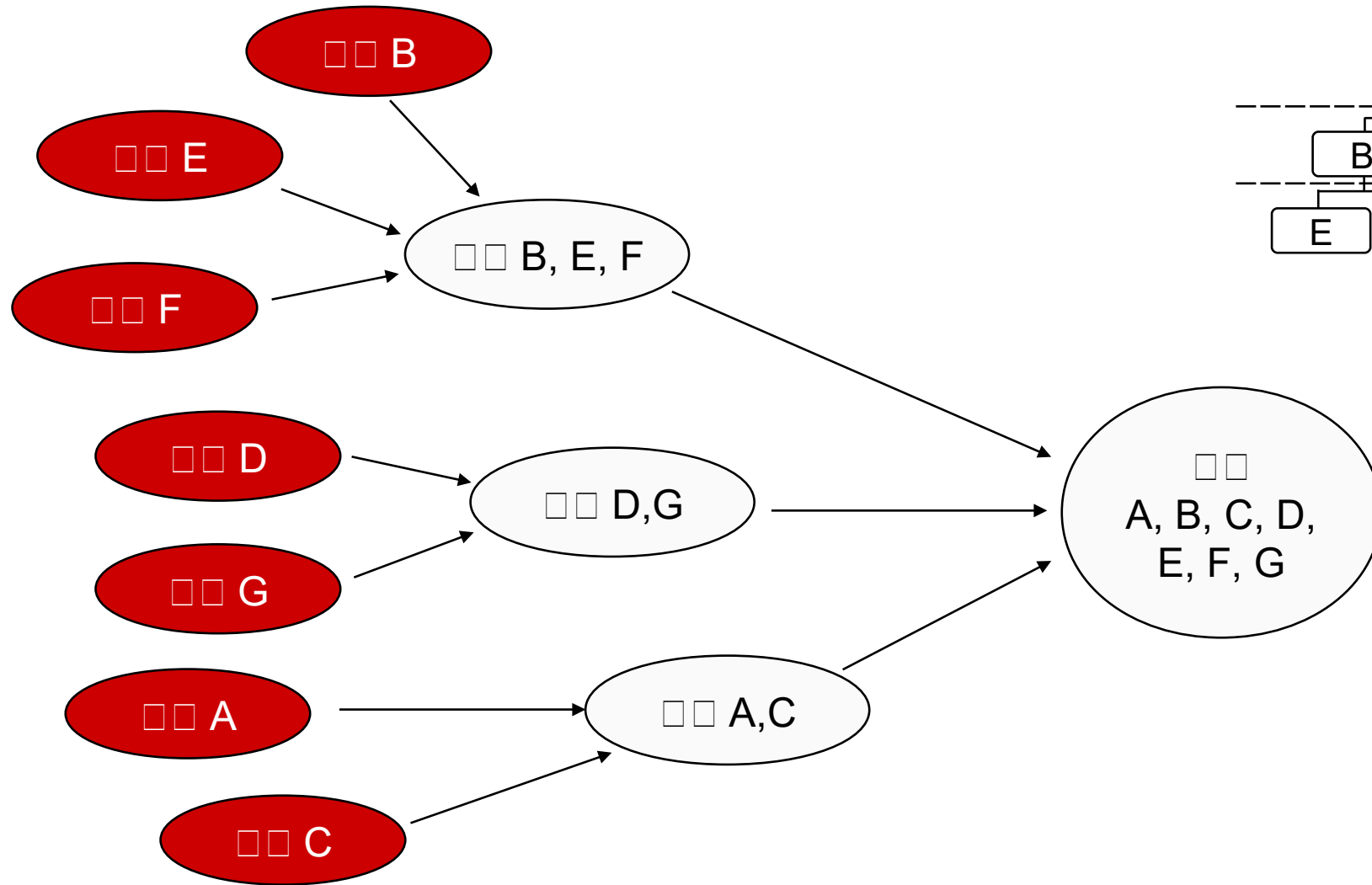
总结：该方法需要驱动模块，可以尽早地验证底层模块的行为，但不能及时发现高层模块设计上的错误；适合底层模块接口比较稳定、高层模块接口变更比较频繁的系统。

三明治式集成

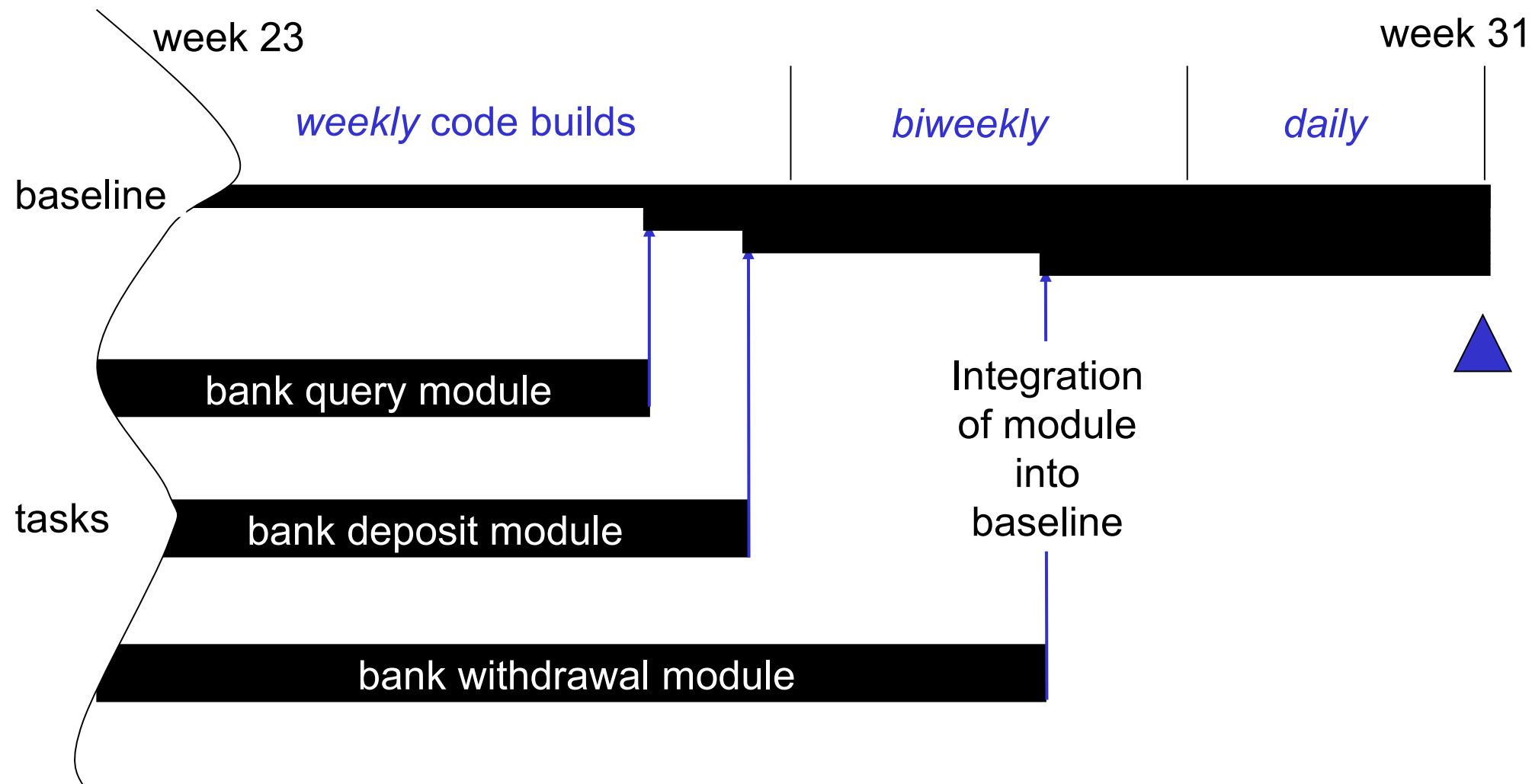


总结: 该方法具有自顶向下和自底向上两种集成策略的优点，运用一定技巧可以减少桩模块和驱动模块的开发，在被集成之前中间层不能尽早得到充分的测试；适用于大多数系统。

改进的三明治集成

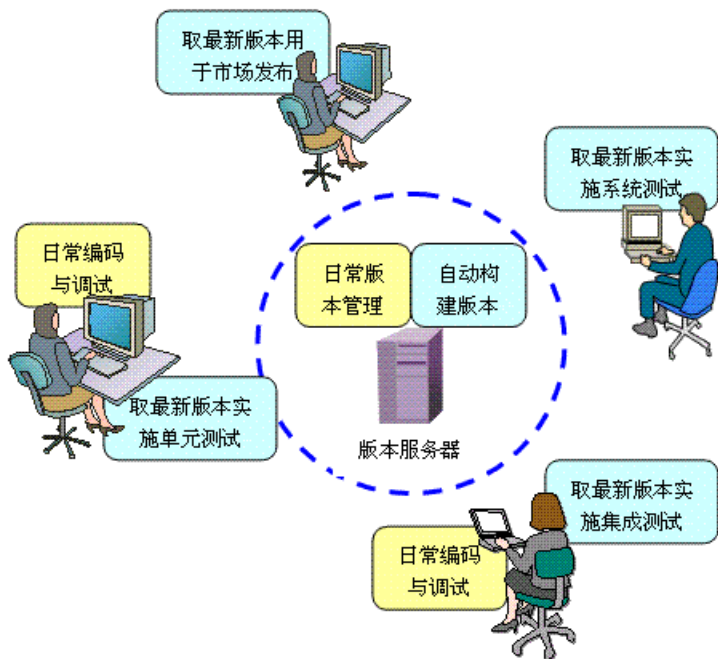


持续集成



持续集成

持续集成是一种软件开发实践，团队开发成员经常集成自己的工作，通常每个成员每天至少集成一次，即每天可能会发生多次集成。



- 所有开发人员需要在本地机器上进行本地构建，然后再提交到版本控制库中，以免影响持续集成
- 开发人员每天至少向版本控制库中提交一次代码，至少从版本控制库中更新一次代码到本地机器
- 需要有专门的集成服务器来执行集成构建，并通过自动化的构建（包括编译、发布、自动化测试）来验证，从而尽快地发现集成错误

