

Перспективные техники тестирования крупных систем

ПЕТРЕНКО Александр Константинович

petrenko@ispras.ru

<http://ispras.ru/~petrenko>

ICAS-2011, 23 ноября 2011 года

Ориентиры развития технологии или Критерии оценки эффективности

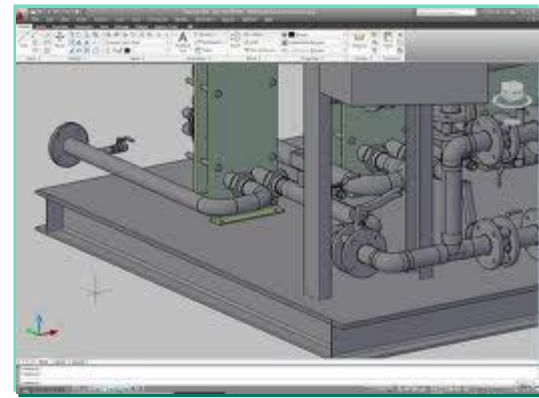
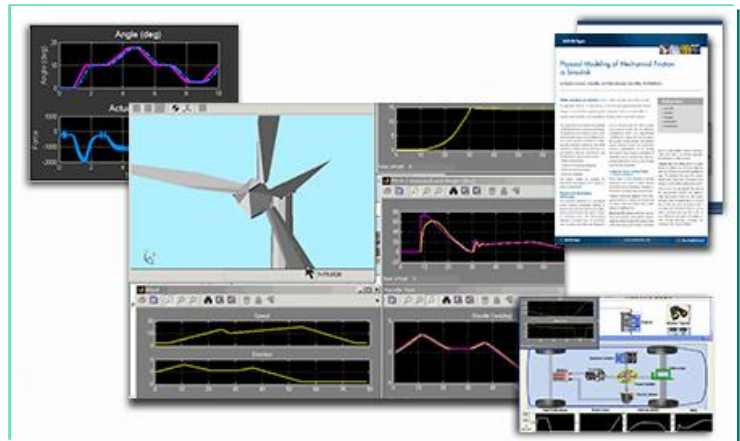
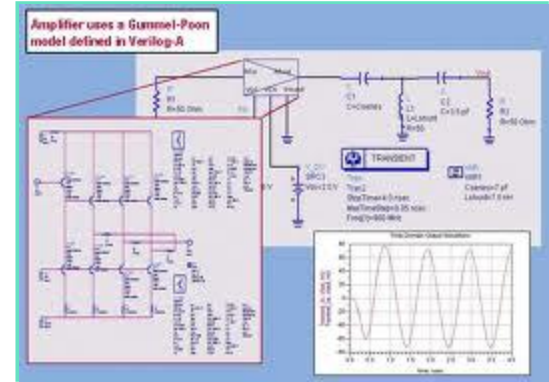
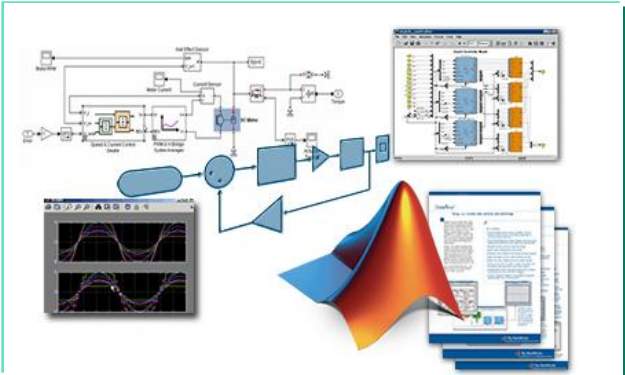
- Плохой ориентир
 - Повысить время наработки на отказ
 - Сократить расходы сопровождение (тестирование) информационной системы
- Хороший ориентир
 - Повысить уровень переиспользования компонентов в системе тестирования и поддержки
 - Внедрить систему измерения качества/надежности информационной системы

Перспективные техники

- Тестирование на основе моделей
(Model Based Testing – MBT)
- Метрики для измерения
качества/адекватности тестирования

