

# 软件质量保证与软件测试

## --系统测试

主讲教师:

Email:

西安交大软件学院

## ■ 系统测试概要介绍

- 系统测试是测试的三级层次中最接近日常测试实践的
- 我们购买东西时要进行测试，这些是根据预期来评估产品、而不是根据规格说明或标准
- 测试软件系统是否符合所有需求，包括功能性需求（主要依据是需求规格说明书）和非功能性需求（如可用性测试、性能测试等）
- 一般由独立测试人员（测试小组）执行，通常采用黑盒测试方式

西安交大软件学院

## ■ 系统测试之功能性测试

- 目标
  - 检查应用程序是否达到了它预期的要求
- 判断依据
  - 行为/功能说明书
- 测试用例
  - 一系列的原子系统功能（Atomic System Function, ASF）

## ■ 系统测试之压力测试

- 将系统压到极限，看其极端情况下的表现
- 使系统承受高负载或强度的检验，在很短时间内达到数据峰值或操作数量峰值
  - 如：12306宕机事件
- 极限压力测试举例：
  - 1) 接收大数据量的数据文件时间
  - 2) 大数据恢复时间
  - 3) 大数据导入导出时间
  - 4) 大批量录入数据时间
  - 5) 大数据量的计算时间
  - 6) 多客户机同时进行某一个提交操作
  - 7) 采用测试工具软件
  - 8) 编写测试脚本程序
  - 9) 大数据量的查询统计时间

## ■ 系统测试之其它测试

- 系统测试之性能测试
  - 用户的角度观察
    - delay, throughput
  - 系统的支撑强度
    - memory、CPU等
- 系统测试之易用性测试
  - 系统操作过程中人的要素
    - GUI, messages, reports, ...

西安交大软件学院

## ■ 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

西安交大软件学院

## ■ 线索 (thread)

- 线索的多种看法
  - 一般使用的场景
  - 系统级测试用例
  - 激励/响应对
  - 由系统级输入序列产生的行为
  - 端口输入和输出事件的交替序列
  - 系统状态机描述中的转换序列
  - 对象消息和方法执行的交替序列
  - 机器指令序列
  - 源指令序列
  - MM-路径序列
  - 原子系统功能序列
- 根据系统级线索看系统测试

西安交大软件学院

## ■ 线索的层次

- 线索有不同层次：
  - 单元级线索是源指令执行时路径，或DD-路径序列
  - 集成级线索是模块执行和消息交替序列，即MM-路径
  - 系统级线索是原子系统功能（ASF）序列
    - 由于ASFs在输入和输出时有端口事件，因此ASFs包含端口输入和输出事件的交替序列
- 线索提供三层测试的统一视图：
  - 单元测试进行单个函数测试
  - 单元之间的集成测试检查交互
  - 系统测试检查ASFs之间的交互

西安交大软件学院

## 问题

- 本章重点在于系统级线索，并回答如下问题：
- 线索有多大？
- 到哪里寻找线索？
- 怎样测试线索？

西安交大软件学院

## 线索定义

- 定义：
  - 原子系统功能（ASF）是一种在系统层可以观察得到的端口输入和输出事件的行动
  - ASF开始于一个端口输入事件，遍历一个或多个MM-路径的一部分，以一个端口事件结束
- 在事件驱动系统中，ASF由事件静止点分开
  - 当系统（接近）空闲、等待端口输入事件以触发进一步处理时，会出现事件静止
- 例子：**ATM卡输入、PIN输入、现金给付和会话关闭都是SATM系统的ASF**

西安交大软件学院

## 线索定义（续）

- 原子系统功能表示集成测试与系统测试之间的缝隙，是集成测试的最大测试项，是系统测试的最小测试项，因此可以在两个级别上测试ASF
- 举例---SATM的数字输入ASF
  - 系统测试过程
    - 端口输入事件是由keySensor检测到的按下物理键，并作为字符串变量发送给GetPIN。GetPIN确定是否按下数字键或取消键，并做出相应的应答。ASF最后显示屏幕2或4
  - 集成测试过程
    - 不要求系统输入键并在屏幕上显示，而是使用驱动器提供的这些功能，并通过集成测试检查数字输入ASF

西安交大软件学院

## 线索定义（续）

- ASF图（给定通过ASF定义的系统）
  - 是一种有向图，其中的节点表示ASF，边表示串行流
- 源ASF/汇ASF
  - 是一种原子系统功能，在系统ASF图中作为源/汇节点出现
  - 例子---SATM系统
    - “ATM卡输入”—源ASF；会话结束ASF---汇ASF



西安交大软件学院

## ■ 线索定义（续）

### □ 系统线索

- 在系统的ASF图中，是一条从源ASF到汇ASF的路径

- 上述这组定义提供了线索更广的视图的内聚集，以非常短的线索开始（一个单元内），以系统级线索之间的交互结束

西安交大软件学院

## ■ 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

西安交大软件学院

## ■ 需求规格说明中的基本概念

### □ 目标

- 根据一组需求规格说明构造，讨论系统测试

### □ 五种需求规格说明的构造

- 数据(data)
- 行动(actions)
- 设备(devices)
- 事件(events)
- 线索(threads)

西安交大软件学院

## ■ 需求规格说明中的基本概念

### ■ 数据(data)

- 当系统以其数据描述时，关注的是系统所使用和创建的信息
- 采用变量、数据结构、字段、记录、数据存储和文件来描述数据
- 高层数据描述常见的模型是E/R图，在更细层次上使用一些常规表达式（如Jackson图或数据结构图）
- 数据是指经过初始化、存储、更新或（可能）销毁的信息
- 有时线索可以直接通过数据模型标识
- 数据实体之间的关系可以是一对一、一对多、多对一或多对多，这些差别在处理数据的线索中都有应用
- 只读数据是一种源ASF指示器

西安交大软件学院

## ■ 需求规格说明中的基本概念

### ■ 行动/行为(actions)

- 行动有输入和输出，输入输出可以是数据，也可以是端口事件
- 同义词有数据转换、控制转换、处理、活动、任务、方法和服务
- 行动可以分解为低层活动，如数据流图；
- 行动的输入/输出视图正好是功能性测试的基础，行动的分解（以及最终实现）则是结构性测试的基础

西安交大软件学院

## ■ 需求规格说明中的基本概念

### ■ 设备(devices)

- 每个系统都有端口设备，这些端口设备是系统级输入和输出(端口事件)的源和目的地
- 区分端口和端口设备之间的微小差别有时对测试人员很有帮助
  - 端口是I/O设备接入系统的点，物理行动在端口设备上发生，要从物理行动转换为逻辑行动(或从逻辑行动转换为物理行动)
  - 如果没有实际端口设备，系统测试可以通过"将端口边界向内移动"到端口事件的逻辑实例上实现
- 考虑端口有助于定义功能性测试和系统测试所需要的输入空间
- 输出设备提供技术输出的功能性测试信息

西安交大软件学院

## ■ 需求规格说明中的基本概念

### ■ 事件(events)

- 事件是发生在端口设备上的系统级输入/输出
- 事件可以是行动的输入和输出
- 事件可以是离散的（如SATM键盘输入）、也可以是连续的（如温度）
  - 离散事件必须有一定的持续时间，这在实时系统中是一个关键因素
- 事件是现实世界物理事件和这些事件的内部逻辑表示的转换点
- 端口输入是物理到逻辑的转换，端口输出是逻辑到物理的转换
- 系统测试人员应该关注事件的物理层面、而集成测试人员关注事件的逻辑层面

!仅限个人使用\*请勿上传至互联网\*违者必究! 西安交大软件学院

## ■ 需求规格说明中的基本概念

### ■ 线索(threads)

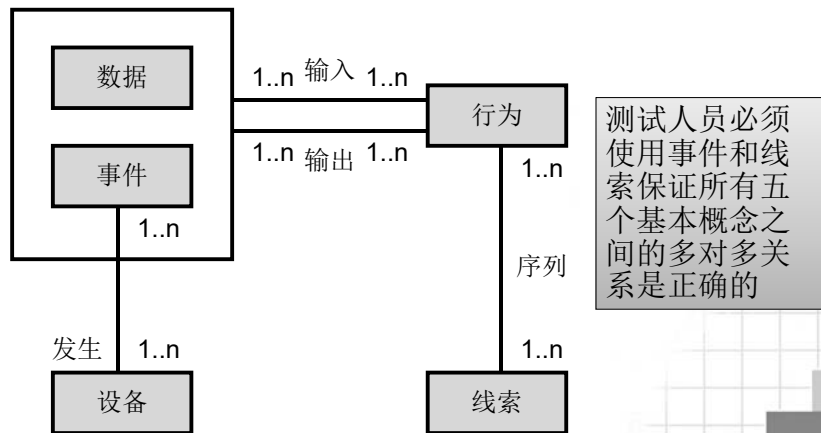
- 5种构造中最不常用
- 使用快速原型法并结合场景记录器，会用到线索
- 在控制模型中容易找到线索，但控制模型只是模型而非实现系统

