Software Test Automation Introduction

By: Andy Liu Email: liuygneusoft@hotmailcom

2006.06

MYPM永久免费的国产测试管理软件新秀
www.mypm.cc/test

How This Course Is Organized

Part 1 **Software test foundation**

Part 2 **Software test automation foundation**

Objectives of the Course

1. Mastering test foundation
2. Learn software test methods and test technique
3. Software test management and organization
4. Learn how to test design and analysis
5. Mastering software test automation foundation
6. Learn how to Cost-Benefit analysis
7. Integrated, Effective Test Design and Automation
8. Managing and organising STU process
9. Learn how to apply STU technique to test process

Part 1 :Software Test Foundation

1. Software test summary
2. Software test methods and test technique
3. Managing software test process
4. Test design and analysis
5. Management Practices on Software Testing Procedures

Paer 1:

Software test summary

Why do we need to perform Software Testing?

- **Background**

  With the rapid development of information technologies, the software applications are applied to various industries. As the competitions is becoming more and more drastic, the software quality no doubt becomes the important objective of software development and the primary condition of competitions. For a software product with poor quality, it needs more maintenance cost, and may increase risk. This will affect the benefits of developer. Obviously, the quality becomes first.

Paer 1:
Software test summary

2、Reasons that defects exist

- Communications are insufficient, or there are misunderstandings　Between the communications, or no communication exists at all.
- The software is too complex.
- Perform the wrong programming.
- The requirements varied.
- Be under the pressure of time.
- Be short of code documents.
- Wrong of development tools

Paer 1:
Software test summary

3、 definition of woftware testing

- Software testing covers a series of activities performed in order to discover errors.
- The purpose of testing is to testify that there exist errors in the program, not to testify that there exists no error in the program.
- **Not only to discover the errors, also improve the process of software management.**

Paer 1:
Software test summary

4、Complexity and Economy of Software Testing

- Both Black Box software testing and White Box software testing can not perform thorough testing due to the great quantities of testing cases.

- The general objective of software engineering is to accomplish the high-efficiency and high-quality testing with limited workforce and materials resources. Insufficient testing means to make the customer undertake the hazard resulting from the hidden errors, and over-testing will waste many valuable resources.

Paer 1:
Software test summary

4、Complexity and Economy of Software Testing

- The testing task is so arduous. It is impossible to accomplish the testing. Then what is the purpose of testing? When can we close the testing?

- The final purpose of testing is to ensure the final products delivered to customers meet the demands of customers. Discover more problems as possible and fix them before you deliver the products to customers.

Paer 1:
Software test summary

4、Complexity and Economy of Software Testing

- Considerations on when to close the testing
- It is impossible to stop testing when we find out all the defects.
- **The key is whether it is economic or not.**

  Pressure from the market

  Quality objectives

  Requirements of customers

  Limit of cost

  All testing cases are performed?

  No severe problem exists, or the risk is not severe.
- **In brief, the principle you should follow is: economic and good enough.**

Paer 1:
Software test summary

4、Complexity and Economy of Software Testing

- In brief, the principle you should follow is: economic and good enough.

1. Good-enough is to measure the input/output ratio: Insufficient testing is carefree; over testing will waste the resources, and is also carefree.
2. Various quality assurance activities should be performed in
3. the early of software development procedures.

3. Redo the testing immediately after the fix, to avoid new error.

Paer 1:
Software test summary

4、Complexity and Economy of Software Testing

- **In brief, the principle you should follow is: economic and good enough.**

4. During the course of doc and code modifications, you should prevent new error from the modification (modification check and regression test).

5. Analyze and record each of the testing results to avoid similar error later.

Paer 1:
Software test summary

5、Prejudices needed to be corrected

- **Is software testing equal to program testing?**
- **Can anyone do the test work?**

It is considered that it is difficult to develop a program, but easy to test a program. Actually it is not. Test design is a meticulous job and needs professional skills. Or it will be out of control, and careless omission would happen.

- **Accomplishments that a good testing engineer should possess:**

Good communication skills

Paer 1:
Software test summary

5、Prejudices needed to be corrected

- **Accomplishments that a good testing engineer should possess:**

Empathy ability

Professional skills

Self-confidence

Sense of humor

Better memory

Patience and suspicion

Insight

Paer 1:
Software test summary

5、Prejudices needed to be corrected

- **Accomplishments that a good testing engineer should possess:**

Insight

A good testing engineer should have the point of "test is to destroy", the ability of capturing customer's points, strong pursuit on the quality and the ability of focusing on the details. He can judge the high risk of an application, to put the limited testing on the important things.

