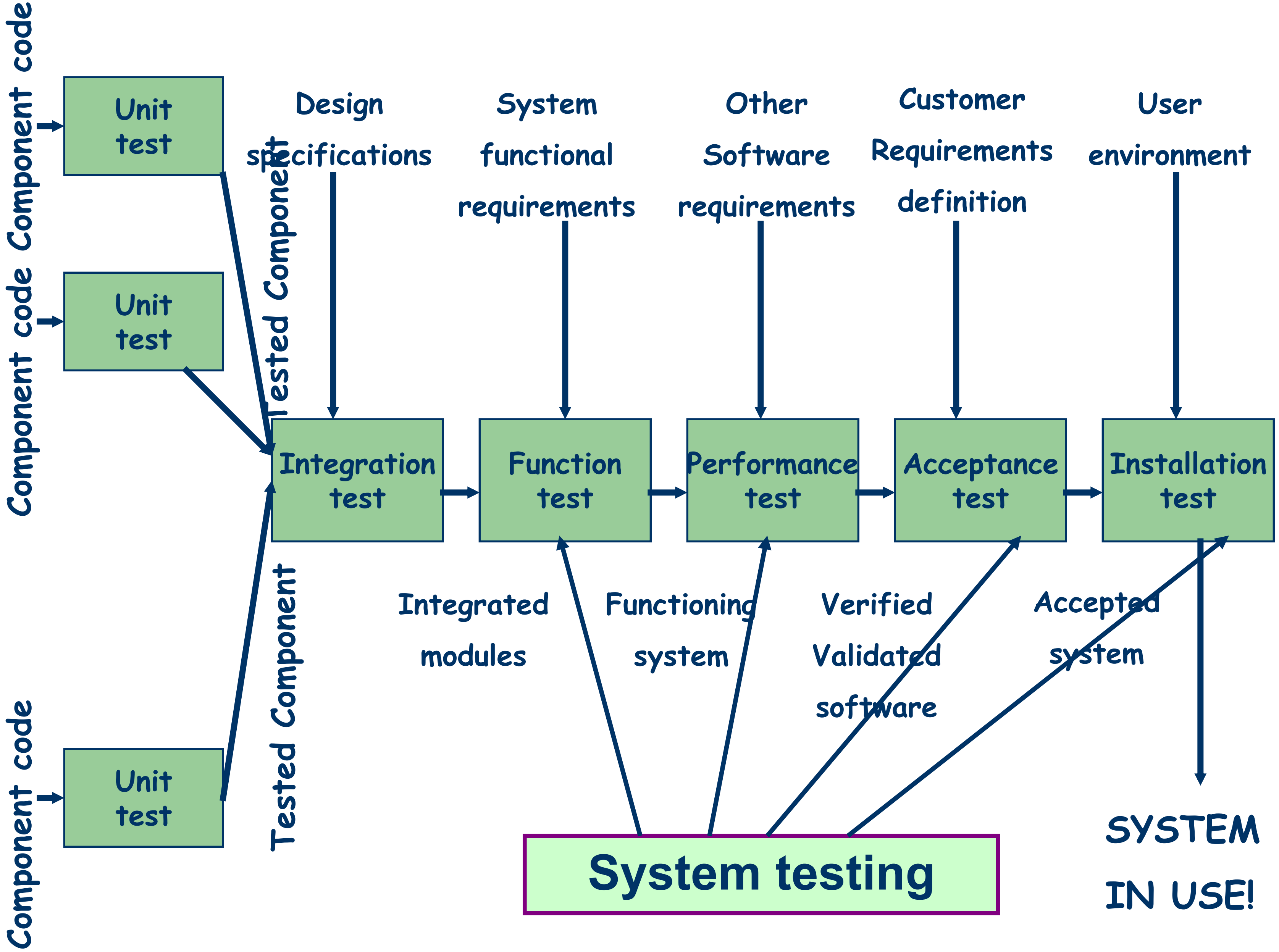
Chapter 9 Testing the System

Note A: unit and integration testing----by yourself
or a small part of the development
team（you can complete control
over the testing process.）

B: system testing----by the entire development
team (you can’t control the testing
process，you must work with the
entire development team.）

-

[①②③④⑤⑥]



Component code

Component code

Component code

Unit test

Unit test

Unit test

Tested Component

Tested Component

Design specifications

System functional requirements

Other Software requirements

Customer Requirements definition

User environment

Integration test

Function test

Performance test

Acceptance test

Installation test

Integrated modules

Functioning system

Verified Validated software

Accepted system

System testing

SYSTEM IN USE!