手工构建

日常调试通常面对随意的且尚在变化中的测试脚本，经过整理后可成为规范的测试集，直接用于回归测试。

日常调试

自动构建

通常冒烟测试与每日构建配合应用，以每天为单位开展自动测试，运行一些基础用例。

冒烟测试

自动构建

回归测试是阶段性开展的工作，它选用的版本应由构建服务器自动生成，即自动取最新源码并自动构建。

回归测试

集成测试原则

尽早测试关键模块： 关键模块实现了系统的主要功能， 尽早测试有利于提高系统的测试效率。

关键模块的特征：

- 满足某些软件需求
- 在程序的模块结构中位于较高的层次（高层控制模块）
- 较复杂、较易发生错误
- 有明确定义的性能要求

□□□□□□ I/O □□□□□□□□□□□□□□□□

集成测试原则



为了做好集成测试，需要遵循以下原则：

- 所有公共接口都要被测试到
- 关键模块必须进行充分的测试
- 集成测试应当按一定的层次进行
- 集成测试的策略选择应当综合考虑质量、成本和进度之间的关系
- 集成测试应当尽早开始，并以总体设计为基础
- 在模块与接口的划分上，测试人员应当和开发人员进行充分的沟通
- 当接口发生修改时，涉及的相关接口必须进行再测试
- 测试执行结果应当如实记录

2

□ □ □ □

- □ □ □ □ □
- □ □ □ □ □
- Web □ □ □ □ □
- □ □ □ □ □



界面

数据

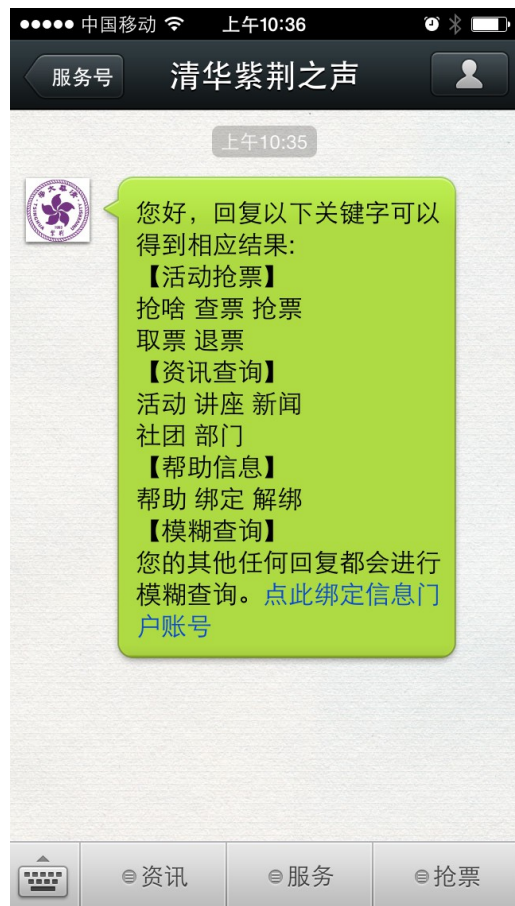
操作

逻辑

接口

- 程序安装、启动正常，有相应的提示框、错误提示等
- 每项功能符合实际要求
- 系统的界面清晰、美观
- 菜单、按钮操作正常、灵活，能处理一些异常操作
- 能接受正确的数据输入，对异常数据的输入有提示、容错处理等
- 数据的输出结果准确，格式清晰，可以保存和读取
- 功能逻辑清楚，符合使用者习惯
- 系统的各种状态按照业务流程而变化，并保持稳定
- 支持各种应用的环境
- 配合多种硬件周边设备
- 软件升级后，能继续支持旧版本的数据
- 与外部应用系统的接口有效

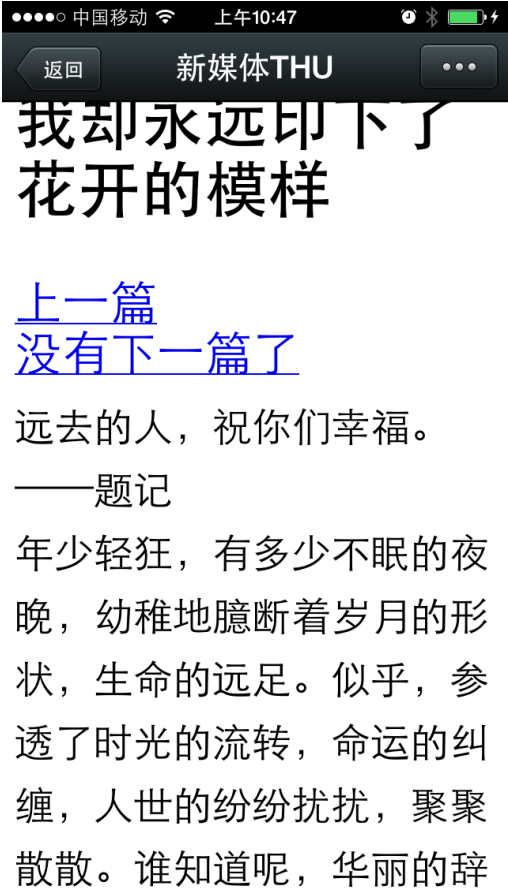
案例分析：团委微信项目



思考：

- 左图是首次关注后系统显示的信息，你有什么看法？
- 作为一个用户，你希望首次关注后看到什么信息？

案例分析：团委微信项目



美观？
清晰？

案例分析：团委微信项目

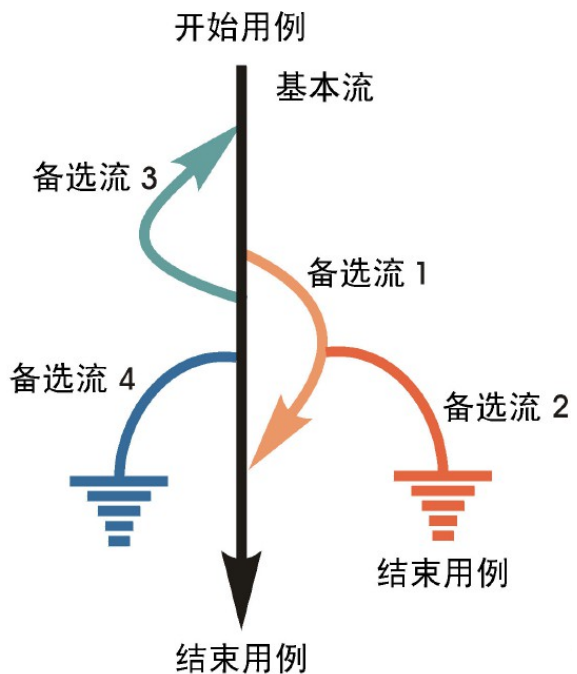


适合性测试

适合性测试是评估和确认软件产品是否能够帮助用户执行期望的任务或者实现了用户需要的功能。



适合性测试应该从客户或用户的角度评判

[illegible]

□□ 1	□□□
□□ 2	□□□□□□□ 1
□□ 3	□□□□□□□ 1 □□□□ 2
□□ 4	□□□□□□□ 3
□□ 5	□□□□□□□ 3 □□□□ 1
□□ 6	□□□□□□□ 3 □□□□ 1 □□□□ 2
□□ 7	□□□□□□□ 4
□□ 8	□□□□□□□ 3 □□□□ 4

-

- 29

案例分析：团委微信项目



□□ 1 □□□□□□□□



□□ 2 □□□□□□□□



□□ 3 □□□□□□□□



□□ 4 □□□□□□□□

案例分析：团委微信项目



- 抢到多张票时，系统如何显示电子票？
- 抢票未开始时是否可以抢票？
- 抢票结束且活动未开始时是否可以抢票？
- 绑定不同账号进行抢票，互相有影响吗？
- 一个活动抢票时会影响其他活动状态吗？

准确性测试

准确性测试是验证软件产品提供具有所需精度的正确或者相符的结果或者效果的能力。



- **预期的准确性：**在运行期间执行任务的实际结果与预期结果的差别。
-
- **计算的准确性：**由于计算本身的错误和数据精度方面的问题导致的错误计算结果。

准确性测试



□□□□□□□□□□“□□□□”

商品名	单件积分	市场单价	网站单价	优惠	数量	删除
WENGER双肩电脑包	0	¥620.00	¥290.10	减¥91.10	1	删除
空间大师DPC003金属四层车	0	¥256.00	¥100.00	减¥15.00	1	删除
欧姆龙电子血压计HEM-8102	269	¥438.00	¥269.00	无	1	删除
爱国者 aigo (4GB) U盘 8380	0	¥99.00	¥69.00	减¥4.00	2	删除
好孩子木餐椅MY303A-E531绿色	339	¥369.00	¥339.00	无	1	删除
讲故事学折纸 (全3册)	399	¥65.40	¥39.90	无	2	删除
您共节省：¥910.00 可获得商品积分：1406		商品金额总计：¥1101.80		结算->		