西安交大软件学院

Downloader: 王博玉

## 基本概念之间的关系

上述基本概念的实体/关系(E/R)模型



西安交大软件学院

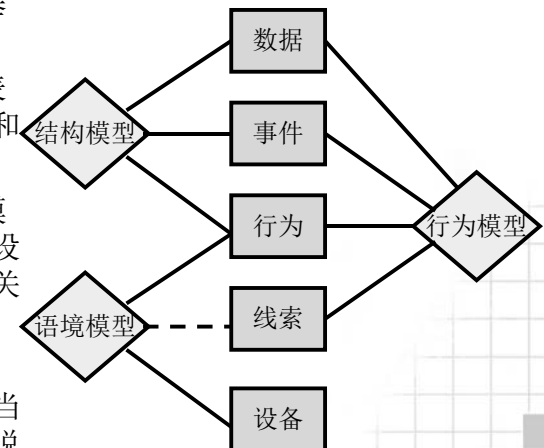
## 基本结构之间的建模关系

■需求规格说明模型的基本形式:

■结构模型用于开发,表示功能分解、数据分解和组件之间的接口

■语境模型常常是结构模型的开始点,强调系统设备、行为,非常间接地关注线索

■行为模型(控制模型)集成四种构造,选择适当的行为模型是需求规格说明的基础



西安交大软件学院

## 建模通常采用的方式

■建模的方式: 决策表、有限状态机、Petri网等

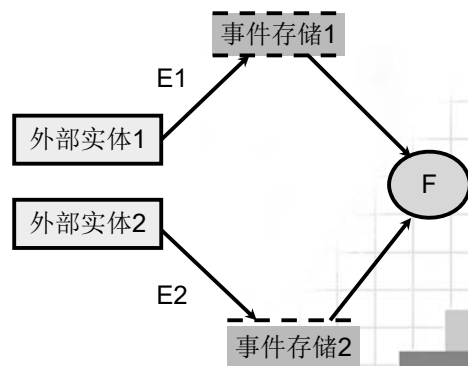
■考虑某个系统,其功能F只有在前提事件E1和E2发生时才会出现,这些事件能够以任意顺序出现

■使用事件划分概念进行建模,如右图

■在事件划分视图中,事件E1、E2根据外部设备分别发生,当这些事件发生时,要保存在各自的事件存储中

■当两个事件发生后,功能F从事件存储中得到其前提信息

■注意: 不能通过模型区分到底哪个事件先发生,只知道两个事件一定发生



功能F的事件划分视图

西安交大软件学院

## 建模通常采用的方式

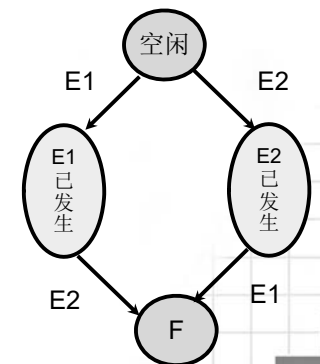
■考虑某个系统,其功能F只有在前提事件E1和E2发生时才会出现,这些事件能够以任意顺序出现

■使用有限状态机进行建模,如右图

■其中的状态记录所发生的事件

■状态机视图显式地显示事件的两种顺序

■对于这两种模型,状态机对于测试人员来说更有用,因为路径马上可以转换为线索



功能F的有限状态机

西安交大软件学院

## 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

西安交大软件学院

## 寻找线索 --- SATM系统

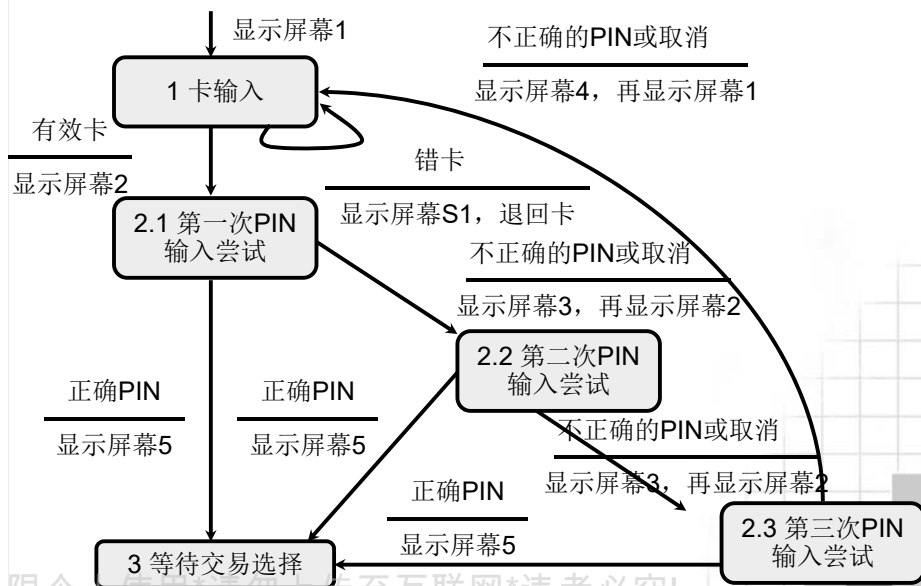
- 目的：根据模型得到系统
- SATM系统的有限状态机模型

- 状态对应过程的阶段，转移由逻辑（而不是端口）事件引起

- 有限状态机可以为层次结构，宏状态可以分解得到更详细状态
- 如：“PIN输入”状态分解如下页ppt所示



## “PIN输入”有限状态机

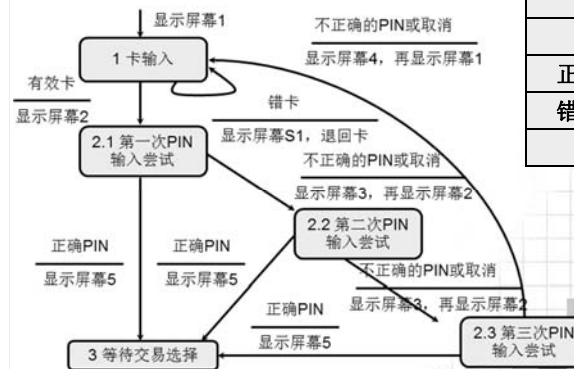


!仅限个人使用\*请勿上传至互联网\*违者必究 西安交大软件学院

## “PIN输入”有限状态机中的事件

为了开始线索标识过程，首先列出表示状态转移的端口事件

端口输入事件	端口输出事件
有效卡	显示屏幕 2
错卡	显示屏幕 1
正确的PIN	显示屏幕 5
错误的PIN	显示屏幕 3、2
取消	显示屏幕 3、2



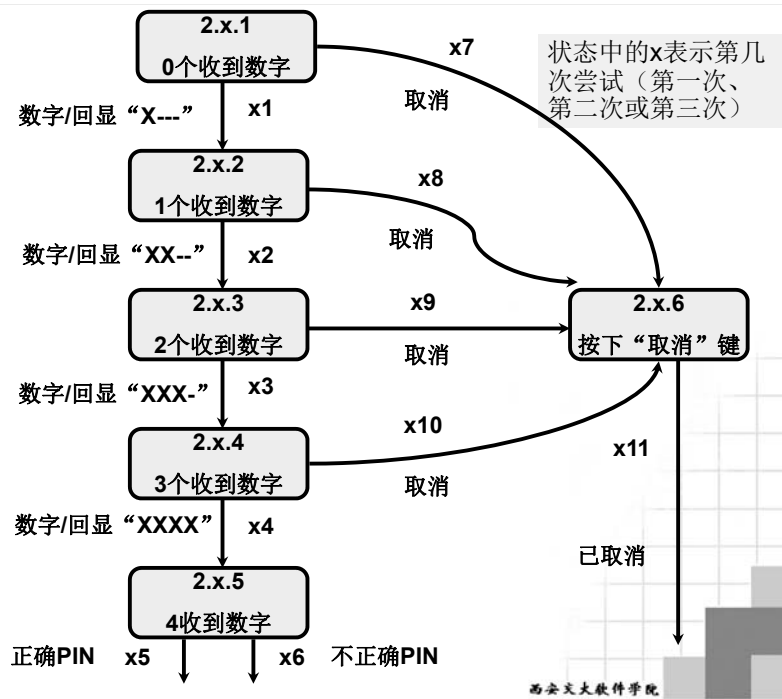
- 注意：“正确的PIN”和“错误的PIN”是合成端口事件，实际上不能输入完整的PIN，在输入数字的任何时候都可以键入取消键