Fig 8.3 Testing steps.

Chapter 9 Testing the System

- **9.1 Principles (原理) of System Testing**

Focus A: The objective of unit and integration

-------ensure the code implemented

the design properly

(程序员/测试者满足设计师的设计要求)

B: The objective of system testing

-------ensure the customer wants it to do

(开发团队确保系统满足客户的需求)

软件的最终要求

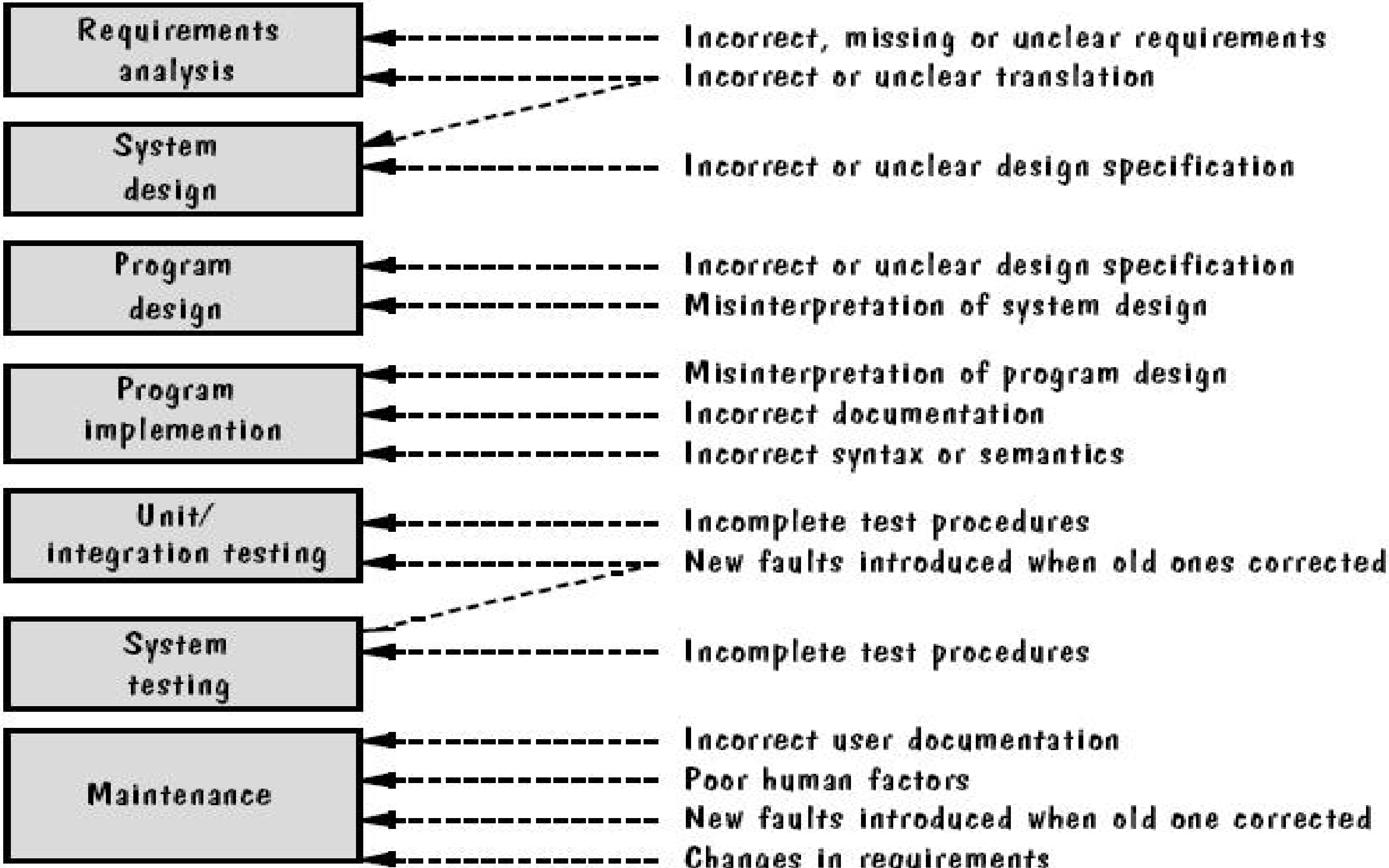
Chapter 9 Testing the System

- 1. Sources of Software Faults (软件缺陷的来源)
  - ①The objective(目标)of testing : eliminate all faults that might lead to failures
  - ②appearance of faults : can be inserted in any phase during software process (Fig9.1)

    A: faults in the requirement (uncertain, incorrect, missing, misunderstand， etc.)