互操作性测试

互操作性测试是验证软件产品在多种指定的目标环境下（如硬件、软件、中间件、操作系统、浏览器等）是否可以正常工作。

- 输入互操作性：软件产品可以正常使用与之集成的其他软件产品或系统的输出（数据或协议格式等）。
- 输出互操作性：软件产品必须输出标准或协议规定的格式。
- 软件产品是否具有自动检测与该软件交互的系统的通信请求并作出相应的处理。



案例分析：团委微信项目



- 与学校信息门户连接（用户认证）
- 与学生清华系统连接（抢票结果）
- 不同品牌的手机
- 不同的操作系统
- 与扫码器连接（电子票、纸质票）
- 与打印机连接（打印电子票）

安全性测试

安全性测试是查找软件自身程序设计中存在的安全隐患，并检查软件产品对非法侵入的防范能力。

- 对应用程序或数据进行未授权的复制
- 未授权的访问控制
- 缓存区溢出
- 服务拒绝
- 在网络上窃听数据传输来获得敏感信息
- 破解保护敏感信息的加密代码
- 逻辑炸弹



案例分析：清华大学网络学堂



清华大学
Tsinghua University

网络学堂
Web Learning

大字版本 | 学堂公告

教师工作室

课程公告

课程信息

课程文件

教学资源

课程作业

课程答疑

课程讨论

自由讨论区

学生信息管理

学生界面浏览

学生界面管理

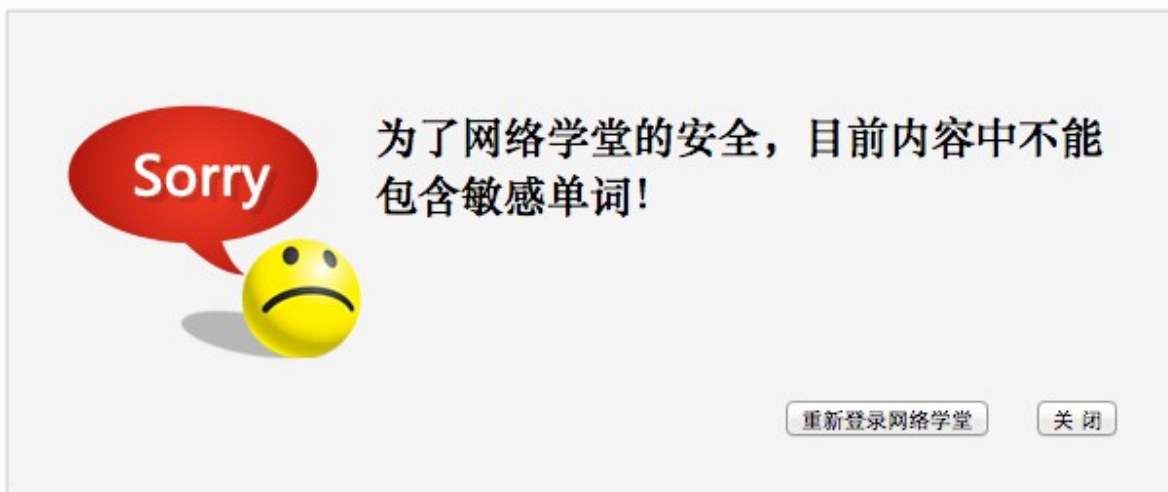
软件工程 (3) (0)(2013-2014秋季学期)

发表话题

一般讨论区 | 开发团队博客 | 分类管理

讨论主题	作者	回复/点击	发表时间	讨论管理
求交互设计课的PPT	文庆福	2/37	2013-11-13 17:58	删除
为什么软件开发周期总是预估的2~3倍?	刘强老师	4/75	2013-10-20 18:05	删除
如何向开源社区提问题	刘强老师	2/70	2013-10-05 09:58	删除
关于10.9预习材料的一些建议	钱康来	1/126	2013-10-02 14:20	删除
IBM黑衣小组	刘强老师	-821/403	2013-10-02 11:20	删除
为什么移动端更适合扁平化而非拟物化设计	文庆福	1/70	2013-09-27 15:52	删除
分享: 编码、设计和测试规范	刘强老师	1/43	2013-09-26 23:46	删除
配置作业中支持英文界面题目	刘强老师	1/55	2013-09-26 19:34	删除
转发: 努力成为优秀的工程师	刘强老师	1/72	2013-09-21 12:09	删除
大家对课程有没有什么要建议的~?	王得希	2/78	2013-09-20 13:02	删除
抢地板	文庆福	0/17	2013-09-16 13:04	删除
抢板凳	巩运青	0/15	2013-09-15 16:39	删除
抢沙发	陈华榕	0/14	2013-09-15 12:51	删除

XSS □□□□□□□□□□□□□□□□ **Script** □□□□□□□□□□□□



系统是否防止 XSS 注入

回复人	网络学堂	回复时间	2013-10-02 20:31	删除	编辑
正文	求别黑...				
附件	无附件				

案例分析：清华大学网络学堂

课程文件	求交互设计课的PPT	文庆福	2/36
教学资源	为什么软件开发周期总是预估的2~3倍?	刘强老师	4/75
课程作业	如何向开源社区提问题	刘强老师	2/70
课程答疑	关于10.9预习材料的一些建议	钱康来	1/126
课程讨论	IBM黑客小组	刘强老师	-821/402
	为什么移动端更适合扁平化而非拟物化设计	文庆福	1/70
	分享：编码、设计和测试规范	刘强老师	1/43

```
</tr>

colspan="5">
able id="table_box" width="100%" cellpadding="0">
<tbody>
▼ <tr class="tr2">
  ▼ <td width="26%">
    <a href="talk_reply_teacher.jsp?
bbs_id=8172986&id=1501901&rep_num=2&up_url=talk_list_teacher.jsp&default_cate_id=1&course_id=103508">求
交互设计课的PPT</a>
  </td>
  <td width="25%">文庆福</td>
  <td width="15%">2/36</td>
  <td width="15%">2013-11-13 17:58</td>
  ▶ <td width="18%">...</td>
</tr>
```

没有对删贴操作
进行权限验证!

http://learn.tsinghua.edu.cn/MultiLanguage/public/bbs/talk_manager_action.jsp?
note_id=1501901&bbs_id=8172986&Commit=%E5%88%A0%E9%99%A4&course_id=103508&default_cate_id=1

--	--	--	--	--



软件安装测试

- 安装过程中对于缺省目录及任意指定目录，是否都能正确安装；
- 若是选择安装，查看能否实现其相应的功能；
- 在所有能中途退出的位置退出安装程序后，验证程序并未安装成功；
- 软件安装后，对其它已经安装的软件是否有影响；
- 裸机安装后，各功能点是否可用；
- 在安装前，安装程序是否判断可用磁盘空间大小，如果不满足安装空间要求，安装程序能否继续；
- 安装时查看版权声明、版本信息、公司名称、LOGO 等是否符合标准；
- 安装过程中界面显示与提示语言是否准确、友好；
- 重复安装时系统是否有提示、是否可以覆盖安装、是否可以升级安装、是否允许多版本共存；
- 是否有注册码或硬件加密狗，在没有它们（或错误）存在的情况下能否顺利安装。

软件卸载测试

- 卸载后注册表中的注册信息及相关的程序安装目录是否能完全删除掉；
- 卸载过程中完全删除共享文件后，看其它程序能否正常运行；
- 卸载后，是否对其它已经安装的软件有影响；
- 系统卸载后用户建立文档是否保留；
- 软件卸载画面上的软件名称及版本信息是否正确；
- 在所有能中途退出卸载的位置是否能正确退出；
- 卸载过程中界面显示与提示语言是否准确、友好；
- 卸载后安装此系统能否打开原来保存的文件，并一切运行正常；
- 卸载程序如果要求重新启动机器，在重启之前是否提示用户保存现有的已运行程序的资料；
- 是否可以选组组件进行卸载；
- 卸载过程中，对意外情况的处理（掉电等）。
- 在卸载过程中，是否有终止或者结束按钮。

运行与关闭测试

- 运行时是否与其它应用程序有冲突（内存冲突）；
- 是否可以同时运行多个程序；
- 任务栏有无程序运行提示；
- 若有未保存的数据，关闭系统时是否有提示；
- 后台服务程序在点击关闭按钮时是否有确认提示；
- 运行时是否过份占用系统资源、退出时能否完成释放占用的系统资源。

案例分析：清华大学网络学堂

用户登录

用户名

密 码

登录总数: 115041833

今日登录: 22569

在线人数: 3254

***** 课 件 *****

教学类

物理 力学

生物 医学

数学 美术

材料 机械

计算机 电子电工

文史艺术 法律法规

经济管理 外语学习

土木建筑

兴趣类

体育 素材库

特别报道 精彩演讲

科学探索 业余爱好

新闻中心网上资料

优秀影片 清华资料

... 管理文件 ...

国家法规 教师手册

2005学生行为准则

27日上午7点至7点半网络学堂服务器调整通知

2013-03-26

舒继武老师《数据结构》第一节课时间地点通知

2013-03-01

欢迎同学们新学期为老师领取爱心水

2012-09-04

倡议书——为老师献上一瓶爱心水

2010-10-26

更多>>

教务通知

再发：《海外名师讲堂》系列讲座...

12-17 16:00

我校学生荣获北京市研究生英语演...

12-17 11:13

English Open Le...

12-17 06:55

请及时参评——本学学期网上“学生...

