Lec14

Smoke test（测试最重要的部分

A quick set of tests run on the daily build.

Cover most important functionalities of the software but NOT exhaustive

Check whether code catches fire or "smoke" (breaks)
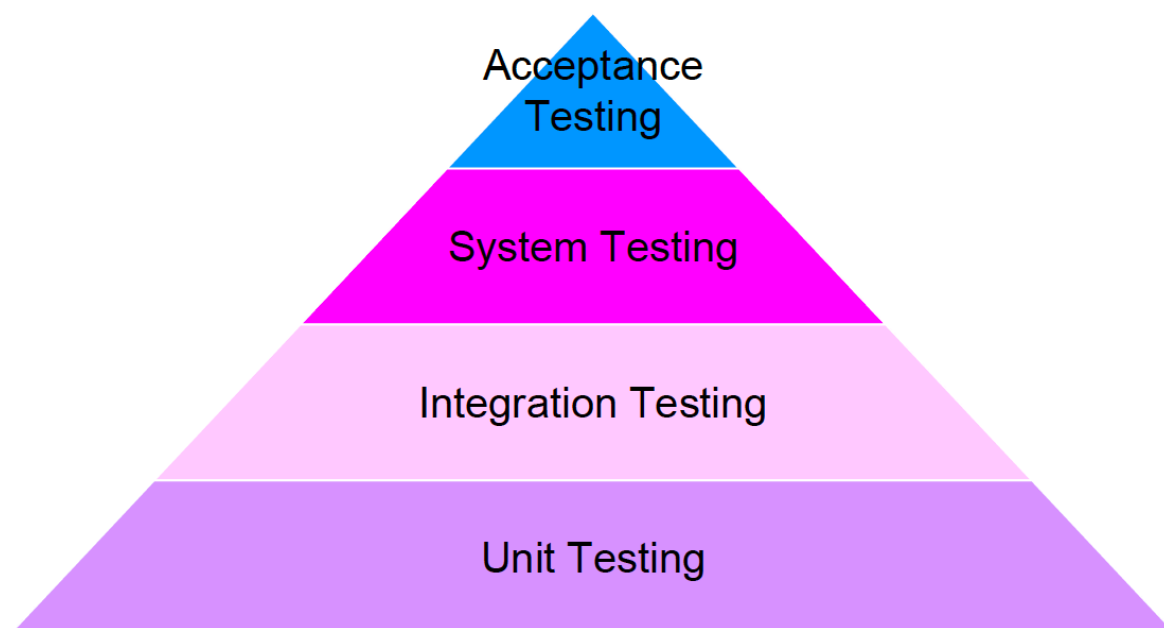
Expose integration problems earlier

CI例子

Jenkins

Bamboo

Travis CI

Levels of Software Testing

Unit testing

Test individual units of a software

Integration testing

Integration testing: Verify software quality by testing two or more dependent software modules as a group.

Big-bang Integration Testing

All component are integrated together at once

pro：Convenient for small systems

con：Finding bugs is difficult; Due to large number of interfaces that need to be tested, some interfaces could be missed easily.

Incremental Integration Testing

Develop a functional "skeleton" system

Design, code, test, debug a small new piece

Integrate this piece with the skeleton

Advantages:

Errors easier to isolate, find, fix

Reduces developer bug-fixing load

System is always in a (relatively) working state

Good for customer relations, developer morale

Disadvantages:

May need to create "stub" versions of some features that have not yet been integrated

Top-down integration/Bottom-up integration

UI/Logic layer

Stub versus Driver

测试的时候用的simulation的模块

Stub: Dummy function gets called by another function

Driver: Dummy function to call another function

Continuous Delivery

"The **essence** of my philosophy to software delivery is to **build software** so that it is always in a state where it could be put into production. We call this **Continuous Delivery** because we are continuously running a deployment pipeline that tests if this software is in a state to be delivered."

Continuous Delivery = CI+ automated test suite

Key：automated testing

Continuous Delivery：人工去deploy

Continuous Deployment：自动deploy

Continuous Deployment

Continous Deployment = CD + Automatic Deployment

Deployment strategies

Strategy 1: Zero-downtime deployment

   \1. Deploy version 1 of your service

   \2. Migrate your database to a new version

   \3. Deploy version 2 of your service in parallel to the version 1

   \4. If version 2 works fine, bring down version 1

   \5. Deployment Complete!