- **Is it valuable if a test did not discover any error?**

Paer 1:
Software test summary

6、Conclusion: it is a work with professional skills

- Testing is important, and the testing task is arduous. It is impossible to perform a complete testing.
- There will be error regardless of any technology and method are used. It can decrease the error when new language, advanced development manner, improved development procedures are used, but it can not put an end to the error. The errors needs to be found by testing. To make the testing have more efficiency and pertinency, to meet the demands of customers with least expense, we must find out the rules to be followed.
- What is the first step to make good testing work?

We should find a good method to do any work. So does the testing.

Paer 1:
Methods of Testing and Analysis

Main methods: black box testing for functions and white box testing for internal activities of software

- Black box testing
- White box testing

In addition, it can be broken to static and dynamic testing. Static testing refers to check and review the documents, etc. Dynamic testing refers to run and use the software itself.

What are the problems to be resolved for dynamic testing?

Paer 1:
Methods of Testing and Analysis

What are the problems to be resolved for dynamic testing?

To find out a mature method for the dynamic testing, we must answer the questions below:

- **How to select or generate testing data?**
- **How to organize the testing run of the software?**
- **How to check and record the dynamic running behaviors of the software?**
- **How to judge the validity of the dynamic running behaviors?**
- **When to stop the testing process?**
- **How to analysis the software nature through the testing results?**

Paer 1:
Methods of Testing and Analysis

Which tests are dynamic tests?

Dynamic tests can be classified as below according to the testing purpose:

1、Function test

2、Regression test

Perform a same testing. Check if there exists any problem, if the bug is fixed and if there is any negative affect after the program is changed.

3、Pressure test

4、Performance test

5、Error Acceptance test

6、Stability test

7、Availability test

8、 Globalization test

9、Configuration test, etc.

Paer 1:

Methods of Testing and Analysis

The testing can be classified into 3 categories according to the lifecycle of software:

1、 **Unit test**

2、 **Integration test**

3、 **System test**

What is the difference between unit test, integration test, and system test?

Paer 1:

Methods of Testing and Analysis

Unit test：

The object of unit test is the smallest unit in the software design - module. For this test, you should design test case for all important control paths in a module.

The tasks of unit test include:

1 Test the module ports

2 Test the partial data structure of the module;

3 Test the module boundary condition

4 Test all separate execution paths in a module; 5 Test every error process path in a module

Functions of unit test:

Discover and fix the negative affect resulted from the code change. If such problem is discovered till tester run test and find it, the cost would increase doubly.

Paer 1:
Methods of Testing and Analysis

Integration Test:

Each module can work separately. But they may not work properly after their integration. The main reason is because new problems are introduced into the interface when the modules are called each other

Which method can be used for integration test

System test:

What can be referred to for system test?

Requirements and various specifications of system

The software should be integrated with the other components after its development. At that time, a series of system integration and confirmation tests need to be performed to verify and confirm if the whole system achieve the original objective

What is the preconditions for a successful test?

.

Paer 1:

Organizational Management of Software Testing

(1) Testing configuration management

(2) Procedures in a software testing flow

Testing plan -> testing design -> testing execution -> testing estimation and measurement

(3)Tracking analysis of defects

(4)Various audits in every procedure

Paer 1:

Organizational Management of Software Testing

(1) Testing configuration management

- Background of configuration management

  during the project implementation procedure, the change is unavoidable. The changes make more misunderstanding between the engineers of one project.

- Functions of testing configuration management

  make the evolution visual during whole project processes

  configuration management is the protective activity through the project processes.

.

Paer 1:

Organizational Management of Software Testing

(1) Testing configuration management

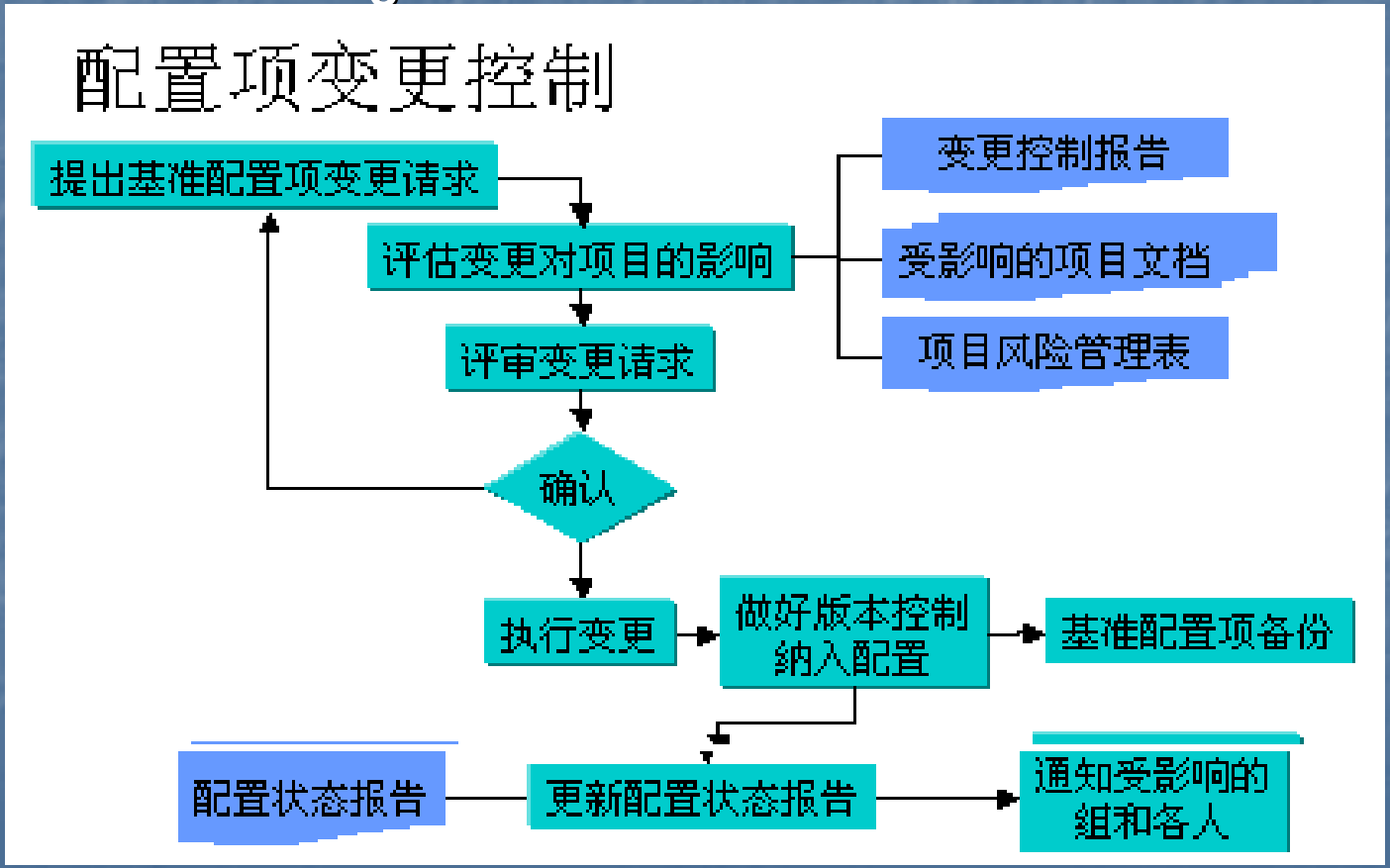
- The organization and management work need to be done for configuration management is listed as below:
- Create a configuration library
- Define the configuration entries
- Audit the configuration
- Control the version
- Control the configuration alteration .

Paer 1:

Organizational Management of Software Testing

(1) Testing configuration management

- Control the configuration alteration



Paer 1:

Organizational Management of Software Testing

(1) Testing Plan

<1>Why do you need a testing plan?

Testing plan is the foundation of high-efficiency testing.

Sufficient preparation can avoid the blindness effectively

shorten the testing circle, improve the testing efficiency, and have an effect of mutual checking between testing documents and development documents.

Testing plan also make a chance of communications with test team, developers and management department.

Testing plan defines the work scope and the persons who will do the work.

Paer 1:

Organizational Management of Software Testing

(1) Testing Plan

<2>The Contents of Testing Plan

Background of testing

Objectives and purpose of testing

Testing strategy

Testing requirements

Preparation for testing environment

Tasks and responsibilities of testing team

Schedule

A good testing strategy and testing method will get twice the testing result with half the effort. It can leverage limited workforce and materials resources to accomplish a high-efficiency and high-quality testing.

Paer 1:

Organizational Management of Software Testing

(1) Testing Plan

<3>General Method of Determining the Testing Strategy

Determine the testing requirements

Testing requirements may have many sources, including test cases, case models, supplementary requirements, design requirements, business cases, discussion with the customers and documents of software architecture, etc.