- 更细节信息如下页所示

Downloader: 王博玉

西安交大软件学院

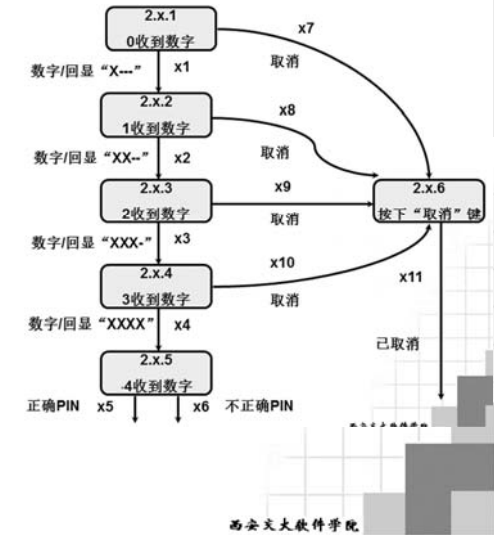
## “PIN输入尝试”有限状态机



## “PIN输入尝试”有限状态机中的端口事件

端口输入事件	端口输出事件
数字 取消	回显“X---” 回显“XX--” 回显“XXX-” 回显“XXXX”

- 除真正的端口事件之外，还有三个逻辑输出事件（正确的PIN、错误的PIN、取消），正好对应“PIN输入”状态机中的高层事件

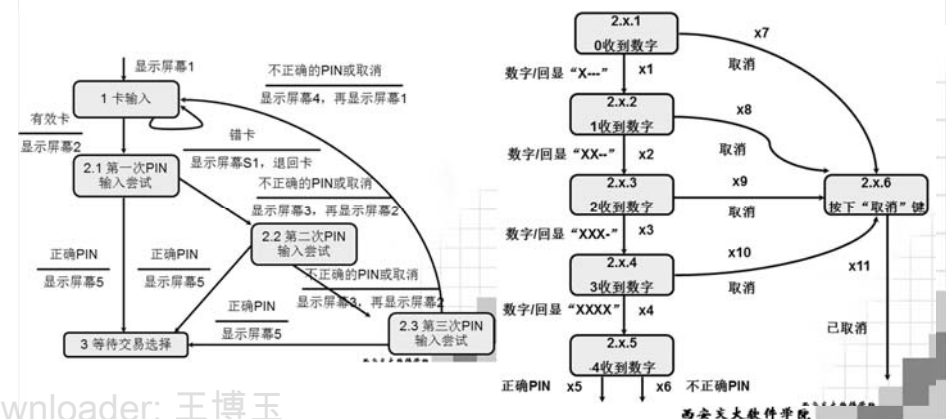


## 寻找线索

- 有限状态机的层次结构，使线索的数量成倍增长
- 对于“PIN输入”状态机，从“第一次PIN输入尝试”状态到“等待交易选择”或“卡输入”状态之间有156条不同路径
- 有31条路径对应最终正确的PIN（第一次尝试有1条、第二次尝试有5条、第三次尝试有25条）
- 其他125条路径对应错误的数字或按下取消键
- 这是相当典型的比例
- 系统的输入部分，特别是交互系统的输入部分，往往有大量线索处理输入错误或例外

## 练习

- 找出31条对应正确PIN的路径
- 其他125条路径对应错误的数字或按下取消键，请从图中找出这些路径





## 寻找线索

- 给出状态机是一种很好的方式，其中：实际端口输入事件引起转移，转移行动是端口输出事件
- 如果有这种状态机，那么为这些线索生成系统测试用例就成为机械过程
- 即：只需追踪路径转移，并在遍历路径时注意端口输入和输出
- 这种交替序列由测试执行器（人或程序）完成
- 对于包含较多状态的系统，需要注意内存问题和依赖问题

西安交大软件学院

## SATM层次状态机中的两条路径

- 左边表对应于PIN在第一次尝试被正确输入的一个线索
- 右边表对应于PIN在第一次尝试被错误输入、在第二次尝试时输入第三个数字后键入了取消键、到第三次才输入正确的一个线索。（为了使测试用例明确，我们假设正确的PIN是1234）

端口输入事件	端口输出事件
（第一次尝试）	屏幕2显示“——”
按下1	屏幕2显示“X——”
按下2	屏幕2显示“xx——”
按下3	屏幕2显示“XXX—”
按下4 （正确的PIN）	屏幕2显示“XXXX” 显示屏幕5

线索可以是另一个线索的子集

端口输入事件	端口输出事件
（第一次尝试）	屏幕2显示“——”
按下1	屏幕2显示“X——”
按下2	屏幕2显示“XX——”
按下3	屏幕2显示“XXX—”
按下5 （错误的PIN）	屏幕2显示“XXXX” 显示屏幕3
（第二次尝试）	屏幕2显示“——”
按下1	屏幕2显示“X——”
按下2	屏幕2显示“XX——”
按下3	屏幕2显示“XXX—”
按下取消键 （第二次尝试结束）	显示屏幕3
（第三次尝试）	屏幕2显示“——”
按下1	屏幕2显示“X——”
按下2	屏幕2显示“XX——”
按下3	屏幕2显示“XXX—”
按下4 （正确的PIN）	屏幕2显示“XXXX” 显示屏幕5

## 练习

- 尝试找出其他线索，如：
- 第一次尝试取消
- 第一次尝试取消、第二次尝试输入错误
- 第一次尝试输入2个位数字后取消
- .....

## 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

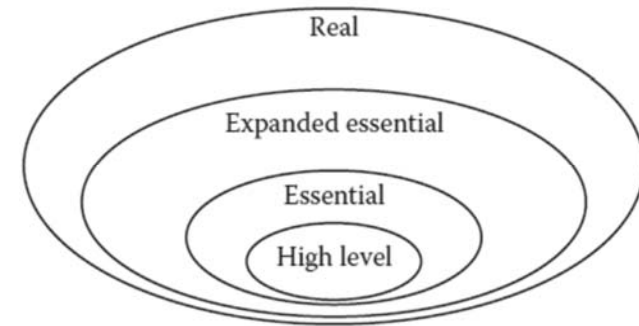
## ■ 基于用例的线索

- 用例是UML中的核心部分
- 用例的主要优点是它们容易被客户/用户和开发人员理解
- 用例关注的是系统的行为，而不是系统的结构
- 因此，使用用例来寻找系统线索是一个合理选择

西安交大软件学院

## ■ 用例的层次结构

- 用例模型是层次结构的，每一层是在前一层的基础上增加信息
- 分为：高层用例（类似于一个敏捷用户故事）、基础用例、扩展用例、实际用例



## ■ 高层用例

- 高层用例类似于敏捷开发中的用户故事
- 一组高层用例给出了系统行为的概要描述
- 第一次尝试输入正确PIN的 高层用例

Use case name	Correct PIN entry on first try
Use case ID	HLUC-1
Description	A customer enters the PIN number correctly on the first attempt.

!仅限个人使用\*请勿上传至互联网\*违者必究!

西安交大软件学院

## ■ 基础用例

- 基础用例在高层用例的基础上增加端口输入和端口输出事件
- 在这个阶段，端口边界对客户和开发者变得清楚
- 第一次尝试输入正确PIN的 基础用例

Use case name	Correct PIN entry on first try
Use case ID	EUC-1
Description	A customer enters the PIN number correctly on the first attempt.
Event Sequence	
Input events	Output events
	1. Screen 2 shows '- - - -'
2. Customer touches 1st digit	
	3. Screen 2 shows '- * - -'
4. Customer touches 2nd digit	
	5. Screen 2 shows '- * * -'
6. Customer touches 3rd digit	
	7. Screen 2 shows '- * * *'
8. Customer touches 4th digit	
	9. Screen 2 shows '* * * *'
10. Customer touches Enter	
	11. Screen 5 is displayed

Downloader: 王博玉

## 扩展用例

- 扩展用例在基础用例的基础上增加前置条件和后置条件
- 第一次尝试输入正确PIN的扩展用例

Use case name	Correct PIN entry on first try
Use case ID	EEUC-1
Description	A customer enters the PIN number correctly on the first attempt.
Preconditions	1. The expected PIN is known 2. Screen 2 is displayed
Event Sequence	
Input events	Output events
	1. Screen 2 shows '- - - '
2. Customer touches 1st digit	
	3. Screen 2 shows '- - * '
4. Customer touches 2nd digit	
	5. Screen 2 shows '- - * * '
6. Customer touches 3rd digit	
	7. Screen 2 shows '- * * * '
8. Customer touches 4th digit	
	9. Screen 2 shows '* * * * '
10. Customer touches Enter	
	11. Screen 5 is displayed
Cross reference to functions	
Postconditions	Select Transaction screen is active