12-16 16:15

更多>>

本学期课程 [2708门]使用网络学堂

课程名称	任课教师	开放范围	开课系所	浏览次数
经济学原理(...	钟笑寒	本班	经管学院	42309
物理实验A(...	晋青珍	本班	物理系	30734
实验室科研探...	汤彬	本班	训练中心	30146
数据与算法	陈健生	本班	电子系	27622
信息科学技术...	孙家广	本班	信息学院	23258
经济思想史	齐良书	本班	经管学院	21791
数据结构	邓俊辉	全校	计算机系	20410
微积分B(1...	王晓峰	本班	数学系	19990
生物化学(2...	李珍	本班	生命学院	15723
环境保护与可...	钱易	本班	环境学院	15485

更多>>>

所有系 全部课程

课程名:

教 师:

清华图书馆

教学网站

教学门户 教师憩园

教务在线 数学试验

外语自学

计算机文化基础课

计算机应用辅修专业

人文学院电子资源

实验室科研探究

实验室网站

开放实验室

普通物理实验室

现代生命科学实验教学中心

计算机辅助设计教学中心

电工电子学教学实验中心

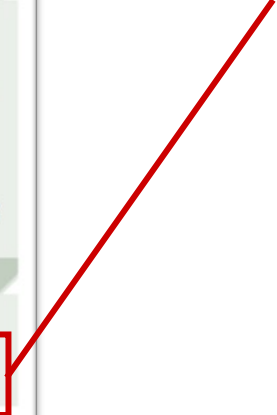
精密仪器光电工程研究所-精密
计量测试教学实验室

校内网站

清华主页 学生清华

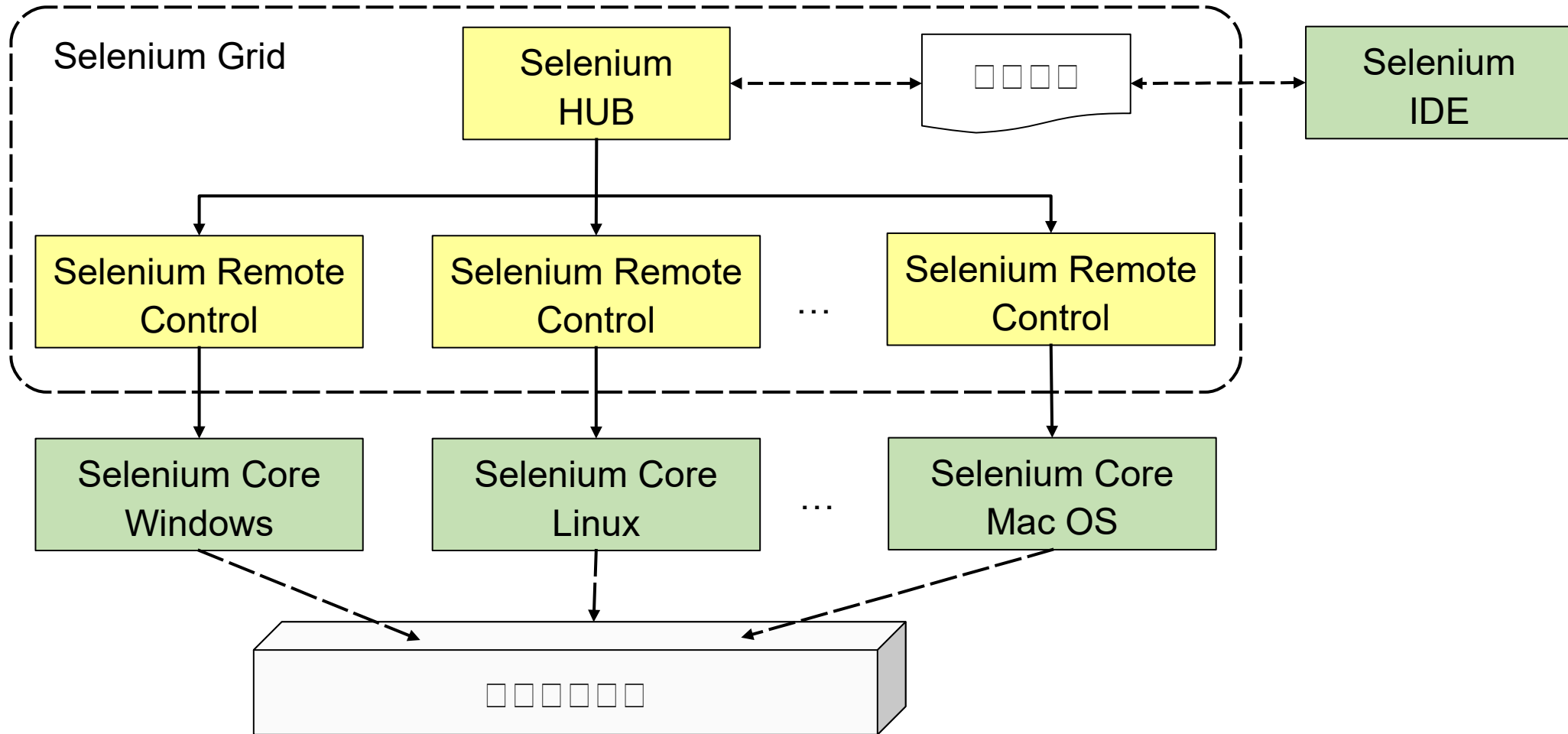
我的家园 藤影荷声

清华大学信息门户 远程教学



- 46

Selenium



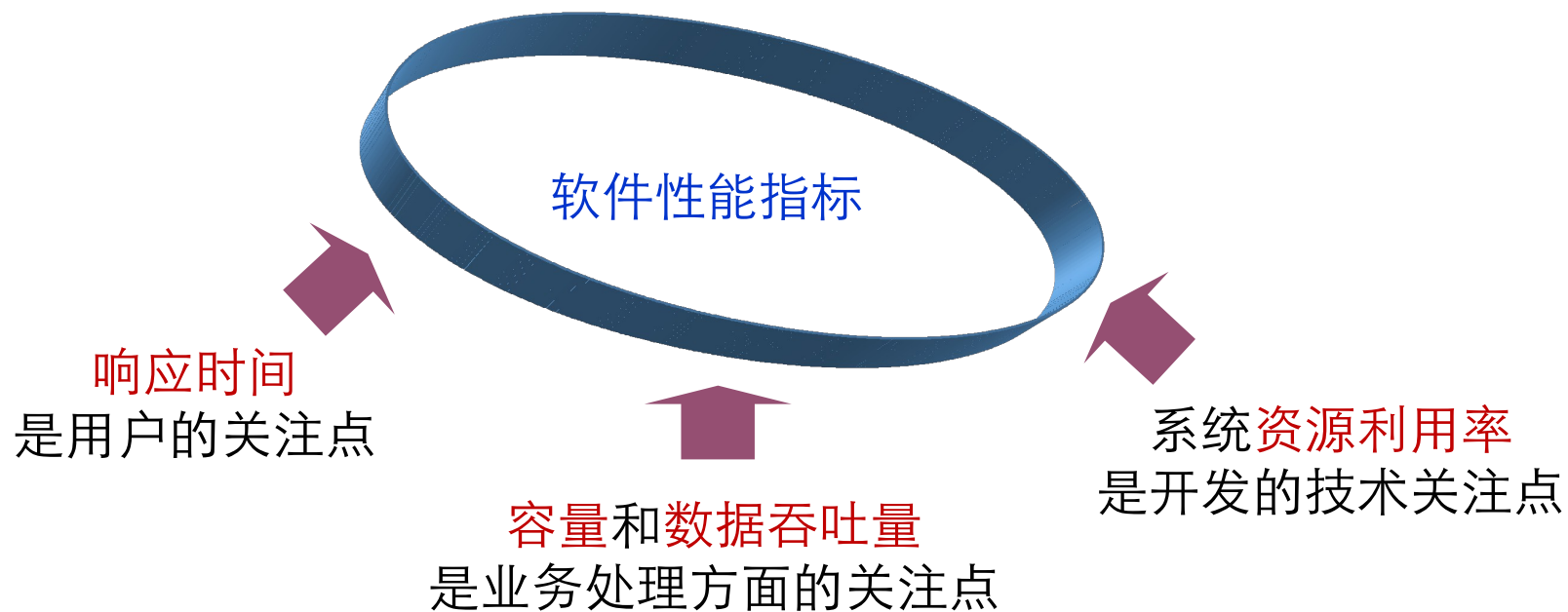


3

-
-
-
-

软件性能

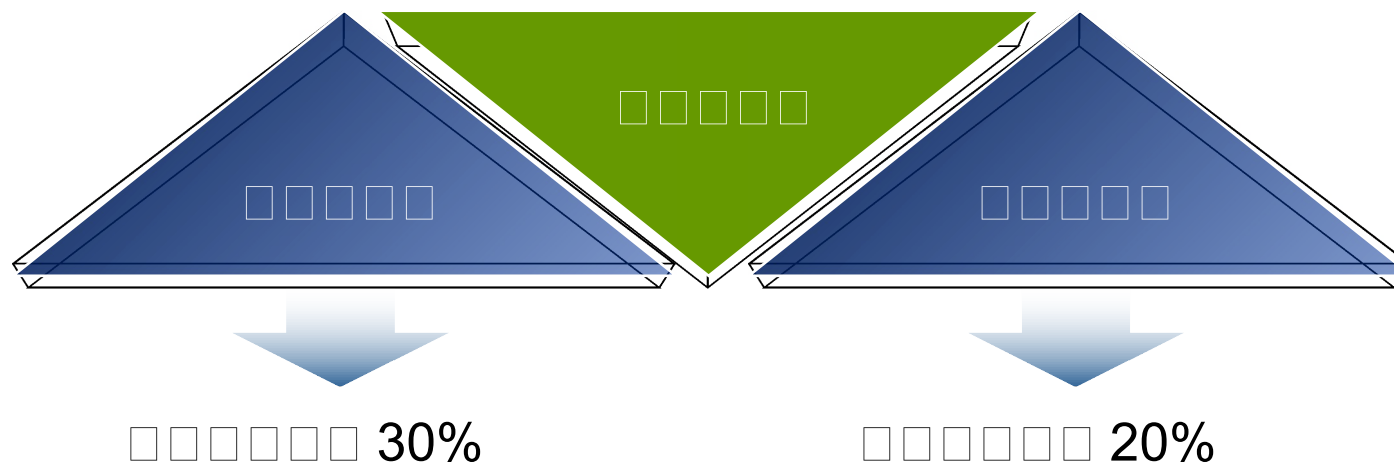
软件性能是软件的一种非功能特性，它关注的不是软件是否能够完成特定功能，而是在完成该功能时展示出来的及时性、资源占用、稳定性、安全性、兼容性、可扩展性、可靠性等。