Estimate the risks and determine the priorities of testing

Paer 1:

Organizational Management of Software Testing

(1) Testing Plan

<3>General Method of Determining the Testing Strategy

A good testing strategy should cover the following contents:

1.Testing type to be implemented and test objectives**.**

2.Phase of implementing the test

3.Testing technologies and tools to be used

4.Estimation and its criterion used to estimate the testing results and to test if the testing has accomplished.

The purposes to give the accomplishment criterion are as below:

Determine the acceptable product quality

Determine the time of successful testing implementation.

Paer 1:

Organizational Management of Software Testing

(1) Testing Plan

<3>General Method of Determining the Testing Strategy

A good testing strategy should cover the following contents:

4.Estimation and its criterion used to estimate the testing results and to test if the testing has accomplished.

The accomplishment criterion should declare the following contents clearly:

The functions, behaviors or conditions to be estimated

Method of estimation.

Criterion, that is, the tally degree with estimation

Special matters that affect the testing operations described in testing strategy

Paer 1:

Organizational Management of Software Testing

(2) Testing Design

<1>The objectives and methods of testing design

Objectives of test cases design

The elementary objective of test case design is to determine a set of testing data, to discover that one error or one kind of errors has high probability.

Test near infinite testing scenarios by using finite test cases.

Methods of test cases design

Logic Coverage

Statement coverage

Decision coverage

Condition coverage

Condition combination coverage

Path coverage

Paer 1:

Organizational Management of Software Testing

(2) Testing Design

Methods of test cases design

Equivalence Partitioning Method

Select minor representative input data to expect more program error discovered with less cost.

- **Boundary Value Method**

Errors occur easily when the program deals with boundary circumstance. Use the boundary value to choose one set of test cases, to check the capability of processing boundary value.

Paer 1:

Organizational Management of Software Testing

(2) Testing Design

Methods of test cases design

Specification Export Method

Design test cases according to the relative criterions. Each test case can be used to test one or more statement in the criterions.

For example, consider the criterion of calculating the square root of a function.

Input: real number

Output: real number

specification:

- **Combination Method of Input and Output**
- **"CRUD" Coverage Method**

Paer 1:

Organizational Management of Software Testing

(3) Testing Execution

Should all test cases be performed during the testing?

Arrange and run testing tasks on the basis of testing objectives.

Two modes are involved during the testing execution procedure.

Manual testing

Automated testing

Use maximum testing resources.

Improve the accuracy of manual testing.

Properly save all history data of testing.

Select the proper test package

How to improve the testing procedure?

Paer 1:

Organizational Management of Software Testing

(4) Estimation and Measurement of Testing

Estimate the completeness of testing

Estimate the code coverage probability

Estimate the defects trend, the defects density, the defects distribution, the workload distribution, the process execution, the design and code quality.

The measurement to the testing procedure is helpful to hold the project conditions, to analyze the procedure data, to discover the advantages and disadvantages easily, to find out the places that need to improve, and to adjust the testing strategy in time. It reduces the testing risks and is the foundation of continual improved software testing procedure.

What are the three common testing models? What are the differences between them?

Paer 1:

Organizational Management of Software Testing

What are the three common testing models? What are the differences between them?

Comparison of three testing procedure models:

V model

W model

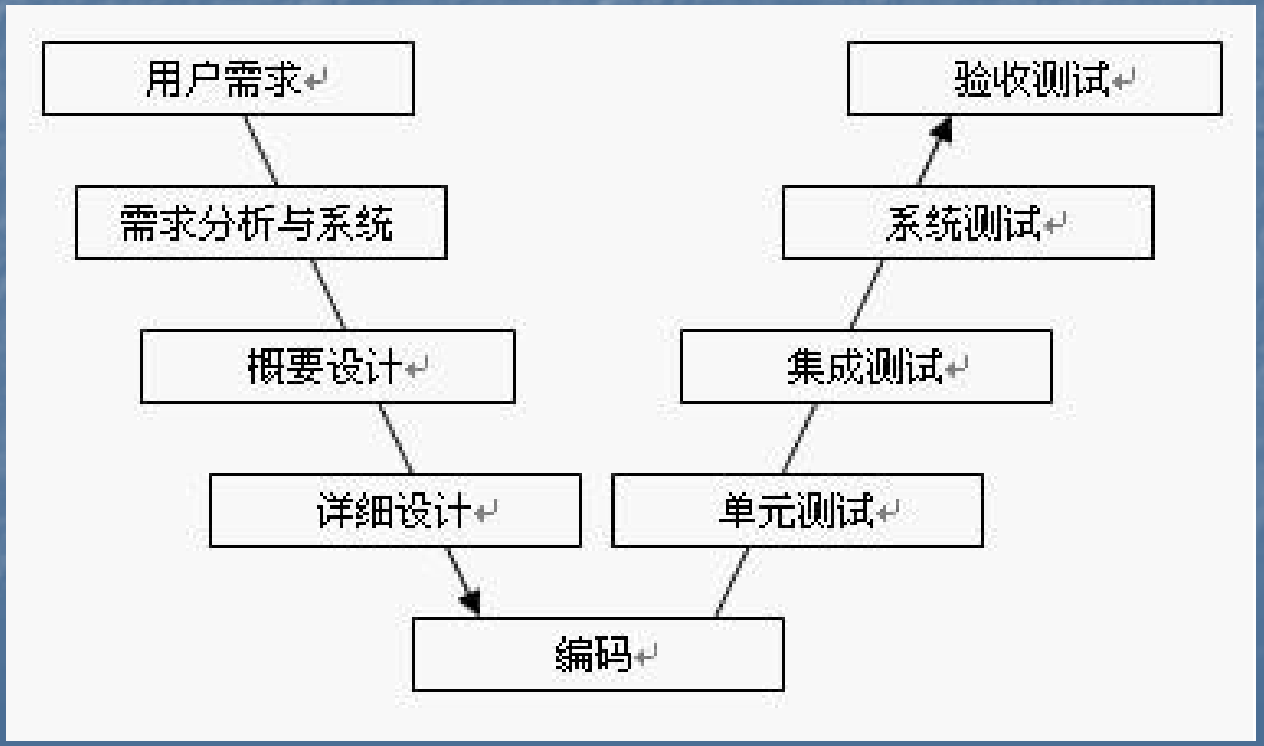
H model

Paer 1:

Organizational Management of Software Testing

What are the three common testing models? What are the differences between them?