    B: faults in the system design(incorrect or unclear design, incorrect translation from SRS) （有时低效的设计进一步维护时产生紊乱，新的缺陷难以检测）

Chapter 9 Testing the System



Requirements analysis
System design
Program design
Program implemention
Unit/ integration testing
System testing
Maintenance
Incorrect, missing or unclear requirements
Incorrect or unclear translation
Incorrect or unclear design specification
Incorrect or unclear design specification
Misinterpretation of system design
Misinterpretation of program design
Incorrect documentation
Incorrect syntax or semantics
Incomplete test procedures
New faults introduced when old ones corrected
Incomplete test procedures
Incorrect user documentation
Poor human factors
New faults introduced when old one corrected
Changes in requirements

Chapter 9 Testing the System

C: faults in the program design

(incorrect or unclear program design, misinterpretation of system design)

D: faults in coding

(incorrect documentation, incorrect semantics, misinterpretation of program design)

E: faults in unit/integration testing

----incomplete test process

----correct old mistakes ⟶ generate new errors (correct 3or4 old bugs, then generate 1 new bug)-----Microsoft

----(example P419-s4)

Chapter 9 Testing the System

F: faults in system testing

(incomplete test procedure)

G: faults in maintenance (change requirement, incorrect user document, poor human factors, new faults introduced when correct old errors)

- ③ demanding (要求) to testing process

A: thoroughness

B: satisfaction (by user, customer, developer)

(因为在大型系统中，肯定存在用户允许的、当时不必修复的缺陷存在)