性能指标

用户数

- 注册用户数：所有在系统注册的用户数目
- 在线用户数：所有正在访问系统的用户（不一定做操作）数目
- 并发用户数：在某一给定时间内某特定时刻进行会话操作的用户数





举例：新浪微博系统

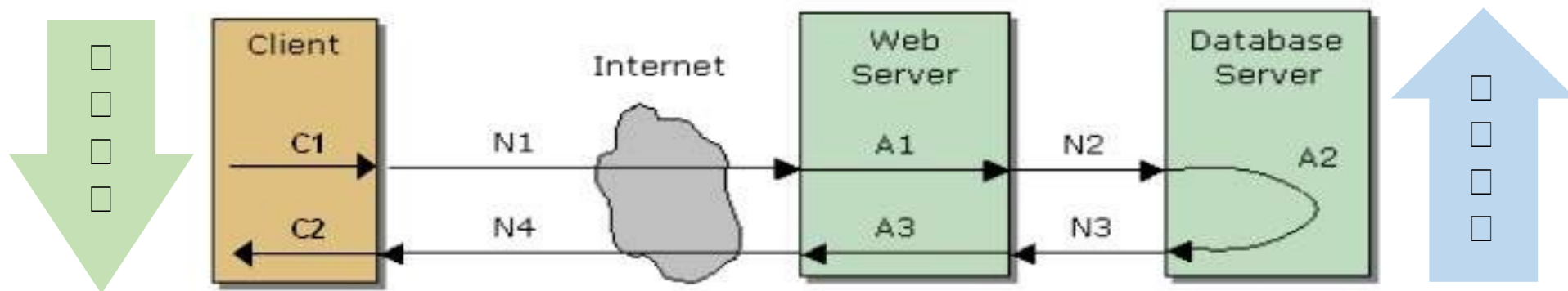
- 假设用户只有注册后才能使用，但注册用户并不是每时每刻都在使用该网站，因此具体一个时刻只有部分注册用户同时在线。
- 在线用户在浏览网站时会花很多时间阅读网站上的信息，因而具体一个时刻只有部分在线用户同时向系统发出请求。
- 由于注册用户可能长时间不登陆网站，使用注册用户数作为性能指标会造成很大的误差。在线用户数和并发用户数都可作为性能指标，但性能测试更多关心并发用户数。

性能指标

响应时间

- 从客户端发出请求到获得响应的整个过程所经历的时间

$$\square\square\square\square\square (A1+A2+A3)+(N1+N2+N3+N4)$$



$$\square\square\square\square\square (C1+C2)+(A1+A2+A3)+(N1+N2+N3+N4)$$

[illegible][illegible]

N1 ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ Web Server ☐ ☐ ☐ ☐ ☐

N3 ☐ DB Server ☐☐☐☐☐☐☐☐☐☐ Web Server ☐☐☐☐☐

N4 Web Server

性能指标



吞吐量

- 吞吐量 Throughput 是指单位时间内系统能够处理的事务数量，通常用每秒事务数（TPS）或每秒请求数（RPS）来衡量。
- 吞吐量的大小与系统的性能、配置、负载等因素密切相关。在相同的系统配置下，负载越高，吞吐量越大；反之，负载越低，吞吐量越小。

资源利用率

- 资源利用率是指系统资源（如CPU、内存、磁盘等）被使用的比例。通常用百分比来表示。例如，CPU 资源利用率为 68%，表示 CPU 正在处理 68% 的任务。
- 资源利用率的高低反映了系统的负载情况。如果资源利用率过高，可能会导致系统性能下降、响应时间增加等问题。因此，需要合理配置系统资源，确保资源利用率处于合理范围内。

性能测试

性能测试是通过自动化测试工具或手段模拟多种正常、峰值以及异常负载条件对系统的各项性能指标进行的一种测试。

负载测试

通过逐渐增加系统负载，测试系统性能的变化，最终确定在满足性能指标的情况下，系统能承受的最大负载量。

目标：在特定的运行条件下验证系统的能力状况。

压力测试

通过逐步增加系统负载，测试系统性能的变化，最终确定在什么负载条件下系统性能处于失效状态，从而获得系统能够提供的最大服务级别的测试。

目标：发现在什么条件下应用程序的性能会变得不可接受。

压力测试类型

- **稳定性压力测试**：高负载下持续运行 24 小时以上。
- **破坏性压力测试**：通过不断加载的手段，快速造成系统的崩溃，让问题尽快地暴露出来。
- **渗入测试**：通过长时间运行，使问题逐渐渗透出来，从而发现内存泄漏、垃圾收集或系统的其他问题，以检验系统的健壮性。
- **峰谷测试**：采用高低突变加载方式进行，先加载到高水平的负载，然后急剧降低负载，稍微平息一段时间，再加载到高水平的负载，重复这样过程，容易发现问题的蛛丝马迹，最终找到问题的根源。