## 实际用例

- 实际用例是实际的系统级测试用例
- 端口事件的抽象名字用实际的值代替
- 某种形式的测试数据需要准备好
- 第一次尝试输入正确PIN的实际用例

Use case name	Correct PIN entry on first try
Use case ID	RUC-1
Description	A customer enters the PIN number correctly on the first attempt.
Preconditions	1. The expected PIN is "2468" 2. Screen 2 is displayed
Event Sequence	
Input events	Output events
	1. Screen 2 shows '- - - '
2. Customer touches digit 2	
	3. Screen 2 shows '- - * '
4. Customer touches digit 4	
	5. Screen 2 shows '- - * * '
6. Customer touches digit 6	
	7. Screen 2 shows '- * * * '
8. Customer touches digit 8	
	9. Screen 2 shows '* * * * '
10. Customer touches Enter	
	11. Screen 5 is displayed
Cross reference to functions	(normally done at this point)
Postconditions	Select Transaction is active

## 系统级测试用例

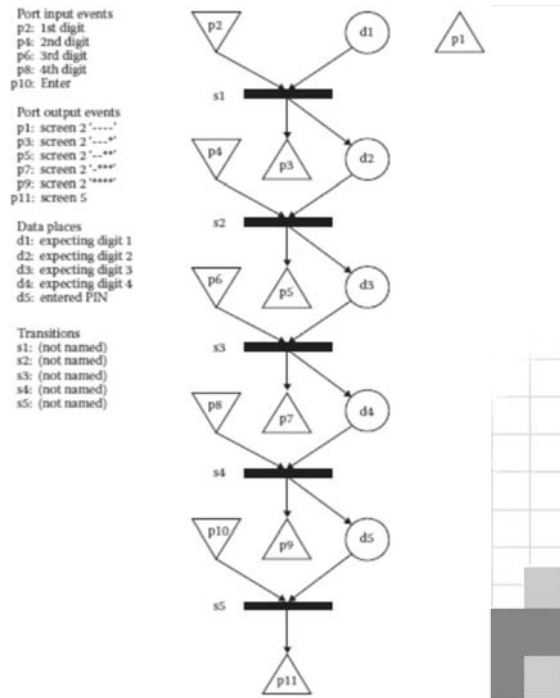
- 系统级测试用例与实际用例相同，除了后面加上测试的结果、以及由谁测试的等信息

Test case name	Correct PIN entry on first try
Test case ID	TC-1
Description	A customer enters the PIN number correctly on the first attempt.
Preconditions needed to run this test case	1. The expected PIN is "2468" 2. Screen 2 is displayed
Event Sequence	
Input events (performed by tester)	Output events (observed by system tester)
	1. Screen 2 shows '- - - '
2. Touch digit 2	
	3. Screen 2 shows '- - * '
4. Customer touches digit 4	
	5. Screen 2 shows '- - * * '
6. Customer touches digit 6	
	7. Screen 2 shows '- * * * '
8. Customer touches digit 8	
	9. Screen 2 shows '- - * * '
10. Customer touches Enter	
	11. Screen 5 is displayed
Cross reference to functions	
Postconditions	Select Transaction is active
Test execution result?	Pass/Fail
Test run by	<tester's name> date

## 将用例转换为事件驱动Petri网

- 事件驱动Petri网（Event-Driven Petri Nets, EDPN）适合描述事件驱动系统，尤其是对上下文敏感的端口输入事件的系统
- 在EDPN图中，端口事件（port events）由三角形表示、数据库所（data places）由圆形表示、变迁（transitions）由窄矩形表示、输入输出连接由箭头表示
- 关于EDPN的详细信息，可以查阅教材的第4章或者其它材料

## 第一次尝试输入正确PIN的EDPN图



## 第一次尝试输入正确PIN的EDPN图

- 从用例自动得到EDPN是部分成功的
  - 从事件序列的输入部分可以得到端口输入事件
  - 从事件序列的输出部分可以得到端口输出事件
  - 输入输出事件的交织也可以保持
  - 前置条件和后置条件映射为数据库所
- 但是, 还存在一些问题:
  - 输出事件p1是孤立的, 它不是由某个迁移创建的
  - 迁移没有命名
  - 假定因果连接是立即的
  - 没有提供可能产生的中间数据
  - 数据库所d1-d5在用例中并没有出现

西安交大软件学院

## 解决问题的思路

- 需要定义形式良好的用例, 一个形式良好的用例应该满足如下需求:
  - 事件序列不能以一个输出事件开始。这只能考虑作为前置条件
  - 事件序列不能以一个输入事件结束。这只能考虑作为后置条件
  - 前置条件必须对用例是充要的。不存在多余的前置条件, 每个前置条件对用例来说是必须的
  - 后置条件必须对用例是充要的
  - 至少有一个前置条件和一个后置条件

## 从用例得到EDPN的目的

- 从用例得到EDPN的目的是利用EDPN的分析能力, 可以进行的分析如下:
  - 用例间的交互, 如一个用例是另一个用例的前提
  - 相互冲突的用例
  - 上下文敏感的输入事件
  - 翻转 (inverse) 用例, 一个用例 “解开(undo)” 另一个用例

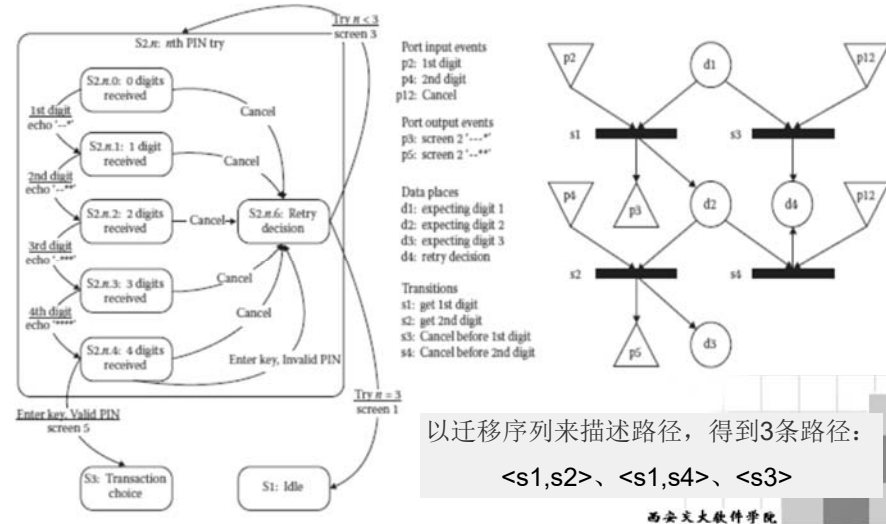
## 将有限状态机转换为EDPN

- 有限状态机是Petri网的一种特殊情况，对于每个迁移有一个输入库所和一个输出库所
- 有限状态机也可以转换为EDPN
- 有两种方法来描述路径：
  - 以端口输入事件序列来描述
  - 以迁移序列来描述

西安交大软件学院

## 将有限状态机转换为EDPN - 示例

### ■ 图14.7的部分变为EDPN



## 讨论

- 用例、有限状态机和EDPN，哪一种对系统测试最有用？
  - 用例最适合客户和开发人员之间进行交流，但不支持太多的分析
  - 有限状态机很常用，但常会碰到状态空间爆炸问题
  - 用例和有限状态机都可以转化为EDPN；EDPN的最大优点是多个EDPN可以组合起来；未标记EDPN和不同标记（执行序列）EDPN之间的竞争/扩展关系使得EDPN成为系统测试的优先选择

## 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

## ■ 线索测试的结构策略

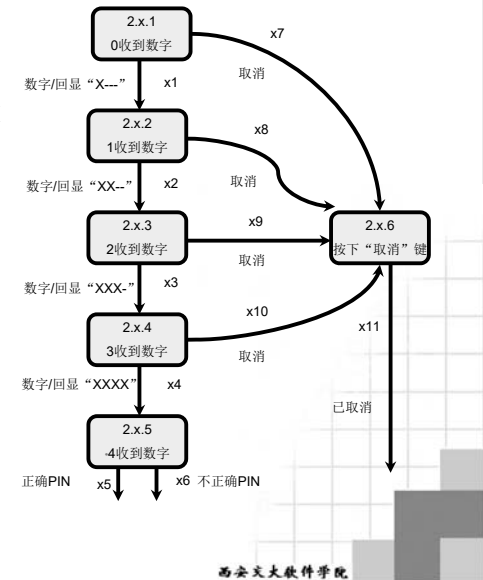
- 生成线索测试用例很容易，但是确定实际使用哪个测试用例更复杂
- 在系统级也同样存在单元级的路径爆炸问题
- 可以使用有向图更明确地选择要测试的线索
  - 当在层次结构中组织状态机时，可以自底向上进行
  - 有限状态机是有向图，可以使用与单元级相同的测试覆盖指标

