

软件质量保证与测试

Software Quality Assurance and Testing

第 5 章 集成测试

5.2 集成测试的方法

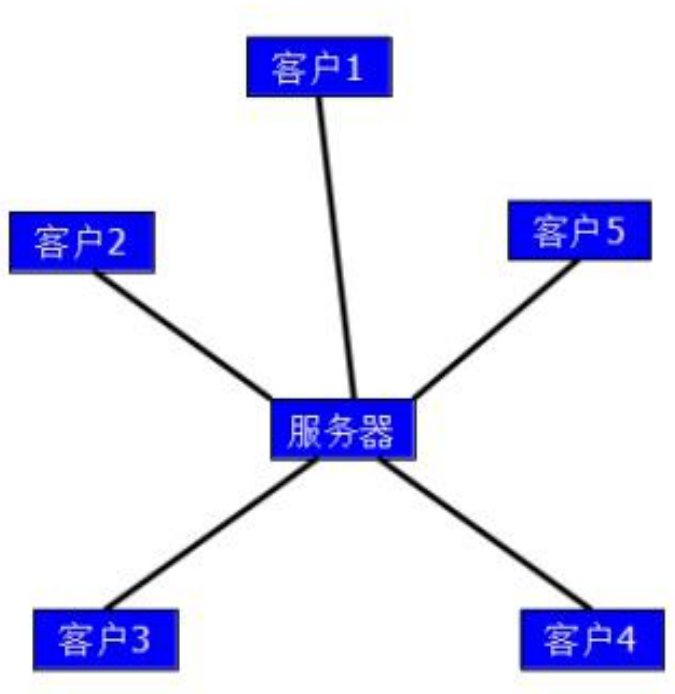


金陵科技学院

客户/服务器集成

1. 客户/服务器结构

随着网络的发展和普及，越来越多的软件为客户/服务器的形式，相应的，客户/服务器集成也应用日益广泛。右图为客户/服务器软件结构图。



客户/服务器集成

2. 客户/服务器集成

客户/服务器软件由于分为客户端和服务端两部分，其集成的过程需要经过三个环节，具体如下：

- ①客户+服务器桩模块集成
- ②服务器+客户桩模块集成
- ③客户+服务器集成

即先分别单独测试客户端和服务端，而用桩模块代替另一端，等到测试通过后，再把客户+服务器组装在一起测试。通过这样三个环节的测试，来检查验证客户和服务端之间交互的正确性、稳定性等。

5.7.2 高频集成测试

1. 高频集成测试简介

高频集成测试是指同步于软件开发过程，每隔一段时间对开发团队的现有代码进行一次集成测试。如某些自动化集成测试工具能实现每日深夜对开发团队的现有代码进行一次集成测试，然后将测试结果发到各开发人员的电子邮箱中。该集成测试方法频繁地将新代码加入到一个已经稳定的基线中，以免集成故障难以发现，同时控制可能出现的基线偏差。

使用高频集成测试需要具备一定的条件: 可以持续获得一个稳定的增量, 并且该增量内部已被验证没有问题; 大部分有意义的功能增加可以在一个相对稳定的时间间隔 (如每个工作日) 内获得; 测试包和代码的开发工作必须是并行进行的, 并且需要版本控制工具来保证始终维护的是测试脚本和代码的最新版本; 必须借助于使用自动化工具来完成。

高频集成一个显著的特点就是集成次数频繁, 显然, 人工的方法是不胜任的。

5.7.2 高频集成测试

2. 高频集成测试步骤

高频集成测试一般采用如下步骤来完成：

步骤一：选择集成测试自动化工具。如很多Java项目采用JUnit+Ant方案来实现集成测试的自动化，也有一些商业集成测试工具可供选择。

步骤二：设置版本控制工具，以确保集成测试自动化工具所获得的版本是最新版本。如使用CVS进行版本控制。

步骤三：测试人员和开发人员负责编写对应程序代码的测试脚本。

5.7.2 高频集成测试

步骤四：设置自动化集成测试工具，每隔一段时间对配置管理库的新添加的代码进行自动化的集成测试，并将测试报告汇报给开发人员和测试人员。

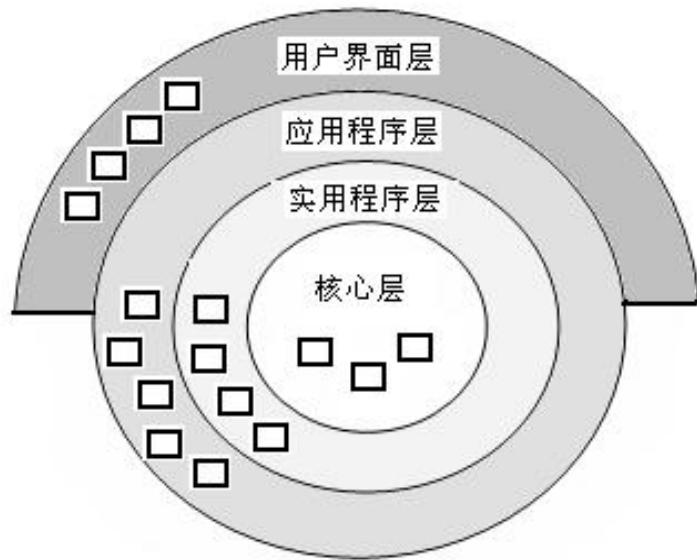
步骤五：测试人员监督代码开发人员及时关闭不合格项。
按照步骤三至步骤五不断循环，直至形成最终软件产品。