大数据量测试

- ## 疲劳强度测试

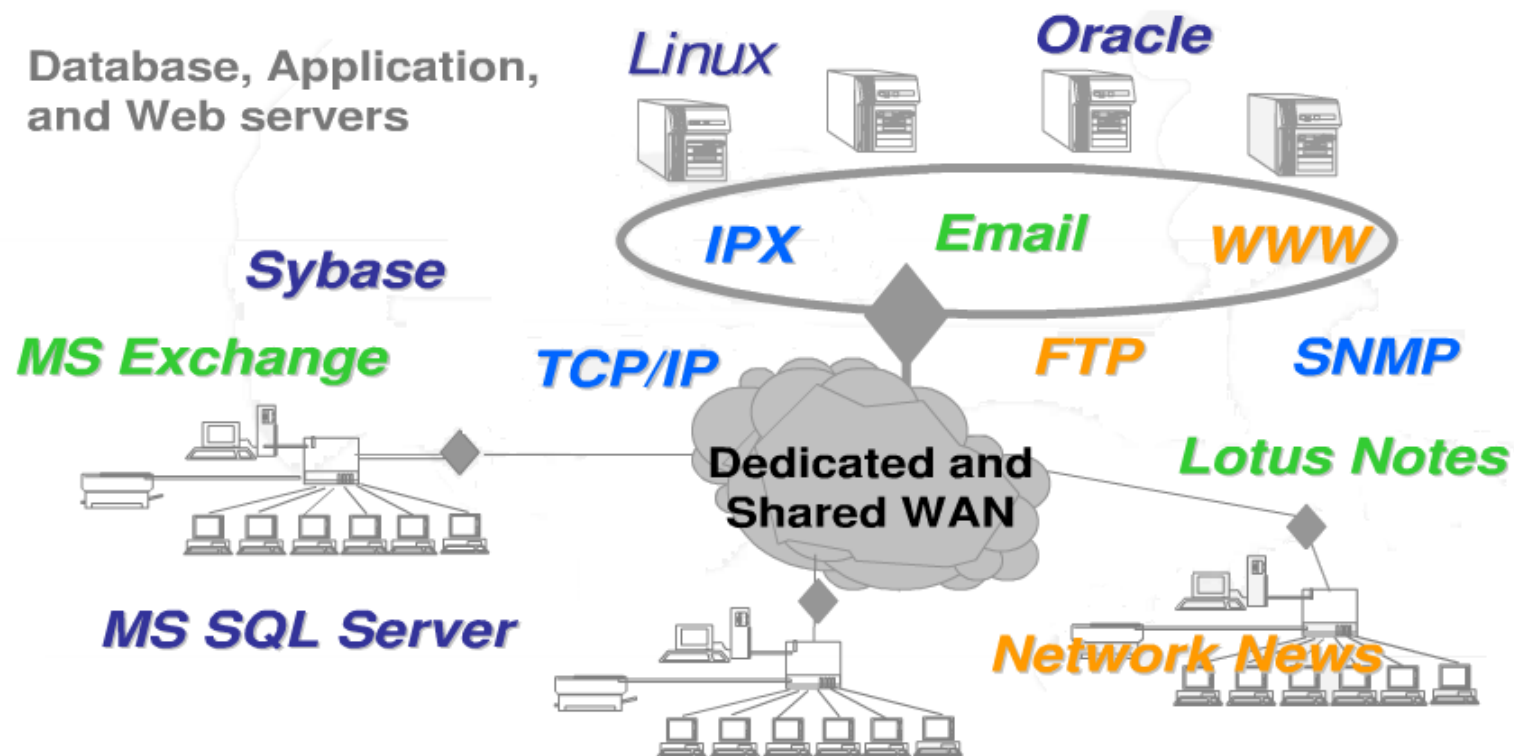
-

失效恢复测试

- ## 基准测试

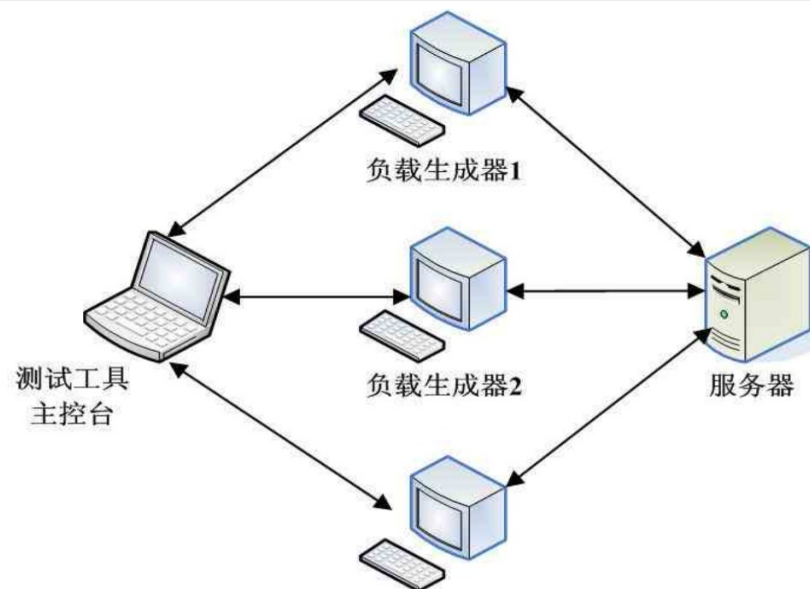
[illegible]

问题：如何找出问题的根源？



性能测试策略

应用在客户端性能的测试：逐渐增加并发虚拟用户数负载，直到系统的瓶颈或者不能接收的性能点，通过综合分析交易执行指标、资源监控指标等来确定系统的并发性能。



□□□□□□□□

□□□□□□□□□□□□□□ PC □□□□□□□□

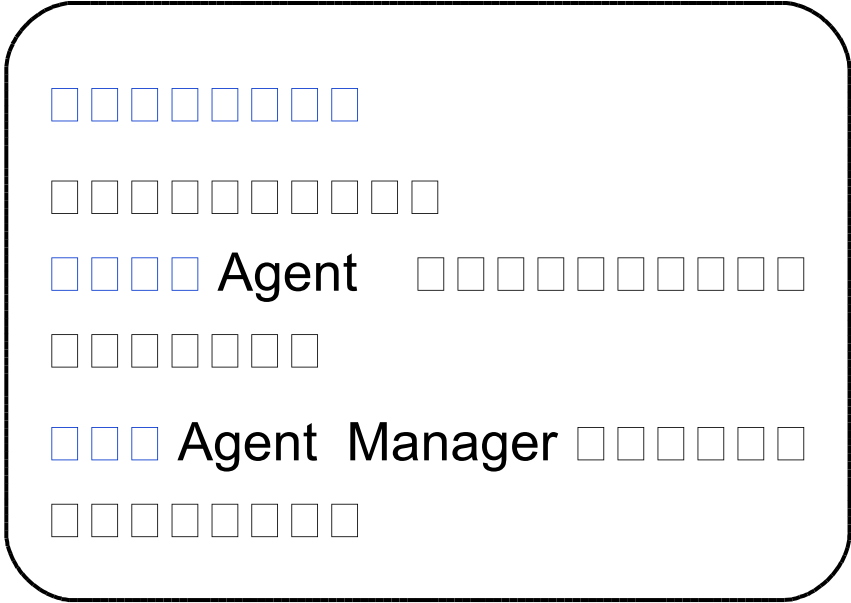
□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□

□□□□□—□□—□□

Diagram illustrating a network packet structure. The packet is divided into three main sections:

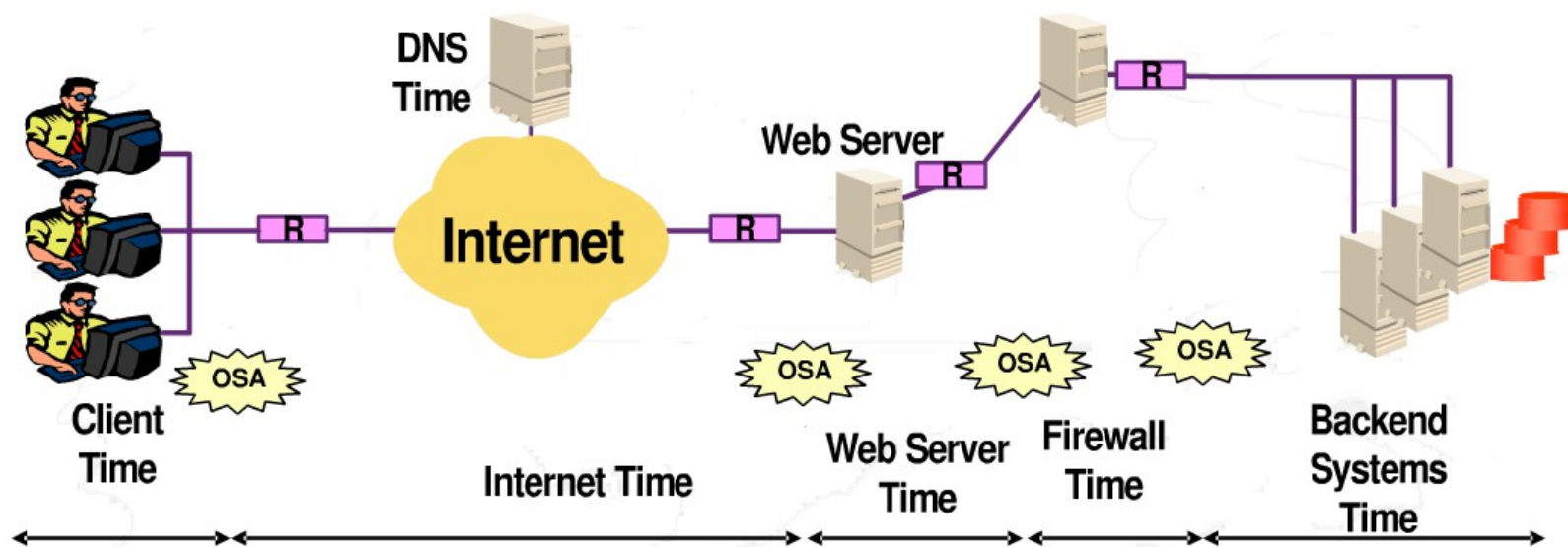
- Header:** Represented by 11 blue-outlined squares.
- Protocol:** Labeled "TCP" in black text.
- Payload:** Represented by 15 black-outlined squares.