Strategy 2: Blue-green deployment

   \1. Maintain two copies of your production environment ("blue" and "green")

   \2. Route all traffic to the blue environment by mapping production URLs to it

   \3. Deploy and test any changes to the application in the green environment

   \4. "Flip the switch": Map URLs onto green & unmap them from blue.

//What is difference between Continuous Integration and Continuous Delivery ?

//Continuous Delivery requires automated testing before release.
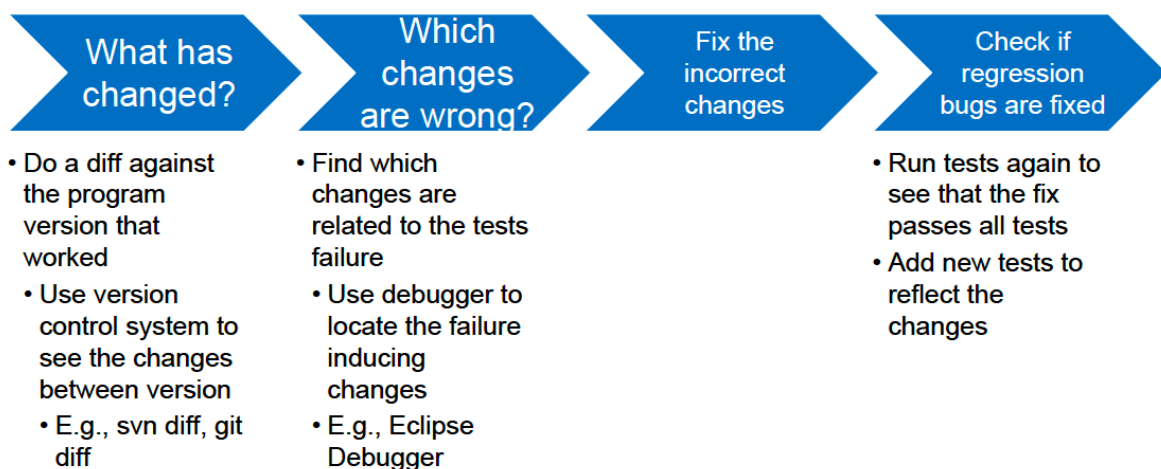
Regression Testing

What is Regression?

Software undergoes changes

But changes can both

improve software, adding feature and fixing bugs

break software, introducing new bugs

We call such "**breaking changes**" regressions

What is Regression testing?

Testing that are performed to ensure that changes made does not break existing functionality（保证change不会破坏现有的功能

It means re-running test cases from existing test suites to ensure that software changes do not introduce new faults（跑现在现有的那些test suites保证不会产生新的错误

fix regression bug

What has changed?	Which changes are wrong?	Fix the incorrect changes	Check if regression bugs are fixed
• Do a diff against the program version that worked • Use version control system to see the changes between version 　• E.g., svn diff, git diff	• Find which changes are related to the tests failure • Use debugger to locate the failure inducing changes • E.g., Eclipse Debugger		• Run tests again to see that the fix passes all tests • Add new tests to reflect the changes

Presubmit Testing

• Uses fine-grained dependencies

• Uses s a m e pool of compute resources

• Avoids breaking the build

• Captures contents of a change and tests in isolation

○ Tests against HEAD

• Integrates with

○ submission tool - submit iff testing is green

○ Code Review Tool - results are posted to the review

Postsubmit testing

● Continuously runs 4.2M tests a s cha ng es are submitted

　○ A test is affected iff a file being changed is present in the transitive closure of the test dependencies. (Regression Test Selection)

　　○ Each test runs in 2 distinct flag combinations (on average)

　　○ Build and run tests concurrently on distributed backend.

　　○ Runs a s often a s capacity allows

● Records the p a s s / fail result for each test in a database

　　○ Each run is uniquely identified by the test + flags + change

　　○ We have 2 years of results for all tests

　　○ And accurate information about what was changed