5.7.2 高频集成测试

高频集成方案能在开发过程中及时发现代码中的问题和错误，能直观地看到开发团队的有效工程进度。在此方案中，开发维护源代码与开发维护软件测试包被赋予了同等的重要性，这对有效防止错误、及时纠正错误都很有帮助。该方案的缺点在于测试包有时候可能不能暴露深层次的编码错误和图形界面错误等。

5.7.3 分层集成测试

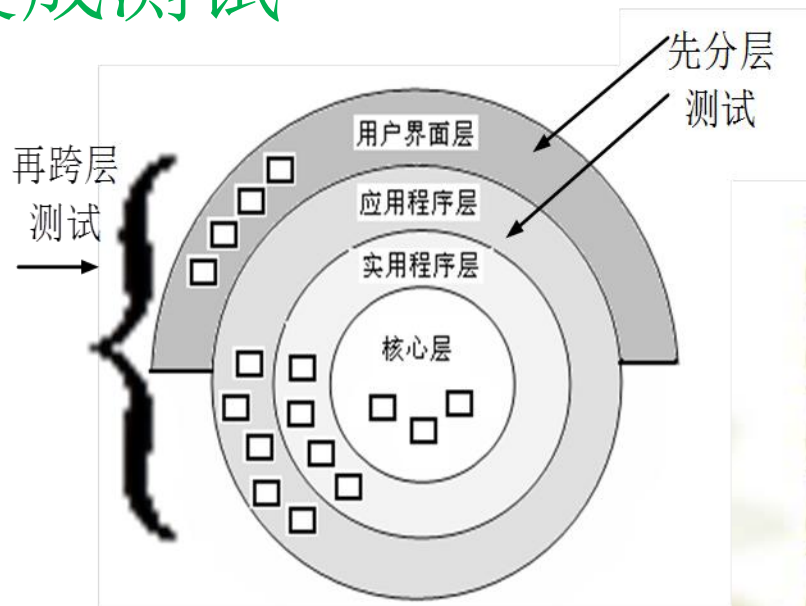
一些软件具有明显的层次，如图下图所示，而分层测试就适用于有明显层次关系的系统。



5.7.3 分层集成测试

分层集成的具体步骤如下：

- ① 划分系统的层次，或者系统设计时就已经有层次；
- ② 确定每个层次内部的集成策略，并分层测试；
- ③ 确定层次间的集成策略，并把多个层次组装起来测试。



分层集成一般只较为适用于具有明显层次结构的软件系统。

5.7.3 基于功能的集成

一个软件可能有很多个功能，但不同的功能其重要性可能不一样，或者进度要求可能不一样，例如某些功能可能急于要投入使用。为优先测试最为重要的功能，最受关注的功能，或者急于要投入使用的功能，可以采用基于功能的集成，具体步骤如下：

- ①确定软件中各个功能的优先级别；
- ②分析优先级最高的功能路径，把该路径上的所有模块集成到一起，必要时使用驱动模块和桩模块；
- ③增加一个关键功能，继续步骤2，直到所有模块都被集成到被测系统中。

采用这一方法的目的是尽早测试和验证系统的关键功能。

5.7.3 基于功能的集成

基于功能的集成最主要的优点就是可以直接验证系统中主要功能，最早地确认所开发的系统中关键功能是否得以实现。缺点是不适用复杂系统，对于部分接口测试不充分，容易漏掉大量接口错误，测试开始的时候需要大量的桩模块，以及容易出现较大的冗余测试。

基于功能的集成适用于：

- ①主要功能具有较大风险性的产品；
- ②探索型技术研发项目；
- ③注重功能实现的项目；
- ④对于所实现的功能信心不强的项目。

5.7.4 基于进度的集成

有的软件项目工期比较紧，需要加快项目的总体进度，为尽可能早地进行集成测试，提高开发与集成的并行性，可以采用基于进度的集成，即把已经开发好的部分尽可能先进行集成，这样就能有效地缩短后续工作所需的工期。

基于进度的集成的优点有：具有比较高的并行度；能够有效加快软件开发进度，缩短软件项目的总体工期。而缺点是：可能最早拿到的模块之间缺乏整体性，只能分头进行集成，导致许多接口必须等到后期才能验证，但此时系统可能已经很复杂，往往无法发现隐藏的接口问题；桩模块和驱动模块的工作量可能会变得很庞大；由于进度的原因，模块可能很不稳定且会不断变动，从而导致测试的重复和浪费。基于进度的集成适用于开发进度优先级高于软件质量的项目。

本节内容就讲到这里，谢谢，再见！



金陵科技学院