V model

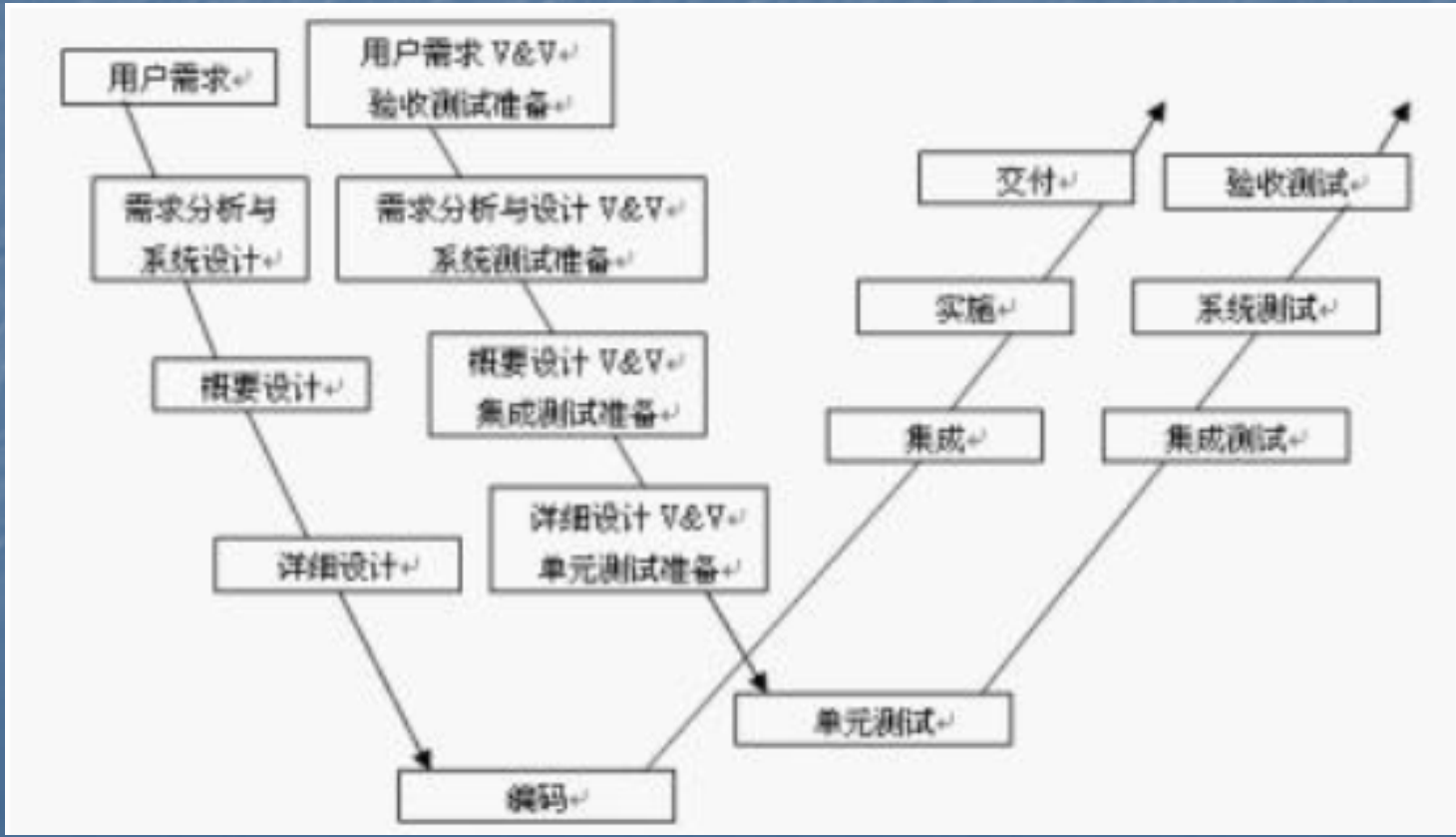


Paer 1:

Organizational Management of Software Testing

What are the three common testing models? What are the differences between them?

W model



Paer 1:

Organizational Management of Software Testing

What are the three common testing models? What are the differences between them?

H model

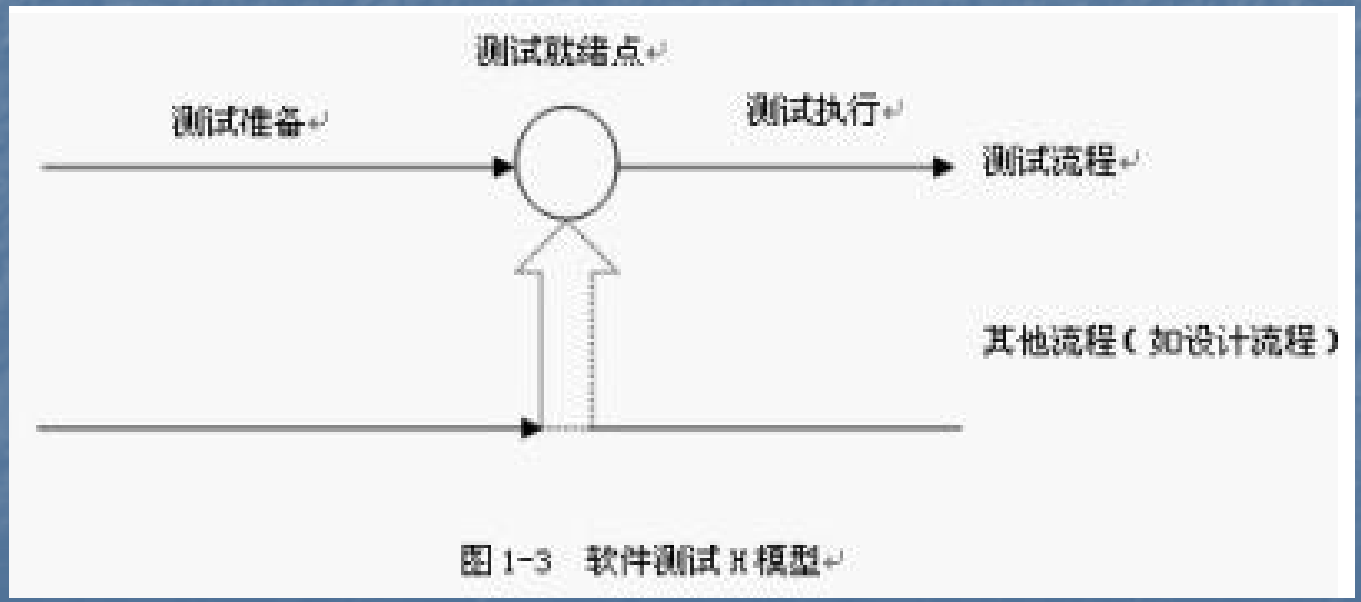


图 1-3　软件测试 H 模型

Part 2 : Software Automated Testing

1、A first look at automated testing

Background

Definition and Purpose

Applicable Scope and Occasion

Manual testing VS. Automated testing

Flow of Automated Implementation

2、Correctly Understand the Automation

3、The Strategy Used When Introducing thAutomated Testing

4、Analyzing technique of automated testing

5、Apply automated test to three test phase

6、General test tool comparing

7、Apply Winrunner , Loadrunner to automated test design and develop

8、casus analysing

Part 2 : Software Automated Testing

1、A first look at automated testing

Why do we introduce the automated testing?