西安交大软件学院

## ■ 自底向上组织线索

- 在“PIN输入尝试”状态机中，共有六条路径
- 遍历这六条路径，需要测试三件事：1) 正确识别并回显所输入的数字、2) 响应“取消”键入、3) 匹配预期PIN和所输入的PIN
- 这些路径在表中描述为图中的转移序列(“取消”键入采用C表示)

输入事件序列	转移的路径
1234	x1, x2, x3, x4, x5
1235	x1, x2, x3, x4, x6
C	x7, x11
1C	x1, x8, x11
12C	x1, x2, x9, x11
123C	x1, x2, x3, x10, x11

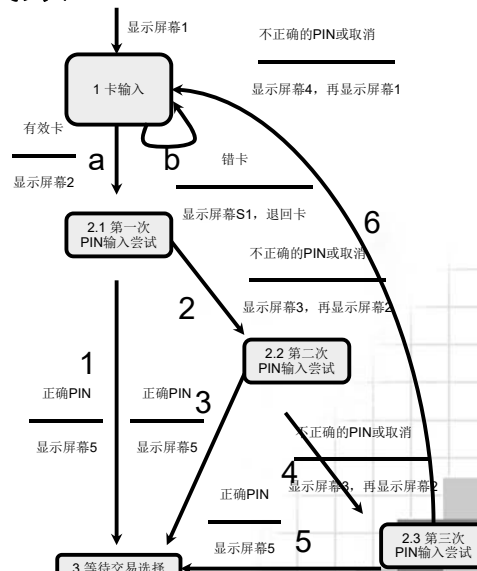


西安交大软件学院

## ■ 自底向上组织线索

- 一旦测试了这个部分之后，可以上升到“PIN输入”状态机，共有四条路径
- 这些路径对应三次尝试机制以及向用户显示的屏幕序列
- 在下表中，PIN输入状态机中的路径(如右图所示)用转移序列命名

输入事件序列	路径转移
1234	1
12351234	2, 3
1235C1234	2, 4, 5
CCC	2, 4, 6



西安交大软件学院

## ■ 自底向上组织线索

- 正如在单元测试中所见，结构性测试可能会产生误导
  - 假设是路径遍历会发现缺陷，遍历各种路径会产生冗余
- 这些线索还有一个更严重的问题：由于是分层状态机，因此实际上这些线索不能单独执行
  - 如：按下“1、2、3、5”，会经过“PIN输入尝试”状态机的有关路径，然后“返回”到“PIN输入”状态机上，引起到状态2.2的转移(“第二次PIN输入尝试”)。如果没有按下其他键，则这个状态机仍然留在状态2.2中

西安交大软件学院

## 节点与边覆盖指标

- 由于有限状态机是有向图，因此可以使用与单元级相同的测试覆盖指标
- 层次结构关系指出上层状态机必须把下层状态机作为输入和返回过程处理
- 两种基本的选择是节点覆盖和边覆盖
  - 节点（状态）覆盖类似于单元级的语句覆盖
  - 边（状态转移）覆盖是更可接受的标准。如果状态机“构建良好”（通过端口事件转换），则边覆盖也可保证端口事件覆盖

西安交大软件学院

## 一个线索的节点和边遍历

- 层次结构关系意味着上层状态机必须把下层状态机作为输入和返回过程处理

- 右表以给出三次尝试线索的节点和边覆盖

端口输入事件	端口输出事件	节点	边
(第一次尝试)	屏幕2显示“——”	2. 1	a
按下1	屏幕2显示“X——”	2. 1. 1	x1
按下2	屏幕2显示“XX——”	2. 1. 2	x2
按下3	屏幕2显示“XXX——”	2. 1. 3	x3
按下5 (错误的PIN)	屏幕2显示“XXXX” 显示屏幕3	2. 1. 4 2. 1. 5	x4 X6, 2
(第二次尝试)	屏幕2显示“——”	2. 2	2
按下1	屏幕2显示“X——”	2. 2. 1	x1
按下2	屏幕2显示“XX——”	2. 2. 2	x2
按下3	屏幕2显示“XXX——”	2. 2. 3	x3
按下取消键 (第二次尝试结束)	显示屏幕3	2. 2. 4 2. 2. 6	x10 X11, 4
(第三次尝试)	屏幕2显示“——”	2. 3	4
按下1	屏幕2显示“X——”	2. 3. 1	x1
按下2	屏幕2显示“XX——”	2. 3. 2	x2
按下3	屏幕2显示“XXX——”	2. 3. 3	x3
按下4 (正确的PIN)	屏幕2显示“XXXX” 显示屏幕5	2. 3. 4 2. 3. 5	x4 x5, 5

## 线索/状态关联

- 节点(状态)覆盖可与单元级的状态覆盖类比
- 在PIN输入例子中，需要较少线索即可得到代码覆盖
- 如果观察表就会看到两个线索(由C1234和123C1C1C引起)遍历两个状态机中的所有状态

输入事件	2.1	2.x.1	2.x.2	2.x.3	2.x.4	2.x.5	2.x.6	2.2	2.3	3	1
1234	X	X	X	X	X	X				X	
12351234	X	X	X	X	X	X		X		X	
C1234	X	X	X	X	X	X	X	X		X	
1C12C1234	X	X	X	X			X	X	X	X	
123C1C1C	X	X	X	X	X		X	X	X		X

## 线索/转换关联

- 边(状态转换)覆盖是更可接受的标准
- 如果状态机是“构建良好”的(通过端口事件转换)，则边覆盖也可以保证端口事件覆盖
- 表中的线索是按结构挑选的，以保证较少重复经过的边(由于按下取消键引起)

输入事件	x 1	x 2	x 3	x 4	x 5	x 6	x 7	x 8	x 9	x1 0	x1 1	1	2	3	4	5	6
1234	X	X	X	X	X							X					
12351234	X	X	X	X	X	X							X	X			
C1234	X	X	X	X	X		X				X		X	X			
1C12C1234	X	X	X	X	X			X	X		X		X		X	X	
123C1C1C	X	X	X					X		X	X		X		X		X

## ■ 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

西安交大软件学院

## ■ 线索测试的功能策略

- 如果一个系统没有行为模型（即FSMs），怎么办？
  - 开发行为模型
  - 在系统级上进行功能性测试类比
- 在标识功能性测试用例时，我们使用了输入和输出空间信息，以及功能本身
- 采用通过三个基本概念导出的覆盖指标描述功能线索
  - 事件、端口和数据

西安交大软件学院

## ■ 基于事件的线索测试

### ■ 5个端口输入事件覆盖指标：

- PI1：每个端口输入事件发生
- PI2：端口输入事件的常见序列发生
- PI3：每个端口输入事件在所有“相关”数据语境中发生
- PI4：对于给定语境，所有“不合适”的输入事件发生
- PI5：对于给定语境，所有可能的输入事件发生
- PI1是最低限度指标，大多数系统都做得到
- PI2覆盖是最常见的，对应于系统测试的直观观点，但难以量化
- 后三种指标采用“语境”定义，语境可看做是事件静止点
  - 如：在SATM系统中，屏幕显示发生在事件静止点上

!仅限个人使用\*请勿上传至互联网\*违者必究! 西安交大软件学院

## ■ 基于事件的线索测试

- PI3：每个端口输入事件在所有“相关”数据语境中发生
  - PI3指标处理与语境有关的端口输入事件。这些是物理输入事件，具有由发生这些事件语境确定的逻辑含义
  - 例如：在SATM系统中，按下B1功能键在五种单独的语境中发生(屏幕显示)，并具有三种不同的含义
  - 这种指标的关键是，事件要在所有其语境中驱动

西安交大软件学院

Downloader: 王博玉



## ■ 基于事件的线索测试

- PI4: 对于给定语境, 所有“不合适”的输入事件发生
- PI5: 对于给定语境, 所有可能的输入事件发生
  - PI4和PI5指标常常被试图分解系统的测试人员用做一种非形式化的基础
  - 在给定语境中, 要提供没有预料到的输入事件看看会出现什么情况
  - 例如: 在SATM系统中, 在"PIN输入"过程中按下功能键会出现什么情况?合适的事件是按下数字和取消键, 不合适的输入事件是按下B1、B2和B3键
  - 这有一部分属于规格说明问题:我们在讨论规定行为(应该发生的事)和禁止行为(不应该发生的事)之间的差别
  - 大多数需求规格说明只尽力描述规定行为, 通常由测试人员发现禁止行为

西安交大软件学院

## ■ 基于事件的线索测试

- PI4: 对于给定语境, 所有“不合适”的输入事件发生
- PI5: 对于给定语境, 所有可能的输入事件发生
  - PI4和PI5指标通常非常有效, 但是会产生一些很奇怪的困难:
    - 测试人员怎么知道对于禁止行为输入的预期响应是什么呢?
    - 系统只是忽略这些输入吗?
    - 是否应该输出警告消息?
    - 通常这些问题的答案要靠测试人员的直觉确定
    - 如果时间允许, 这是对需求规格说明进行反馈的很好时机

西安交大软件学院

## ■ 基于事件的线索测试

- 2个端口输出事件覆盖指标:
  - PO1: 每个端口输出事件发生
  - PO2: 每个端口输出事件在每种原因下发生
- PO1覆盖是可接受的最低限度, 对于有大量错误条件输出消息的系统尤其有效
- PO2覆盖是一种好的目标, 但是很难定量描述
- PO2覆盖是指与端口输出事件有关的交互线索, 通常对于给定输出事件, 只有少量原因
- 例如: 在SATM系统中, 有三种原因会显示屏幕10: 1) 终端储备的现金可能用光, 2) 可能要连接到中央银行获得账户余额, 3) 取款通道可能阻塞

!仅限个人使用\*请勿上传至互联网\*违者必究!