Chapter 9 Testing the System

- 2. System Testing Process (Fig9.2)
  - ① Process Objectives（处理过程的各个目标）

    A: function test ------check <SRS> by developer

    B: performance test -----check <SRS>by developer

    C: acceptance test ---check <requirement definition> by developer and user

    D: installation test -----test in user's environment

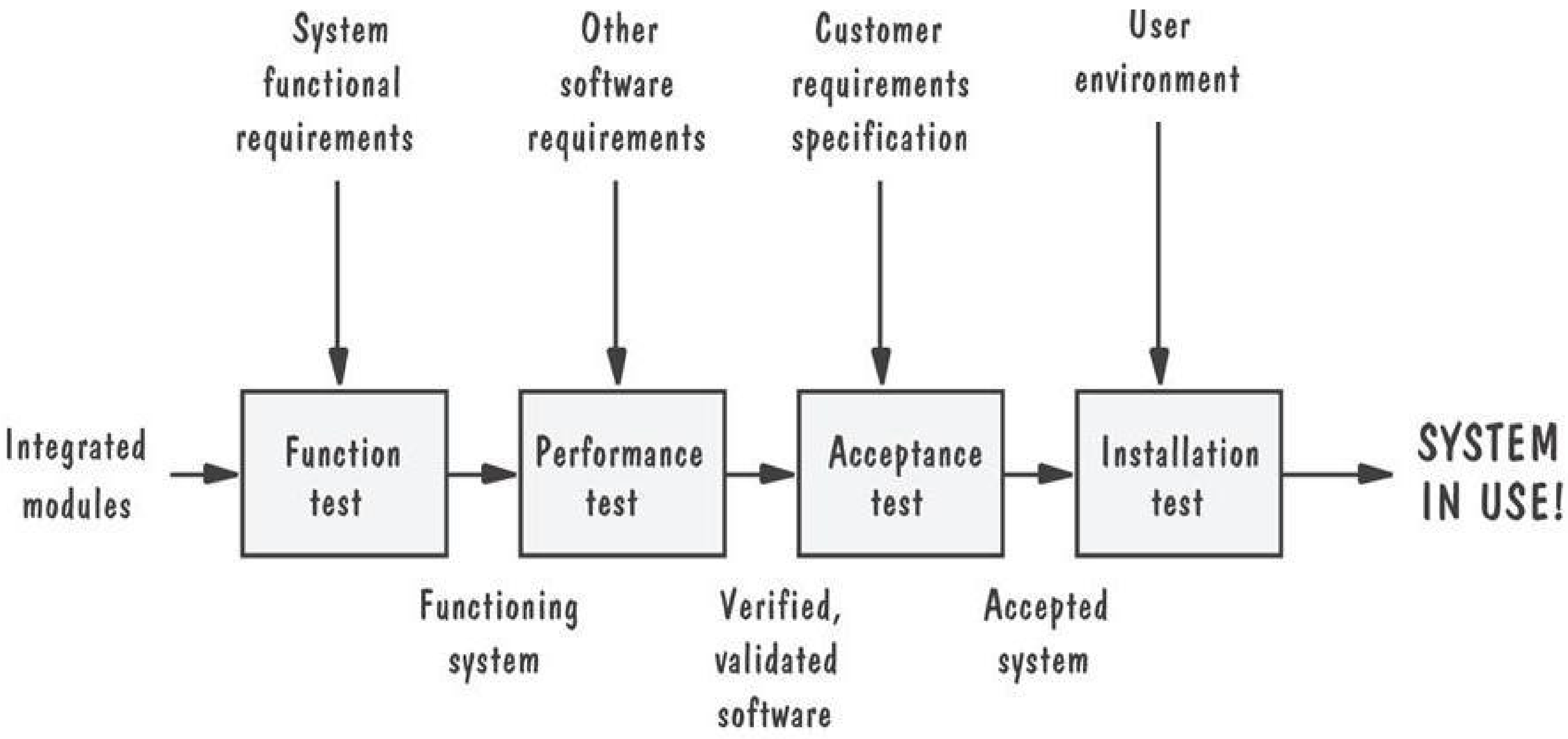
    安装测试报告
  - ② Build Plan（构造/集成计划）

    A: system testing of large system

    X: phased system testing (from small to large)

    Y: nested set of levels (from inner to outside)

- Pictorial representation of steps in testing process



Chapter 9 Testing the System

B: example: routes calls (in telecommunications
system—5 subsystems) (P421)

advantage: X: tested one level--deliver one level
Y: easy to check or correct fault

C: build plan ----incremental test plan for large
system(include how,where,when,who)

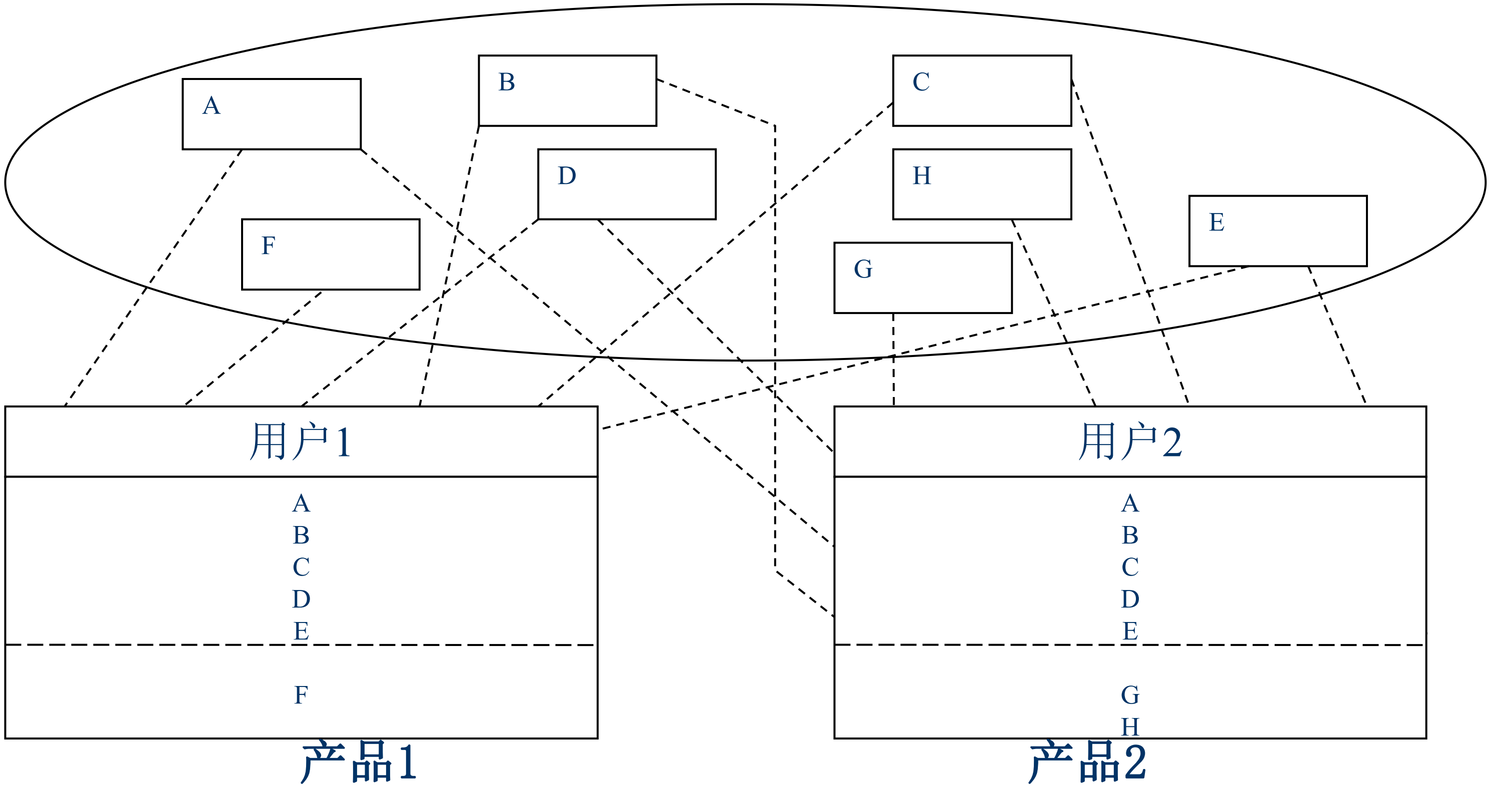
example: build plan for routes calls subsystem
X: table 9.1 : schedule table
Y: focus on: spin(subsystem) —hardware
and software resource + time/schedule
+ personnel + customer