(1)Background

- Today the computer is involved in various aspects of our livings. Many software systems are connected to our lives, wealth and safety directly. With the increase in software scales and the rapid evolution in programming languages, the software test tasks are becoming more complex from complex.
- Just because of this complexity, it is as easy to make mistakes and not feasible in fact for the manual test. Software test tools make the complex test tasks be automated or self-automated. It decreases test overhead and increase the amount of test within the limited time dramatically.

Part 2 : Software Automated Testing

1、A first look at automated testing

Why do we introduce the automated testing?

(2)Definition and Purpose

In the software testing activities, the activity that introducing or developing automated testing tools as a testing mean to perform the testing automatically is the automation of testing. Its purpose is to accomplish the testing with higher efficiency, that is, to decrease workload, save resources, shorten schedule, improve testing quality and improve software quality.

What is the applicable scope and occasion for automated testing?

Part 2 : Software Automated Testing

1、A first look at automated testing

(3)Applicable Scope and Occasion

Both White Box testing and Black Box testing are applicable.

You can make occasion for automated testing when performing the function regression testing, regression testing, performance testing, pressure testing, unit testing, integration testing, and system testing.

Testing execution and the testing output comparison are physical work. They are suitable for testing automation.

Here is just brief introduction. The strategy of automated testing would be discussed later.

What are the advantages of automated testing compared with manual testing?

Part 2 : Software Automated Testing

1、A first look at automated testing

(4) Manual testing VS. Automated testing

Manual Testing	Automated testing
Long duration	Rapid speed
Low reliability	High reliability
Heavy workload	High repeatability
Ease to make mistakes	Reusable
Poor consistency	Programmable

How to implement flow of automated testing?

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<1>The senior management members of the company approve the importance of establishing a testing department. They don't think it is wasting the company's resources.

<2>Before an automated testing is used, there is special person dedicated to it to estimate which tool would be used to get better effect.

<3>Spread the automated testing. Give the trainings on tools usage skills and related knowledge.

<4>Make the analysis on the demands of automated testing.

- The tools can help the tester doing a better testing job. But how we can improve the efficiency by using them? We need think more before the implementation.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<4>Make the analysis on the demands of automated testing.

How to verify the demands of the automated testing?

All agreed on the demands of automated testing.

Decrease the cost by decreasing the workforce.

Perform the test with higher frequency.

Increase the testing coverage.

Ensure the consistency.

Improve the reliability of testing.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<4>Make the analysis on the demands of automated testing.

How to verify the demands of the automated testing?

Allow the person with little professional skills to participate in testing.

Reduce the dependence on the minor acquainted products.

Don't hustle for performing the automated testing at every part. It should be used only if it can gain the best return.

When it is really difficult in some places during the testing, you can consider if it will become easy after the tools are introduces.

The testing demands will make it clear for various measures.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<5>Make the criterion for script maintenance. Add data-driven method in the script to get more branch coverage.

<6>Improve the testing flow

Document your automated testing.

It is required to configure the management support.

Make a detailed description on the testing and audit it.

First write the test cases according to the mode of manual testing. Then improve them from the view of automation technology. Verify that the testing flows are established. Verify that they are simple and cheap to simplify the work.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<7>Durative ability of design

What is test library? What are its functions?

Construct the durative test library. Its benefits include audit-ability, maintainalbility, integrality, independency and repeatability

Maintainalbility:Put the common testing functions into a library file.Independency: Keep the independency without losing the reliability. It is better to run each test separately.

Independency: Keep the independency without losing the reliability. It is better to run each test separately.

Repeatability: It is not allowed to only discover the error intermittently. So, it is required to ensure all tests are same and available during the lifecycle of tested products.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<7>Durative ability of design

What are the difference between testing library and testing framework?

The testing framework is used to separate the testing invoking and testing design.

It is modulized. The components can be reused.

The user interfaces are hidden, that is, you won't face the complex tools. Also, its running does not depend on the user interfaces.

Part 2 : Software Automated Testing

1、A first look at automated testing

(5) Flow of Automated Implementation

<8>Deploy the plan

Pack the testing library and document its usage.

Part 2 : Software Automated Testing

2、Correctly Understand the Automation

It is Return on Investment to introduce the automated testing.

Is automated testing all-purpose?

First, automated testing has limits itself.