西安交大软件学院

## ■ 基于端口的线索测试

- 是基于事件测试的有用补充
  - 通过基于端口的测试, 对于每个端口都要询问端口上会出现什么事件
  - 根据每个端口的事件列表寻找使用输入端口和输出端口的线索
- 对于端口设备来自外部提供商的系统特别有用
- 从E/R模型可看出, 设备和事件之间的多对多测试应该在两个方向上进行
- 基于事件的测试 覆盖 从事件到端口的一对多关系, 反之, 基于端口的测试 覆盖 从端口到事件的一对多关系

西安交大软件学院

Downloader: 王博玉

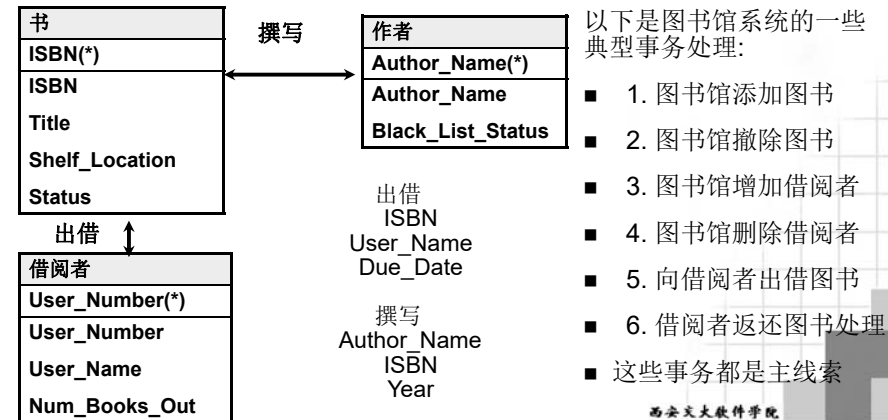
## ■ 基于数据的线索测试

- 基于端口和基于事件的测试适合于主要以事件驱动的“反应式”系统
  - 因为这些系统要对激励（端口输入事件）做出响应，并且反应常常以端口输出事件的形式做出
  - 反应式系统的两个特征：长时间运行、维持与环境的关系
- 一般地，事件驱动的反应式系统没有有意思的数据模型，因此基于数据模型的线索不是特别有用

西安交大软件学院

## ■ 基于数据的线索测试

- 数据驱动的系统称为“静态”的系统，是转换式系统，支持以数据库为基础的事务处理
- 一般采用E/R 模型描述，因此具有大量系统测试线索来源



## ■ 基于数据的线索测试

- 通过基于数据的覆盖指标来描述线索集合
- 这些指标与关系关联
- 关系中的信息一般包含在系统级线索中，而实体中的线索一般局限于单元级
- DM1: 检查每个关系的基数/势(cardinality)
- DM2: 检查每个关系的参与(participation)
- DM3: 检查关系之间的功能(functional)依赖关系

西安交大软件学院

## ■ 基于数据的线索测试

- DM1: 检查每个关系的基数
- 基数是指关系的四种可能性：一对一、一对多、多对一和多对多
- 例如：在图书馆例子中，借出和撰写关系是多对多关系，说明一位作者可以撰写多本书，并且一本书可以有多位作者；一本书可以借给很多借阅者(顺序借出)，一位借阅者可以借阅多本书
- 这些可能性都会产生有用的系统测试线索

西安交大软件学院

## ■ 基于数据的线索测试

- DM2: 检查每个关系的参与
- 参与是指实体的每个实例是否参与到关系中
- 例如：在撰写关系中，Book (书)和Author(作者)实体都要强制参与(不能有没有作者的图书，也不能没有图书作品的作者)
- 在有些建模手段中，用数字限额表示参与。例如，作者实体可以表示为"最少1位最多12位"。
  - 如果这种信息可用，则可以直接产生很明显的边界值系统测试线索

西安交大软件学院

## ■ 基于数据的线索测试

- DM3: 检查每个关系之间的功能依赖关系
- 有时事务处理确定关系之间的显式逻辑联系，这种逻辑联系叫做功能依赖关系
- 例如：不能借出不属于图书馆的图书、也不能删除已经外借的图书、也不能删除手头还有图书的借阅者
- 如果数据库被规范化，则这类依赖关系会减少，但是依然存在，并且可以用来产生有意思的系统测试线索

西安交大软件学院

## ■ 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

!仅限个人使用! 请勿上传至互联网\*违者必究!

西安交大软件学院

## ■ SATM测试线索

- 通过一种全状态模型开发贯穿系统级测试的一组线索，状态表示关键原子系统功能
- 宏级状态：ATM卡输入、PIN输入、事务处理请求、（处理）、会话管理
- 状态的顺序就是测试的顺序，因为这些阶段是以前提顺序给出的
- 需要一些定义带有PAN、预期PIN和账户余额的实际账户前提数据，如下表

SATM测试数据

PAN	预期PIN	支票余额（美元）	储蓄余额（美元）
100	1234	1000.00	800.00
200	4567	100.00	90.00
300	6789	25.00	20.00

Downloader: 王博玉

西安交大软件学院

## SATM测试线索

- 采用表格的形式描述线索，一对行 对应 四个主要阶段中的每一个阶段的端口输入和预期端口输出
- 先讨论三个基本线索，每种线索对应一种事务处理类型(余额查询、存款和取款)

### 线索1：查询支票账号余额

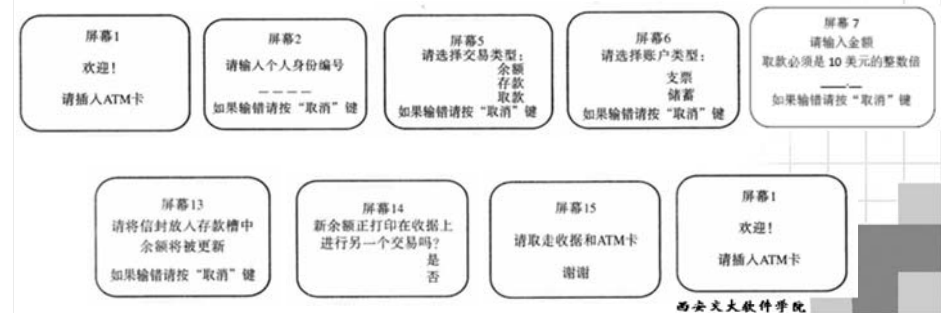
线索1（余额）	ATM卡输入(PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B1, B1	B2
端口输出	屏幕2	屏幕5	屏幕6, 屏幕14, 1000.00美元	屏幕15, 退出ATM卡, 屏幕1



## SATM测试线索

### 线索2：支票账号存款

线索2（存款）	ATM卡输入(PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B2, B1, 25.00	B2
端口输出	屏幕2	屏幕5	屏幕6, 屏幕7, 屏幕13, 存款通道打开, 屏幕14, 1025.00美元	屏幕15, 退出ATM卡, 屏幕1



西安文大软件学院

## SATM测试线索

### 线索3：从储蓄账号中取款

线索3（取款）	ATM卡输入(PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B3, B2, 30.00	B2
端口输出	屏幕2	屏幕5	屏幕6, 屏幕7, 屏幕11, 取款通道打开, 三张10美元钞票, 屏幕14, 770.00美元	屏幕15, 退出ATM卡, 屏幕1