Chapter 9 Testing the System

- 3. Configuration Management (SCM)(配置管理)
  - ① concepts A: system configuration (P423)

（交付给特定客户的一系列部件的集合）

此处重提配置管理，主要是强调大型多版本之下的基本测试原则，及部分回归的概念



Chapter 9 Testing the System

上图:两个产品具有不同的配置,其中A、B、C、D、E是核心部件/基本部件。

B: configuration management(P423)（配置管理）

（对系统不同软件配置的管理及控制方法（既有开发，也有测试），通过控制系统差别以降低风险，减少错误）。

C: importance: (of configuration management)

----coordinating efforts → effective configuration

协调各方努力 get 有效配置

Chapter 9 Testing the System

② 确定配置管理(SCM)计划

- 配置管理是整个开发过程中,用于管理软件产品内容的一系列措施.SCM流程旨在确保在任何时候产品的内容都是已知的、可用的，以及在设计到实施的过程中，产品的功能都是可跟踪的，可以完全控制和保护产品的内容。（例如曾经流行的配置管理工具clearcase等）
- 使用配置管理计划的必要性：
  - 如果不使用适当的配置管理计划，我们就经常无法知道哪一个模块是被确定、增强或测试过的。
  - 我们会开发出功能错误或是缺少功能的产品来，而且我们不知道哪些测试正在进行，哪些缺陷已被确定。
  - 我们不得不重新整理已经完成了的工作。

Chapter 9 Testing the System

- 配置管理计划关键的功能：
  - 每个产品元素（部件）版本的复件；
  - 对每一个基线修改的记录；（修改内容的说明：修改位置，修改时间，修改内容等等）
    - 基线：是指软件文档和其他资料的集合，它们代表了产品在某一时间点的情况(以及其他参考点)。
  - 谁进行了这个改变；
  - 他们什么时候改变的；(控制同时修改只能小规模有序进行)
  - 改变具体内容是什么；
  - 为什么他们要进行改变。
  - 可能涉及的其他功能等等(从涉及的基本文档开始统计)。 (改动配置所进行的测试甚至可发现以前的问题)

Chapter 9 Testing the System

- 技术支持经理负责草拟配置管理计划，并且和全组一起对它进行复核。
  - 确定配置控制委员会和控制过程
  - 确定任何所需的工具和设备。
  - 和小组一起重申整个计划，并取得一致意见

Chapter 9 Testing the System

- 软件配置管理的任务
  - ——制定软件配置管理计划
  - ——确定配置标识规则
  - ——实施变更控制
  - ——报告配置状态
  - ——进行配置审核
  - ——进行版本管理和发行管理

Chapter 9 Testing the System

- 忽视软件配置管理可能导致的混乱现象
  - 发错了版本
  - 安装后不工作
  - 异地不能正常工作
  - 已经解决的缺陷过后又出现错误
  - 开发人员把产品拿出去出售赢利
  - 找不到最新修改了的源程序
  - 找不到编程序的人

Chapter 9 Testing the System

- ③ versions and releases（版本和改进版本）

  versions----a particular configuration for a system（针对特定系统的特定配置）

  release----improved version/system（针对旧版本的改进版本）

  example:

  Version n.m=Version n and release m

  Version 3.0----Version3.1， release 3.2

  target of configuration management (in system testing)----accuracy and promptness in testing

[对于大型系统，人们绝不敢在忽略配置的情况下运行针对全部版本的测试用例集，一来太庞大麻烦；二来无针对性]