It is not suitable for all companies or all projects, such as:

Customized project (once-and-for-all project)

The project whose lifecycle is short.

Object that the business rules are complex.

For the object that the business rules are complex, there are many logic and operation relations. It is hard to test using the tools.

Part 2 : Software Automated Testing

2、Correctly Understand the Automation

It is Return on Investment to introduce the automated testing.

Is automated testing all-purpose?

The tests that focus on the aesthetic feeling, sound, accessbility.

The test is run rarely, such as once per month.

The software is not stable.

The physical interactions are involved.

Second, we must consider the return on investment even if the automated testing is all-purpose.

What are the limits?

It can not replace the manual testing.

Manual testing discovers more defects than automated testing.

The automation of testing can not improve the availability.

Tools themselves do not have imagination.

Testing plan and testing design are not suitable for automated testing for they are brainwork.

Can't generate the test cases based on the code for GUI application.

Part 2 : Software Automated Testing

2、Correctly Understand the Automation

It is Return on Investment to introduce the automated testing.

Is automated testing all-purpose?

Second, we must consider the return on investment even if the automated testing is all-purpose.

What are the limits?

Automatically generate the test cases based on the code only for class or function, not for system testing level.

How to master the return on investment of automated testing?

Part 2 : Software Automated Testing

2、Correctly Understand the Automation

It is Return on Investment to introduce the automated testing.

How to master the return on investment of automated testing?

The more you invest, the more you expect. But the automation of testing can not get effect instantly, because it takes time to train the personnel, to be familiar with the tools and testing flow.

Possibly, you need to spend money on automatic tool, because no testing tool can fit all tests.

The problems from tools also bring some risks.

Automated testing often improves the tester's professional skills that they need.

There are other problems, such as technology problems, organization problems and script maintenance.

Part 2 : Software Automated Testing

2、Correctly Understand the Automation

It is Return on Investment to introduce the automated testing.

How to master the return on investment of automated testing?

You have to consider the progress, investment, new risks, applicable scope, required resources, and the costs to achieve the returns when you consider the automated testing.

Since both the advantages and limits exist, you need to study the strategy when you introduce the automated testing. Then what kind of strategy can you use?

Part 2 : Software Automated Testing

3、The Strategy Used When Introducing the Automated Testing

(1)Identify the demands of automated testing

<1>Understand the costs of automated testing

<2> Do not use unstable program to do the automated testing.

<3> Check if the user interface is stable.

<4> Find out it is how many versions are tested that the product can be stable.

<5> Find out if the product is translated into other languages, if the re-compiling need to be done after the translation, and how many versions of localization need to be done.

<6> A similar testing method can be used to several products of company.

Part 2 : Software Automated Testing

3、The Strategy Used When Introducing the Automated Testing

(1)Identify the demands of automated testing

<7>The product quality is driven by market or the internal improvement from company.

<8>Find out if the non-programmer are demanded to create test cases for automated testing

(2) Selection of Tools

(3) Risks estimation

(4) Automated testing flow

(5) Requirements on the automated testing skills

(6) Automated testing strategies in the different stages of software lifecycle

Part 2 : Software Automated Testing

4、Analysing technique of automated testing

(1)Regression test

purpose

advantages and limits

strategies

(2)Date-driven test

The Strategy Used When determining data set for data-driven test

Analysis application it to test

(3) Automated nightly builds

objectives

implement

Part 2 : Software Automated Testing

4、Analysing technique of automated testing

(4) rganizational Management of automated software Test

configuration

develop

execution

analysis report

(5)TSL maintenance and debug

(6) Automated test library and automated test framework

function,

principle of design

maintenance

changing managing

Part 2 : Software Automated Testing

5、Apply automated test to three test phase

(1) Unit test

Task of unit test

Test tool and methods for unit test

unit test evaluation

(5) Integration test

Two test model for integration test

Task of integration test

Test methods for integration test

Integration test evaluation

(6) System test

Identifying requirement of system test

Introduction automaed test into system test

Application and analysis of automation test for system test

performance

Load

Function

Part 2 ：Software Automated Testing

6、 Contrast ot tool used commonly

Winrunner VS Robot

Loadrunner VSWebload

Junit VS EMMA

7、 Apply Winrunner , Loadrunner to automated test design and develop Process

Automated design of function test and performance test

TSL programme

Results analysis

8、analysing casus

MYPM永久免费的国产测试管理软件新秀 www.mypm.cc/test

The end thanks !

Author :Andy Liu mail: liuygNeusoft@hotmail.com

2006.06

Any question contact me by mail or MSN