性能测试策略

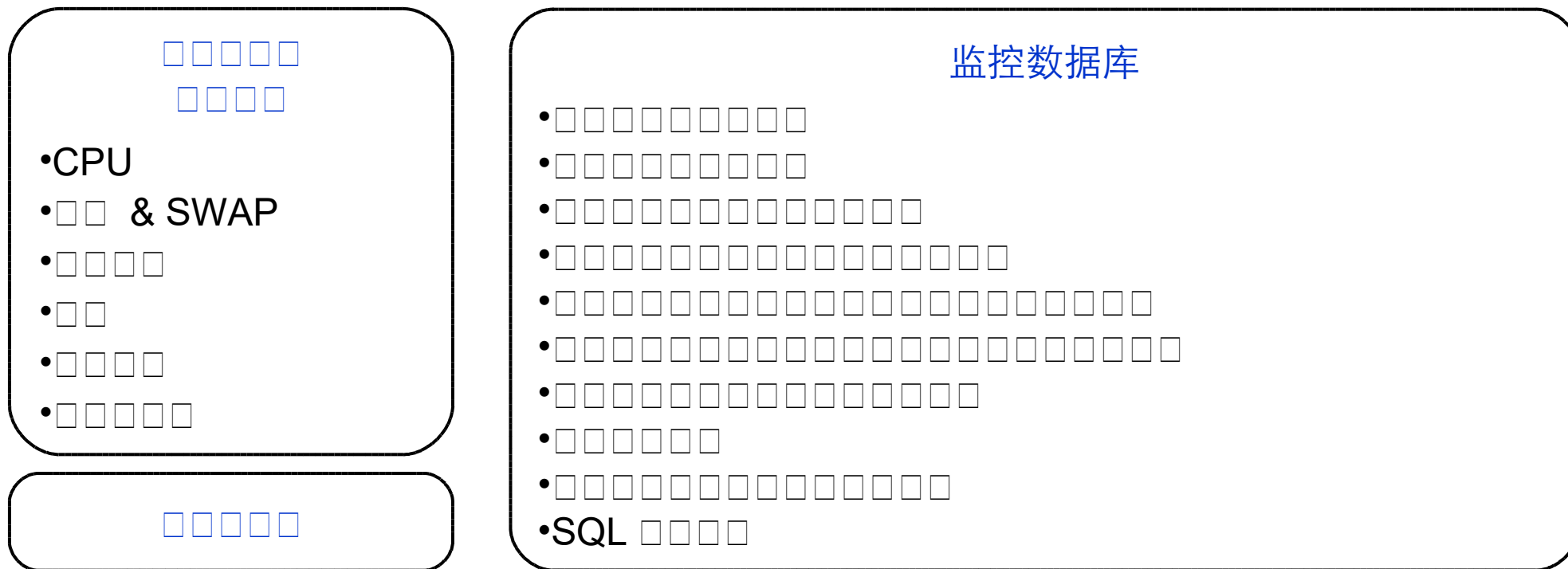
典型的故障诊断策略：设置多点代理服务

- 在应用逻辑路径上多点采集
- 在任何两个节点之间进行数据整合，测量分段响应时间，分析应用故障



性能测试策略

应用在服务器性能的测试：测试关键点是资源占用情况、数据库性能和故障报警。对于应用在服务器上性能的测试，可以采用工具监控，也可以使用系统本身的监控命令。



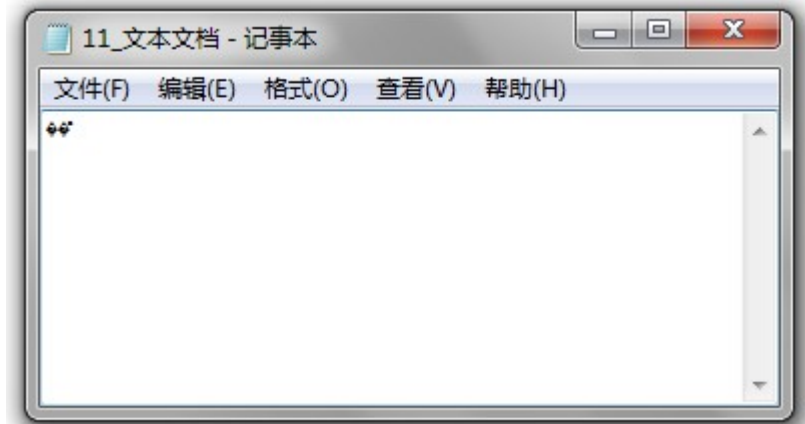
- 64



4

-
-
-
-

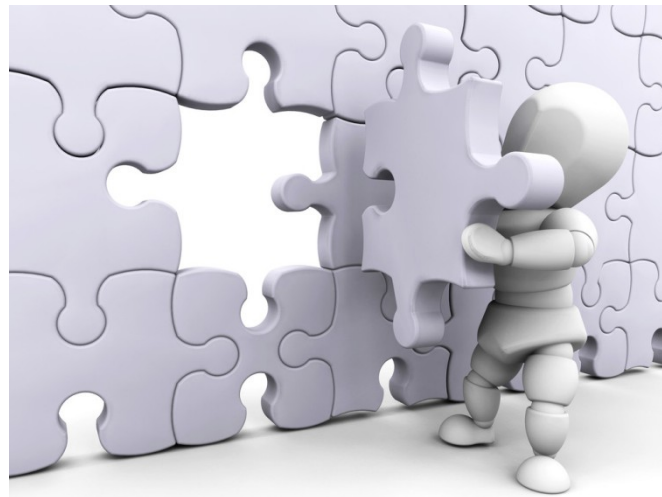
--	--	--	--



严重性与优先级

缺陷的**严重性**是指缺陷对软件产品使用的影响程度，缺陷的**优先级**是指缺陷应该被修复的紧急程度。

- 一般情况：缺陷越严重，越要优先得到修正，缺陷严重等级和缺陷优先级相关性很强。
- 特殊情况：如有些缺陷比较严重，但由于技术的限制或第三方产品的限制，暂时没法修正，其优先级就会低。

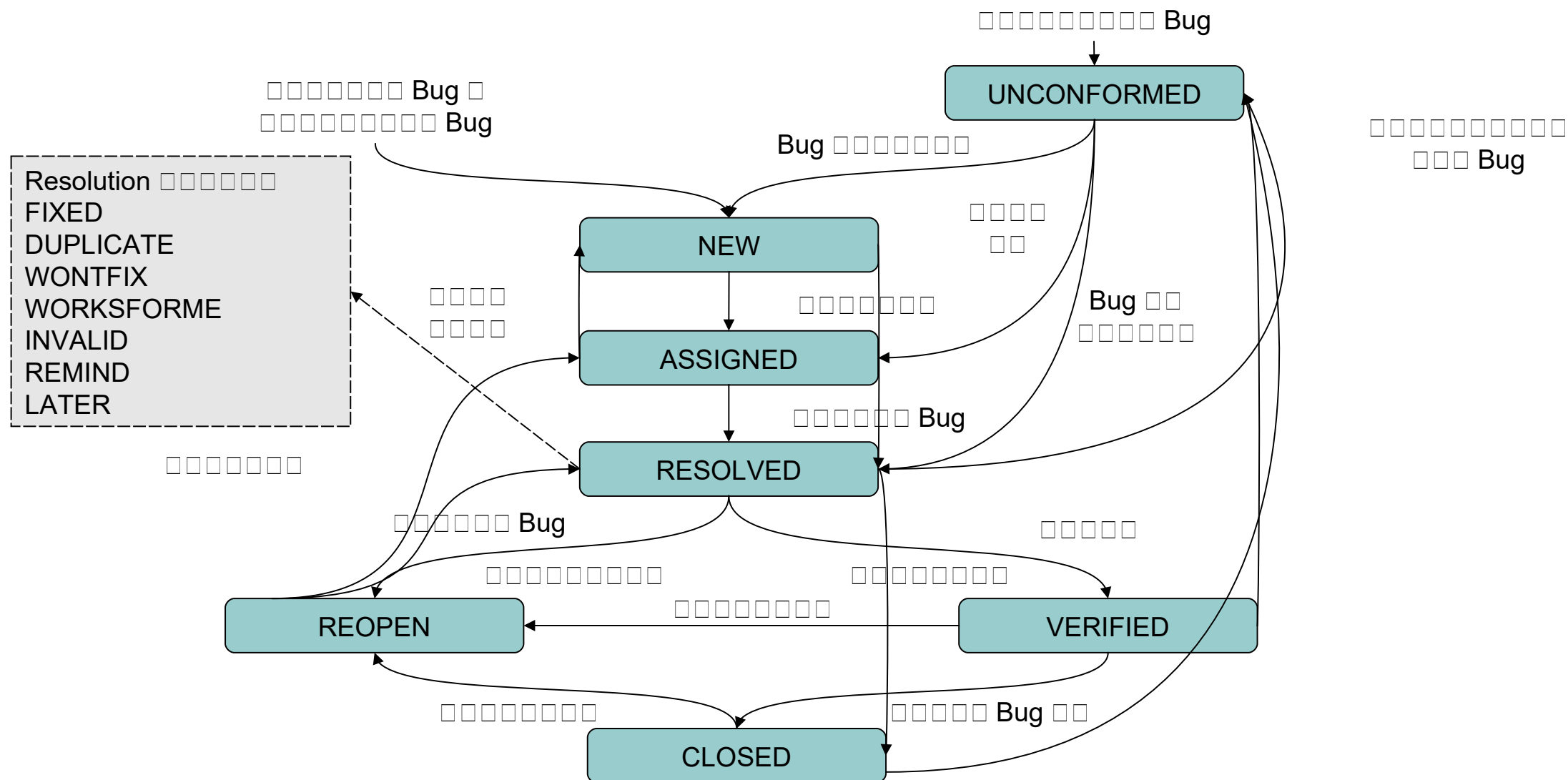


11

[illegible][illegible][illegible][illegible]