Chapter 9 Testing the System

- ④ production system vs. development system

  A:production system:

  B:development system:

- ⑤ regression testing（回归测试）

  ---- a test applied to a new version(verify the correctness of the old and new functions)

  example: version m-----version m+1 (P425)

  (test new version by old and new test cases)

  （大型系统的配置更动测试只是“部分”回归测试）

Chapter 9 Testing the System

- ⑥Change Control (变更控制) （主题：纳入配置管理）

  A: note: system testing ----find faults----correct
  faults----change system (under the control
  by configuration management team)

  B: problem caused by the change

  X: change any part of software----effect all
  documents and all developers

  Y: more developers make change----effect each
  other

  C: change control: 配置管理小组采取措施使不同开发者各自修改后的版本统一为一个版本.(使开发者不同时修改软件同一部分, 或采取其他额外步骤)

Chapter 9 Testing the System

- 4. Test team
  - ① focus on:

    A: unit and integration test----large role is developers

    B: function and performance test----large role is developers

    C: acceptance and installation test– large role is customers

    D: programmers can’t participate the testing of his own modules

Chapter 9 Testing the System

- ② several types (about testers)

  A: professional testers (who organize and run the tests, include former analysts, programmers, and designers) (P429)

  B: analysts: (who created requirement)

  C: system designers (who understand SRS and propose solution)

  D: configuration management specialists (who controls the changing of software)

  E: users: to evaluate issues that arise

Chapter 9 Testing the System

- ## 9.2 Function Testing

  Focus on: A: function testing ----black box

  B: function testing ----based on <SRS>

- ## 1.Purpose and Roles（目的和作用）
  - ①function testing : test for functional requirement (of <SRS>)
  - ②role: have a high probability of finding a fault (because one function testing only deal with the small set of components)

  example : water-monitoring system

  ----acknowledging change in dissolved oxygen, temperature, acidity, radioactivity.

Chapter 9 Testing the System

- ③ guidelines for function testing (P431—6 dots)

  A: 较高的查错概率

  B: 独立的测试团队

  C: 了解预期的输出结果

  D: 对合法与非法的输入都予以测试（假设是弱健壮等价类）

  E: 绝不能仅仅为了测试的方便而修改系统

  F: 停止测试应该有前提条件

- ④ 备注 A: 功能测试是在谨慎的受控制状态下进行的；

  B: 由于一次测试一个功能，若需要，功能测试可以早于整个系统的集成来进行.(某些重要功能)

  C: 有时功能测试又称线程测试. (线程: 与一个功能相关联

  的动作集合)

- ⑤ developing test cases (according <SRS>)

  example: word processing system (P431)

Chapter 9 Testing the System

- **2. Cause-and-Effect Graphs (因果图)**

  ① 说明:

  A: 功能测试的测试用例是由需求定义的功能说明部分转化而来.

  B: 测试用例不应产生冗余.
  (the test cases should not redundant)

  C: 测试用例应能发现需求中不完整和不明确的方面.

  ② 定义: 需求中的INPUT称为原因,OUTPUT称为结果.反映这种因果关系的布尔逻辑图称为因果图.

  基于因果图的测试用例产生过程:

  A: 由布尔逻辑和约束条件产生因果图.

  B: 将因果图转化为判定表. (decision table)

Chapter 9 Testing the System

C: 由判定表产生测试用例.

产生因果图的步骤:

A: 将需求分解为各个单独的功能. (separated functions)

B: 每个功能分出原因和结果. (cause + effect)

C: 中间节点 (extra nodes) 的产生.

③ Example: water-level monitoring system

A: analysis:

requirement: ”the system sends a message to the dam operator about the safety of lake level”

Input: syntax of the function: LEVEL(A,B)

“A”—height in meters of the water

“B”---the number of centimeters of rain

Chapter 9 Testing the System

PROCESSING: The function calculates whether the water level is within a safe range, is too high, or is too low.

output: result of level(A,B):

The screen shows one of the following messages:

- “LEVEL = SAFE”, when the result is safe or low.
- “LEVEL = HIGH”, when the result is high.
- “INVALID SYNTAX”

depending on the result of the calculation.

Chapter 9 Testing the System

B: cause-and-effect graph

causes: 5 (P399)

1.The first five characters of the command “LEVEL”

2.The command contains exactly two parameters separated by a comma and enclosed in parentheses.

3.The parameters A and B are real numbers such that the water level is calculated to be LOW

4.The parameters A and B are real numbers such that the water level is calculated to be SAFE.

5.The parameters A and B are real numbers such that the water level is calculated to be HIGH.