## SATM测试线索

- 以下线索通过作为测试线索目标的输入和输出事件描述
- 线索4是SATM系统中最短的线索，包含无效的ATM卡，立即被退出。

线索4	ATM卡输入(PAN)	PIN输入	事务处理请求	会话管理
端口输入	400			
端口输出	退出ATM卡, 屏幕1			

西安文大软件学院

## SATM测试线索

- 继续给出线索1的宏状态，下面进行各种"PIN输入"。通过产生"PIN输入"有限状态机的边覆盖的表，可得到四个新线索

线索5 (余额)	ATM卡输入 (PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	12351234	与线索1相同	
端口输出	屏幕2	屏幕3, 2, 5		
线索6 (余额)	ATM卡输入 (PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	C1234	与线索1相同	
端口输出	屏幕2	屏幕3, 2, 5		
线索7 (余额)	ATM卡输入 (PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1C12C1234	与线索1相同	
端口输出	屏幕2	屏幕3, 2, 3, 2, 5		
线索8 (余额)	ATM卡输入 (PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	123C1C1C		
端口输出	屏幕2	屏幕3, 2, 3, 2, 4, 1		

## SATM测试线索

- 转到"事务处理请求"阶段，根据事务处理的类型(余额、存款或取款)、账户(支票或储蓄)
- 以下重点讨论金额驱动的线索：
- 线索9拒绝提取非10美元整数倍现金的尝试

线索9 (提取)	ATM卡输入 ( PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B3, B2, 15.00, 取消	B2
端口输出	屏幕2	屏幕5	屏幕6, 7, 9, 7	屏幕15, 退出ATM卡, 屏幕1

西安交大软件学院

## SATM测试线索

- 线索10拒绝提取现金数额多于账户余额的尝试
- 线索11拒绝提取现金数额大于给付器现有现金的尝试

线索10(提取)	ATM卡输入 ( PAN)	PIN输入	事务处理请求	会话管理
端口输入	300	6789	B3, B2, 50.00, 取消	B2
端口输出	屏幕2	屏幕5	屏幕6, 7, 8	屏幕15, 退出ATM卡, 屏幕1

线索11(提取)	ATM卡输入 ( PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B3, B2, 510.00, 取消	B2
端口输出	屏幕2	屏幕5	屏幕6, 7, 10	屏幕15, 退出ATM卡, 屏幕1

## SATM测试线索

- 会话管理阶段，要测试多事务处理部分

线索12(余额)	ATM卡输入 ( PAN)	PIN输入	事务处理请求	会话管理
端口输入	100	1234	B1, B1	B1, 取消
端口输出	屏幕2	屏幕5	屏幕6, 屏幕14, 1000.00美元	屏幕5, 屏幕15, 退出ATM卡, 屏幕1

- 到此为止，线索提供对除屏幕12之外的所有输出屏幕的覆盖，屏幕12告诉用户存款不能被处理
- 这是一种由硬件失效为前提选择的线索，如：将报纸等塞入存款信封中
- 这里给出线索的名称：线索13

西安交大软件学院

## SATM测试线索

与语境有关的输入事件线索

- 开发线索 14-22, 检验与语境有关的输入事件, 如表所示
- 请注意, 头13个线索中的一些也是与语境有关的
- 这22个线索构成了SATM系统的部分合理测试

线索	按下的键	屏幕(语境)	逻辑含义
6	取消	2	PIN输入错误
14	取消	5	事务处理选择错误
15	取消	6	账户选择错误
16	取消	7	金额选择错误
17	取消	8	金额选择错误
18	取消	13	取款信封未就绪
1	B1	5	余额
1	B1	6	支票
19	B1	10	是(非取款事务处理)
20	B1	12	是(非存款事务处理)
12	B1	14	是(另一个事务处理)
2	B2	5	存款
3	B2	6	储蓄
21	B2	10	否(没有其他事务处理)
22	B2	12	否(没有其他事务处理)
1	B2	14	否(没有其他事务处理)

线索	按下的键	屏幕(语境)	逻辑含义
6	取消	2	PIN输入错误
14	取消	5	事务处理选择错误
15	取消	6	账户选择错误
16	取消	7	金额选择错误
17	取消	8	金额选择错误
18	取消	13	取款信封未就绪
1	B1	5	余额
1	B1	6	支票
19	B1	10	是(非取款事务处理)
20	B1	12	是(非存款事务处理)
12	B1	14	是(另一个事务处理)
2	B2	5	存款
3	B2	6	储蓄
21	B2	10	否(没有其他事务处理)
22	B2	12	否(没有其他事务处理)
1	B2	14	否(没有其他事务处理)

西安交大软件学院

## SATM测试线索

- 当然, 有些方面还没有测试, 比如账户余额
- 考虑两个线索, 一个在账户中存入40美元, 另一个取出80美元, 并假设"ATM卡输入"阶段从中央银行得到的余额是50美元
- 有两种可能:1) 使用中央银行的余额, 记录所有事务处理, 然后再每日向中央银行汇总所有处理; 2) 维护本地流转余额, 即在请求事务中要显示的余额
- 如果使用中央银行的余额, 则应该拒绝上述取款尝试
- 如果使用本地余额, 则可以取款
- 这种细节在规格说明中没有描述

## SATM测试线索

- SATM系统另一种明显未测试的部分, 是屏幕7和8出现的"金额输入"过程
- 假设输入40美元。我们期望对于每个数字键入都有一次回显, 但是回显出现在什么位置呢? 显然有两种解决方案:
  - 永远要求输入6位数字(因此应该输入"004000")
  - 先显示高位数字, 当输入后续数字时向左移位, 如图所示
- 大多数ATM系统都使用移位方法, 这会产生微妙问题: ATM系统如何知道所有金额数字都已经输入了呢?
- ATM系统显然不能预测存款金额是40美元, 而不是400或4000美元, 因为当最后数字被输入后, 没有"输入"键可以用来指示输入结束
- 指出这种问题, 是因为这是测试人员发现需求规格说明常常疏漏的细节问题的好例子

输入 4	屏幕 7 请输入金额 取款必须是 10 美元的整数倍 ____.4 如果输错请按“取消”键
输入 0	屏幕 7 请输入金额 取款必须是 10 美元的整数倍 ____.40 如果输错请按“取消”键
输入 0	屏幕 7 请输入金额 取款必须是 10 美元的整数倍 ____.400 如果输错请按“取消”键
输入 0	屏幕 7 请输入金额 取款必须是 10 美元的整数倍 ____.40.00 如果输错请按“取消”键

图 14-9 向左移位数字回显

## ■ 主要内容

- 线索
- 需求规格说明的基本概念
- 基于模型的线索
- 基于用例的线索
- 线索测试的结构策略
- 线索测试的功能策略
- SATM测试线索
- 系统测试指导方针

西安交大软件学院

## ■ 伪结构系统测试

- 本章定义了两个基于图的指标(节点和边覆盖)，将基于图的指标用做功能线索的一种交叉检查
- 这是伪结构系统测试，因为节点和边覆盖指标都以系统的控制模型定义，不是直接从系统实现中导出的
- 一般来说，行为模型只是系统实际情况的近似，这也是为什么我们可以将模型多层细分的原因
- 如果建立一种真正的结构模型，那么这样模型会太大、太复杂、很难使用
- 伪结构指标的一个很大的弱点是：下层模型可能是差的选择

西安交大软件学院

## ■ 伪结构系统测试

- 三种最常见的行为模型（决策表、有限状态机和Petri网）分别适合转换式、交互式 and 并发系统
- 决策表和有限状态机是原子系统功能性测试的很好选择。如果原子系统功能采用决策表描述，则条件一般会包括端口输入事件，行动就是端口输出事件。然后可以设计覆盖每个条件、每个行动、甚至每个规则的测试用例
- 基于有限状态机模型，测试用例可以覆盖每个状态、每种转换或每条路径
- 基于决策表的线索测试是很麻烦的。可以根据不同决策表的规则顺序来描述线索，但是这样做很难追踪覆盖率
- 有限状态机是最基本的选择，如果存在某种形式的交互，则Petri网是最佳选择。这时可以设计覆盖所有位置(place)、每个迁移(transition)、所有迁移序列的线索测试

!仅限个人使用\*请勿上传至互联网\*违者必究!