软件缺陷生命周期

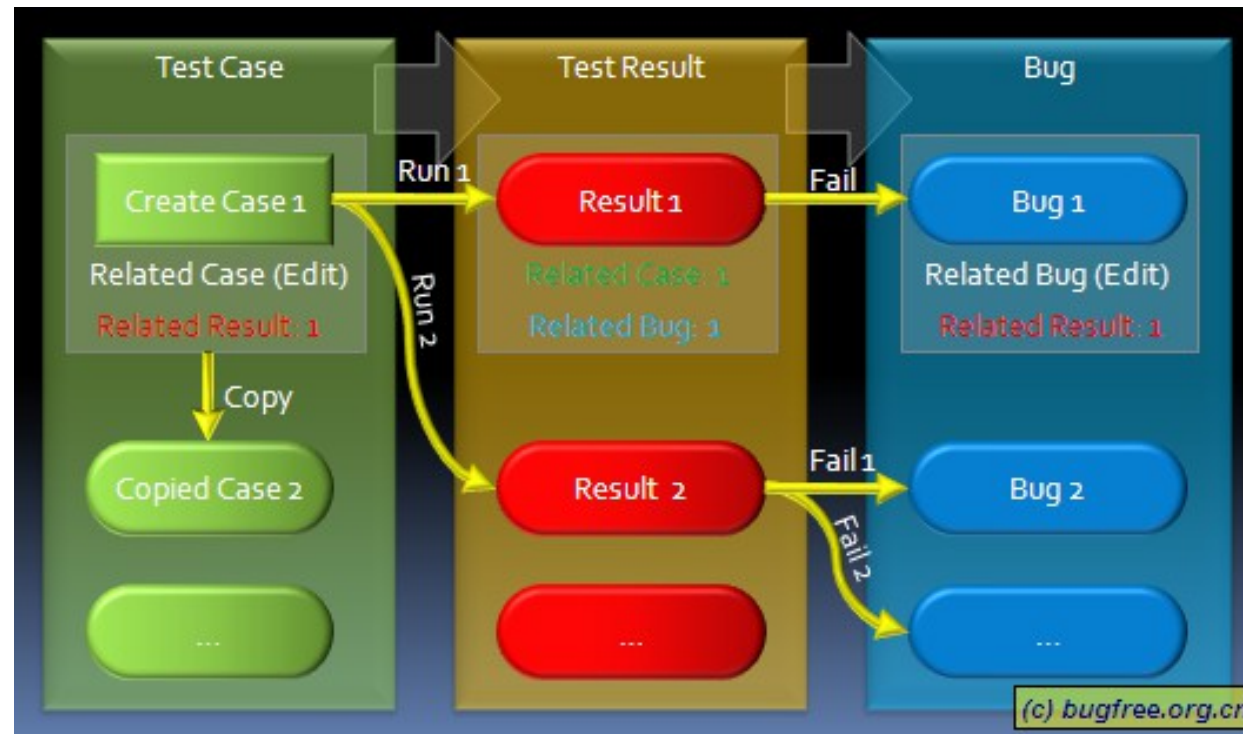


软件缺陷属性

- 缺陷编号：编号是唯一的，可由缺陷跟踪系统自动生成。
- 缺陷标题：简要说明缺陷的类型和内容。
- 缺陷所属：指明缺陷所在模块或所属文档的名称。
- 缺陷严重程度：缺陷对软件产品使用的影响程度。
- 缺陷优先级：缺陷应该被修复的紧急程度。
- 缺陷处理角色：提交人、解决人、确认人
- 缺陷处理状态：
 - ① 确认结果：确认软件缺陷的修正工作是否有效
 - ② 解决结果：预计缺陷修改后能达到的结果
 - ③ 处理结果：缺陷最后的实际处理结果
- 时间信息：提交日期、预计解决日期、确认日期、最终处理日期

缺陷管理系统

BugFree ☐☐☐ PHP ☐ MySQL ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ Linux ☐ Windows ☐☐



Bug □□



Bug □□□□□

□□□□□ Bug □□□□

Active □□□□□□□□□

Bug □□□□□□□□□□□

□□□□□

□□ Bug □□□□□

①By Design □□□□

•Duplicate □□□

•External □□□□

•Fixed □□□

•Not Repro □□□□

•Postponed □□□□

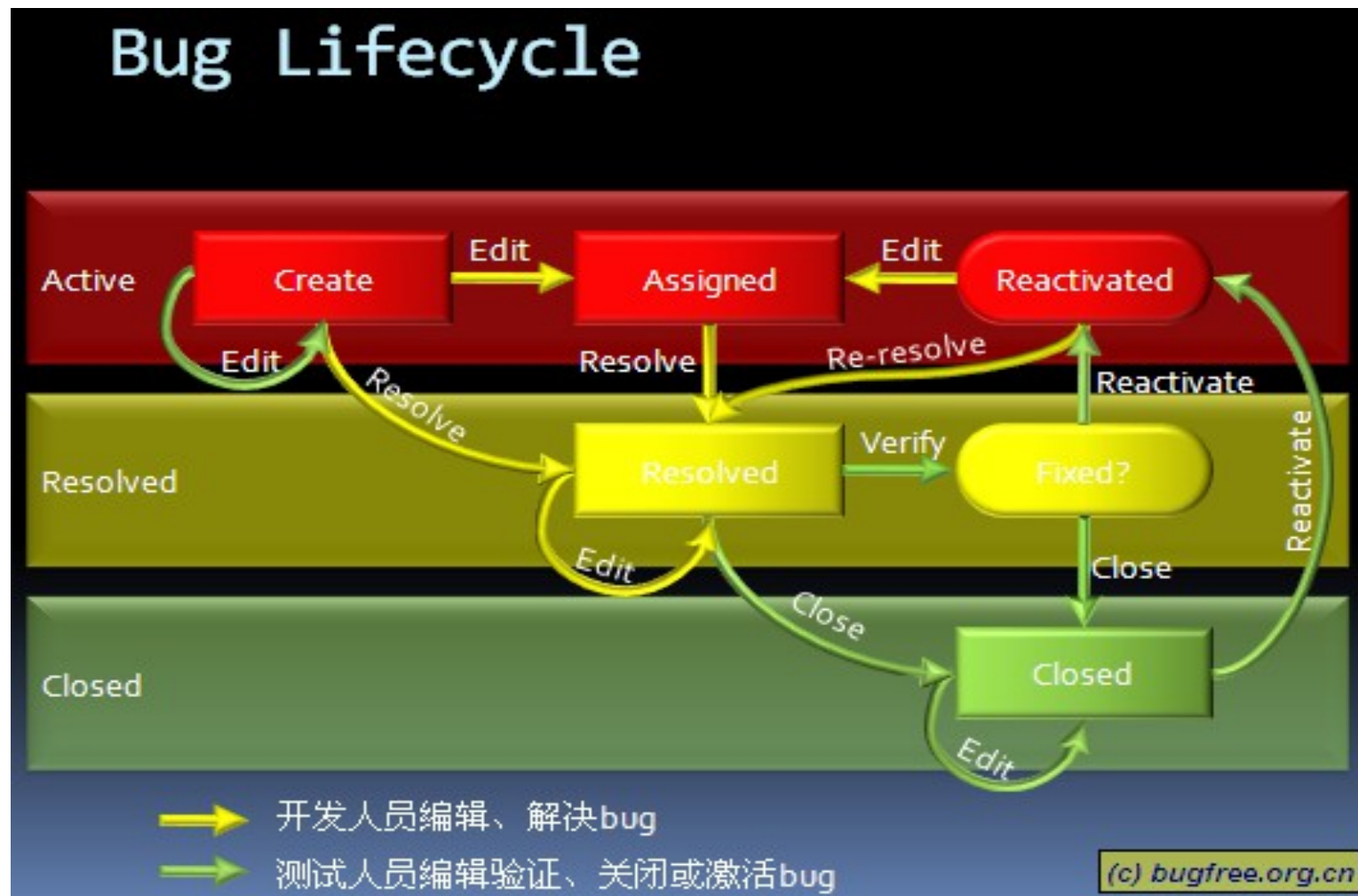
•Won't Fix □□□□□

Bug □□□□□

□□□□□□□□□□□□□□“□□

□”□ Bug □□□□□□□□□□

□ Bug □□□□□□□



□ □ **Bug**

□ □
Bug
□ □ □ □ □ □

保存(S)

另存为模板(T)

Bug标题

模块路径

常规

状态

指派给

抄送给

严重程度

优先级

类型

如何发现

操作系统

浏览器

修改者

修改日期

激活次数

Bug标题：为包含关键词的简单问题摘要，要有利于其他人员进行搜索或通过标题快速了解问题

Sample Product/ /指定问题出现在哪个项目的哪个模块，方便跟踪。如果指定有模块负责人，会自动指派给负责人

新建

创建者系统管理员

创建日期2012-05-26

创建 BuildBug是在哪个版本（Build或Tag）被发现的

解决

解决者

解决日期

解决 BuildBug是在哪个版本（Build或者Tag）被解决的。

预计解决时间

解决方案参考Bug的七种解决方案

重复 Bug

关闭

关闭者

关闭日期

ActiveBug的当前处理人，可以被派给Active，项目或模块负责人再重新分配，或被派给具体人员

需要通知的相关人员，例如测试主管或者开发主管等，可通知多人

1：系统崩溃或者数据丢失的问题；
2：主要功能的问题；
3：次要功能的问题；
4：细微的问题。

2：需要立即解决的问题；
3：需要在指定时间内解决的问题；
4：资源充沛时解决的问题。

其它信息

处理状态Bug处理过程的附属子状态

机器配置测试运行的硬件环境

关键词用于自定义标记，方便查询

Bug相关

相关 Bug

相关 Case

相关 Result

附件(2MB)

浏览

上传Bug的屏幕截图，Log日志或者Call Stack等，方便处理人员。

注释

显示步骤

[illegible]

ee

上一頁(P)

下一頁(N)

編輯(E)

复制(C)

新建Case(S)

解决(S)

关闭(L)

激活(A)

Bug标题

密码列是可视数据

模块路径

示例产品2/模块1

详细

状态

Resolved

指派给

测试二

抄送给

严重程度

1

优先级

1

类型

安全相关

如何发现

功能测试

操作系统

浏览器

修改者

开发二

修改日期

2012-05-28

激活次数

0

新建

创建者

测试二

创建日期

2012-05-28

创建 Build

v 1.0

预计解决时间

2012-06-15

解决

解决者

开发二

解决日期

2012-05-28

解决 Build

v 1.0

解决方案

Fixed

重复 Bug

关闭

关闭者

关闭日期

其它信息

处理状态

机器配置

关键词

Bug 相关

相关 Bug

相关 Case

相关 Result

附件

[bug相关.jpg]

注释

2012-05-28 21:58:00 解决 as Fixed by 开发二

Changed 状态 from "Active" to "Resolved"

Changed 指派给 from "开发二" to "测试二"

Changed 解决方案 from "" to "Fixed"

Changed 解决 Build from "" to "v 1.0"

2012-05-28 21:28:55 编辑 by 经理一

Changed 抄送给 from "经理一, 经理二" to ""

Changed 预计解决时间 from "" to "2012-06-15"

2012-05-28 21:25:25 新建 by 测试二

Added file bug相关.jpg

复现步骤

[步骤]

1. 点击系统管理->用户管理->用户浏览, 打开“用户浏览”界面

[结果]

密码列方可视数据

[期望]

密码列方可视数据

[备注]



谢谢大家！

THANKS