Chapter 9 Testing the System

effects: 3 (P399)

1.The message “LEVEL = SAFE” is displayed on the screen.
2.The message “LEVEL = HIGH” is displayed on the screen.
3.The message “LNVALID SYNTAX” is printed out

intermediate nodes: 2 (P399)

1.The command is syntactically valid.
2.The operands are syntactically valid.

C-and-E: --------- fig9.5 (P399)

C: decision table: table9.2

D: advantage of C-and-E

disadvantage of C-and-E

Chapter 9 Testing the System

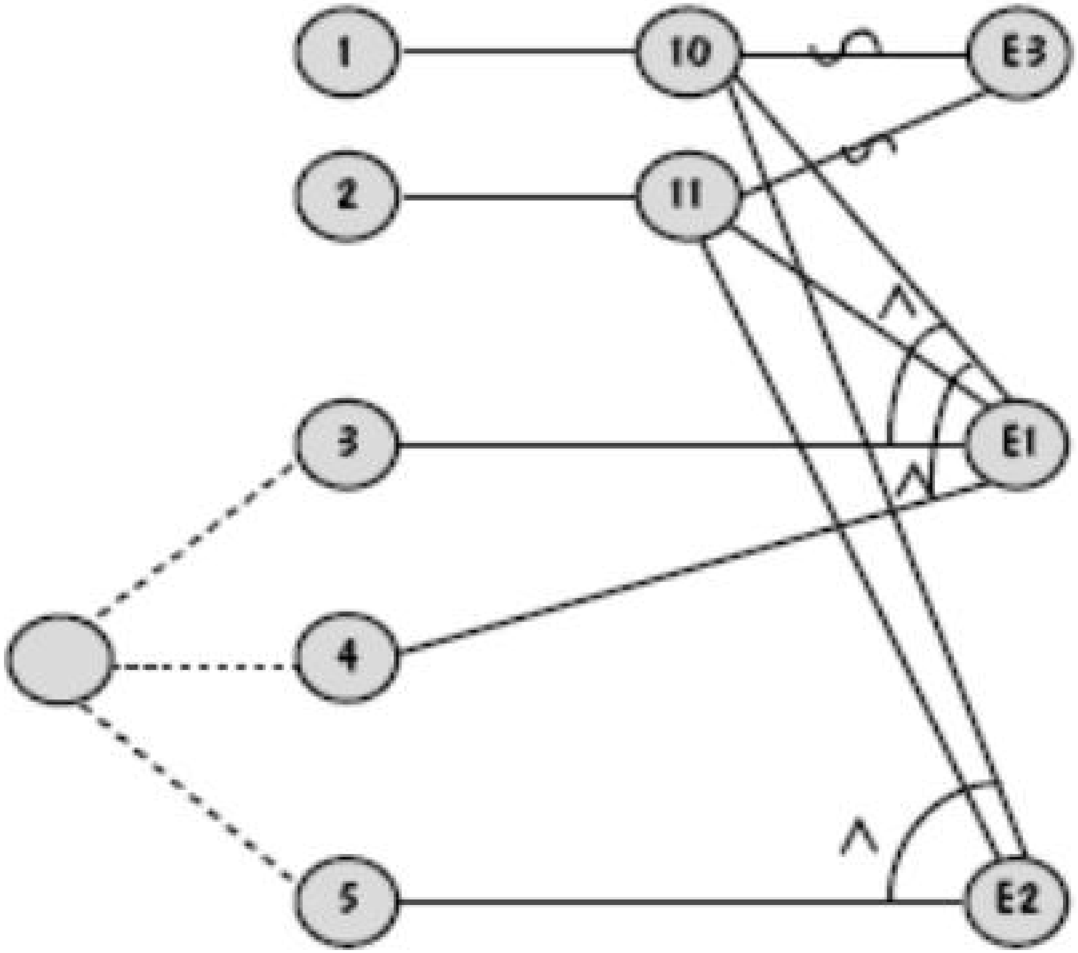


Table 9.2. Decision table for cause-and-effect graph.