西安交大软件学院

## ■ 运行剖面

- 齐夫定律 (Zipf's Law): 80%的活动发生在20%的空间中
- 齐夫定律对于测试人员最有用的解释是，空间由所有可能的线索组成，活动是线索的执行(或遍历)。因此，对于有很多线索的系统，80%的执行只遍历20%的线索
- 测试的整体思想：执行测试用例，使得当失效发生时，可发现缺陷的存在
- 如果仅有的缺陷只“局限”在很少遍历的线索中，则与同样数量缺陷存在于“经常使用”的线索中相比，整体可靠性较高
- 运行剖面的思想：确定各种线索的执行频率，并使用这些信息为系统测试选择线索
- 确定系统运行剖面的一种方法是使用决策树，对于任何状态，找出(或估计) 每种状态向外转移的概率(各种转移概率的和必须是1)

西安交大软件学院

Downloader: 王博玉

## 运行剖面示例

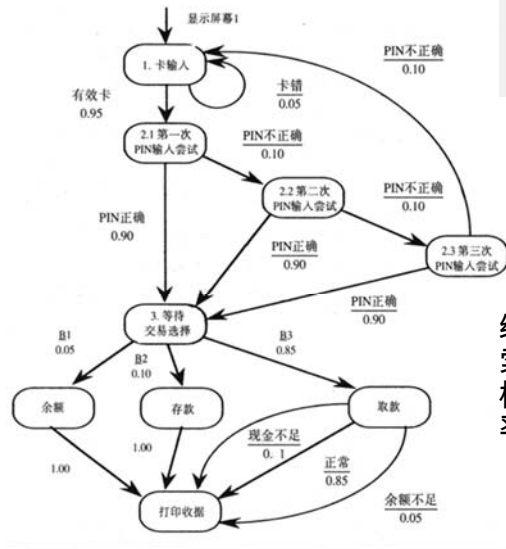


图14-10 SATM系统的转移概率

■图中给出了假想转移概率的结果

■给定转移概率，线索的总概率就是线索各转移概率的积

■表给出了最高和最低频率线索的计算结果

常见线索	概率%
有效ATM卡	0.95
PIN第一次尝试正确	0.90
取款	0.85
正常	0.85
总概率	0.618
罕见线索	概率%
有效ATM卡	0.95
PIN第一次尝试错误	0.10
PIN第二次尝试错误	0.10
PIN第三次尝试正确	0.90
取款	0.85
现金不足	0.10
总概率	0.000727

线索概率

## 累进测试与回归测试

■对于迭代式、增量开发

■当构建2加到构建1上时，要测试构建2中的新内容，还要重新测试构建1，以确定新内容不会对构建1的内容产生有害的影响

■如果项目有多个构建，回归测试意味着测试的大量重复

■对于累进测试，测试的是新内容，因此预期失效率要比回归测试的失效率高

西安交大软件学院

## 总结

- 原子系统功能（ASF）是什么？
- 如何寻找线索？
- 线索测试的结构策略有哪些？
- 线索测试的功能策略有哪些？

## 练习

- 对于NextDate，进行ASF测试

```

1  Main integrationNextDate      'start program event occurs here
   Type   Date
         Month As Integer
         Day As Integer
         Year As Integer
   EndType
   Dim today As Date
   Dim tomorrow As Date
   Output("Welcome to NextDate!")
   2  GetDate(today)              'msg1
   3  PrintDate(today)            'msg2
   4  tomorrow = IncrementDate(today) 'msg3
   5  PrintDate(tomorrow)         'msg4
   6  End Main

8  Function isLeap(year) Boolean
9  If (year divisible by 4)
10 Then
11     If (year is NOT divisible by 100)
12     Then isLeap = True
13     Else
14         If (year is divisible by 400)
15         Then isLeap = True
16         Else isLeap = False
17     Endif
18 Endif
19 Else isLeap = False
20 Endif
21 End (Function isLeap)

22 Function lastDayOfMonth(month, year) Integer
23 Case month Of
24     Case 1: 1, 3, 5, 7, 8, 10, 12
25         lastDayOfMonth = 31
26     Case 2: 4, 6, 9, 11
27         lastDayOfMonth = 30
28     Case 3: 2

```

Downloader: 王博玉



```

29     If (IsLeap(year)) 'msg5
30         Then lastDayOfMonth = 29
31         Else lastDayOfMonth = 28
32     EndIf
33 EndCase
34 End (Function lastDayOfMonth)

35 Function GetDate(aDate) Date
    dim aDate As Date

36 Function ValidDate(aDate) Boolean 'within scope of GetDate
    dim aDate As Date
    dim dayOK, monthOK, yearOK As Boolean
37     If ((aDate.Month > 0) AND (aDate.Month <= 12))
38         Then monthOK = True
39         Output("Month OK")
40     Else monthOK = False
41         Output("Month out of range")
42     EndIf
43     If (monthOK)
44         Then
45             If ((aDate.Day > 0) AND (aDate.Day <=
                lastDayOfMonth(aDate.Month, aDate.Year)) 'msg6
46                 Then dayOK = True
47                 Output("Day OK")
48             Else dayOK = False
49                 Output("Day out of range")
50             EndIf
51         EndIf
52     If ((aDate.Year > 1811) AND (aDate.Year <= 2012))
53         Then yearOK = True
54         Output("Year OK")
55     Else yearOK = False
56         Output("Year out of range")
57     EndIf
58     If (monthOK AND dayOK AND yearOK)
59         Then ValidDate = True
60         Output("Date OK")
61     Else ValidDate = False
62         Output("Please enter a valid date")
63     EndIf
64 End (Function ValidDate)

```

```

'GetDate body begins here
65 Do
66     Output("enter a month")
67     Input(aDate.Month)
68     Output("enter a day")
69     Input(aDate.Day)
70     Output("enter a year")
71     Input(aDate.Year)
72     GetDate.Month = aDate.Month
73     GetDate.Day = aDate.Day
74     GetDate.Year = aDate.Year
75 Until (ValidDate(aDate)) 'mag7
76 End (Function GetDate)

77 Function IncrementDate(aDate) Date
78     If (aDate.Day < lastDayOfMonth(aDate.Month)) 'msg8
79         Then aDate.Day = aDate.Day + 1
80     Else aDate.Day = 1
81         If (aDate.Month = 12)
82             Then aDate.Month = 1
83             aDate.Year = aDate.Year + 1
84         Else aDate.Month = aDate.Month + 1
85         EndIf
86     EndIf
87 End (IncrementDate)

88 Procedure PrintDate(aDate)
89     Output("Day is ", aDate.Month, "/", aDate.Day, "/", aDate.Year)
90 End (PrintDate)

```

西安交大软件学院

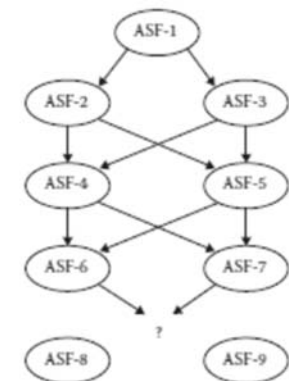
## 找出输入输出事件

Input Events	Node	Output Event Description	Node
e0: Start program event	1	e7: Welcome message	2
e1: Enter a valid month	67	e8: Print today's date	4
e2: Enter an invalid month	67	e9: Print tomorrow's date	6
e3: Enter a valid day	69	e10: "Month OK"	39
e4: Enter an invalid day	69	e11: "Month out of range"	41
e5: Enter a valid year	71	e12: "Day OK"	47
e6: Enter an invalid year	71	e13: "Day out of range"	49
		e14: "Year OK"	54
		e15: "Year out of range"	56
		e16: "Date OK"	60
		e17: "Please enter a valid date"	62
		e18: "Enter a month"	66
		e19: "Enter a day"	68
		e20: "Enter a year"	70
		e21: "Day is month, day, year"	89

## 找出ASF – 第一次尝试

Table 14.16 First Attempt at ASFs

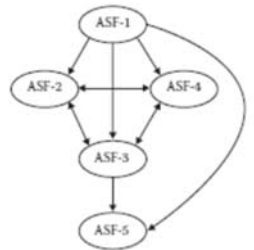
Atomic System Function	Inputs	Outputs
ASF-1: start program	e0	e7
ASF-2: enter a valid month	e1	e10
ASF-3: enter an invalid month	e2	e11
ASF-4: enter a valid day	e3	e12
ASF-5: enter an invalid day	e4	e13
ASF-6: enter a valid year	e5	e14
ASF-7: enter an invalid year	e6	e15
ASF-8: print for valid input		
ASF-9: print for invalid input		



## 找出ASF - 第二次尝试

Table 14.17 Second Attempt at ASFs

Atomic System Function	Inputs	Outputs
ASF-1: start program	e0	e7
ASF-2: enter a date with an invalid month, valid day, and valid year	e2, e3, e5	e11, e12, e14, e17
ASF-3: enter a date with an invalid day, valid month, and valid year	e1, e4, e5	e10, e13, e14, e17
ASF-4: enter a date with an invalid year, valid day, and valid month	e1, e3, e6	e10, e12, e15, e17
ASF-5: enter a date with valid month, day, and year	e1, e3, e5	e10, e12, e14, e16, e21
ASF-6: enter a date with valid month, invalid day, and invalid year	e1, e4, e6	e10, e13, e15, e17
ASF-7: enter a date with valid day, invalid month, and invalid year	e2, e3, e6	e11, e12, e15, e17
ASF-8: enter a date with valid year, invalid day, and invalid month	e5, e4, e6	e14, e13, e15, e17
ASF-9: enter a date with invalid month, invalid day, and invalid year	e2, e4, e6	e11, e13, e15, e17



西安交大软件学院