	Test 1	*Test 2*	*Test 3*	*Test 4*	*Test 5*
Cause 1	I	I	I	S	I
Cause 2	I	I	I	X	S
Cause 3	I	S	S	X	X
Cause 4	S	I	S	X	X
Cause 5	S	S	I	X	X
Effect 1	P	P	A	A	A
Effect 2	A	A	P	A	A
Effect 3	A	A	A	P	P

Chapter 9 Testing the System

- ## 9.3 Performance Testing

Focus on: performance test $\xrightarrow[\text{addresses}]{}$ nonfunctional requirement(in SRS)

- ## 1.Purpose and Roles
  - ① performance testing : test for nonfunctional requirement (of <SRS>)

    example : calculate the trajectory of a rocket

    (the speed of response, accuracy, data accessibility, etc.)
  - ② note :

    A: performance testing(by a team) → get result → give to developer and customer

    B: hardware engineer € test team

Chapter 9 Testing the System

- 2. Types of Performance tests
  - ①several types of performance tests

    A: stress tests（压力测试 / 强度测试）
    (在短时间内加载极限负荷, 以验证系统能力)

    B: volume tests（巨量数据测试 / 容量测试）
    (验证系统处理巨量数据的能力)

    C: configuration tests（配置测试） ---构建测试用例对系统软硬件的各种配置(最小到最大)进行测试。

    D: compatibility tests（兼容性测试）---测试接口等

    E: regression tests (by old and new test cases)

    F: security tests (security requirement)

    G: timing tests

    H: environment tests

Chapter 9 Testing the System

I: quality tests

J: recovery tests

K: human factors test (use interface-----usability)

- ②focus on:

A: performance test ----more difficult than function test（平时缺少必要的场景）

B: requirement must be explicit to ensure performance test

C: requirement quality is often been reflected in the ease of performance testing (需求质量通常可以反映在性能测试的容易度上)

Chapter 9 Testing the System

- 9.4 Reliability,Availability,Maintainability

（可靠性，可用性，可维护性） (关于性能测试)

focus:A: performance test----Reliability, Availability, Maintainability----is critical issues

B: it is difficult to measure directly before delivery

- 1. Definitions

software reliability：(probability ： 0 → 1)

----软件系统在给定的时间间隔和给定条件下运行成功的概率

availability: (0/1)

----软件系统在给定的时间点（按SRS要求）成功运行的概率

Chapter 9 Testing the System

maintainability:（ 0⟶1 ）

----是指在给定的使用条件（预定的时间间隔、维护程序、维护资源之下进行维护）下，维护活动能被执行的概率

- 2. Four different levels of failure severity(严重性)

  A: reason: because reliability, availability, and maintainability are defined in terms of failures, they must be measured once the system is complete and working

  B: four different levels (P438)

- 3. Failure data：根据失效时间记录，来理解系统的不确定性，以图改进. 或者以此理解系统可靠性的改进程度.

Chapter 9 Testing the System

- 9.5 Acceptance Testing（确认/验收测试）
- 1. Purpose and roles
  - ① acceptance testing: customer check if the system meet the need in requirement definition
  - ② roles: customer ----play large roles (in acceptance test，customer write, conduct, and evaluate the test result)

    developers----play small roles (answer or explain technical questions when the customer requests)

Chapter 9 Testing the System

- 2. Types of Acceptance tests
  - ① benchmark test（基准测试） (formal test)

    A: customer prepare test cases (with formal cases)

    B: install on experimental basis

    C: customer evaluate the performance
  - ② pilot test（引导测试） (informal test)

    A: user/developer install on a experimental basis

    B: pilot test rely on daily working(no formal cases)

    alpha test: pilot test in organization (by developer)

    beta test: customer’s pilot (in working site)

    example ---microsoft corporation’s test strategy

Chapter 9 Testing the System

- ③ Parallel test

  A: definition:

  （新老版本并行运转，使用户逐渐适应新版软件系统。）

  X: used in phased development model

  Y: new system operates in parallel with the previous version (no formal test cases)

  B: features:

  X: users will be gradually become accustomed to the new version (by comparing and contrasting them)

  Y: allowing the skeptical users to build their confidence in the new version

Chapter 9 Testing the System

- 3. Results of Acceptance Tests
  - ① It is the customer’s chance to verify that what was wanted is what was built.
  - ② It will help customer to discover ambiguous aspects (in definition) .
  - ③ Change in requirements may be needed.
  - ④ Configuration management staff identify the changes and record other modifications in design , implementation, and testing

（客户根据验收测试的结果，将告诉开发方哪些需求没有得到满足，哪些是因为需求变化而必须删除、修改或增加，配置管理人员将配合记录软件所有阶段的变更。）

Chapter 9 Testing the System

- 9.6 Installation Testing（安装测试）
- 1. definition: testing in user's environment (to solve the problems caused by the differences between the developing environment and the user's environment)

  example: the primitive data file and devices
- 2. focus on: (about installation testing)

  A: completeness (of the installed system of functional or nonfunctional features)

  B: verification: final checking (in working site)

  example : system working in a ship 。