Тестирование на основе моделей

Традиционное понимание моделей



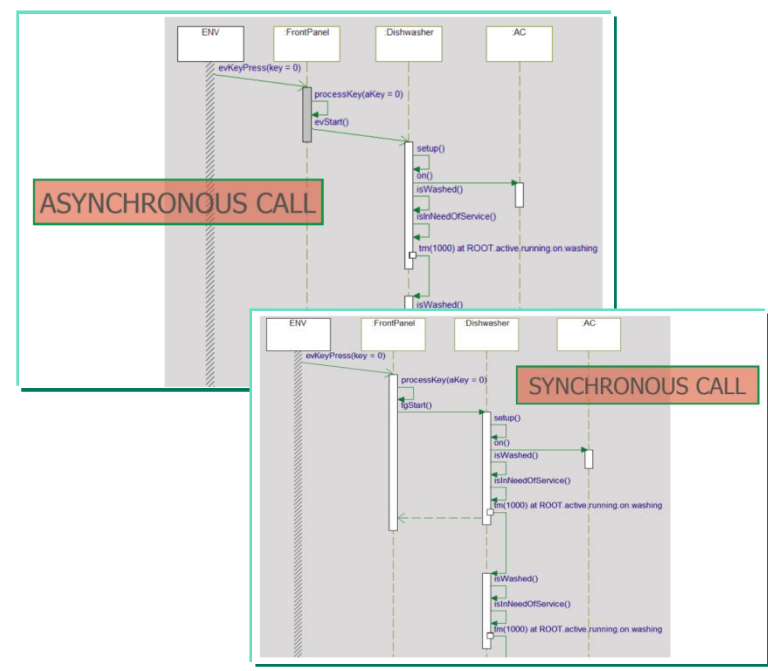
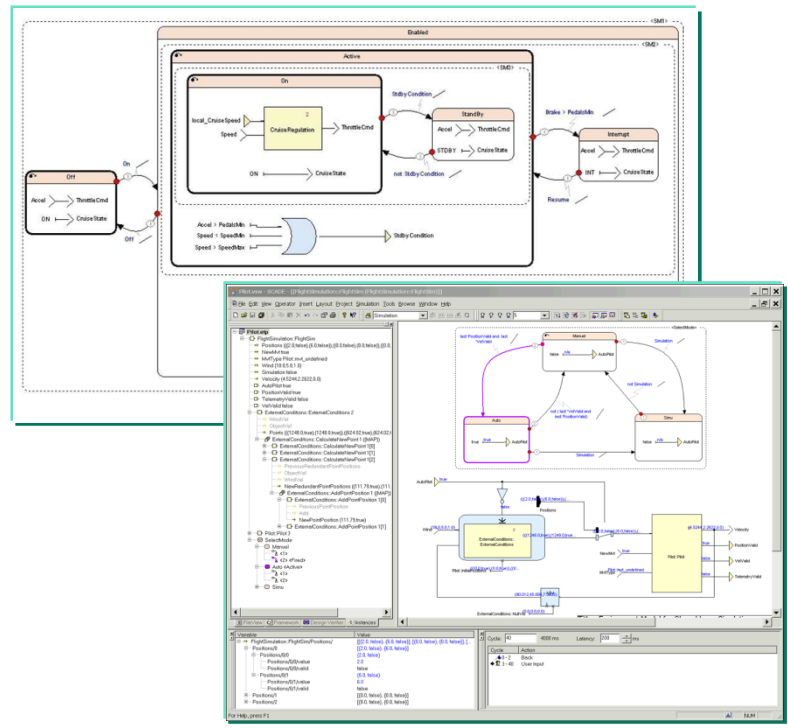
Тестирование программ на основе моделей

- Объекты моделирования
 - Модель реализации целевой системы
 - Модель требований к системе
 - Модель сценариев использования
 - + модели «атак»
 - + модели отказов
 - Интегрированная модель бизнес процессов и целевая система

Примеры моделей программного обеспечения встроенных систем управления

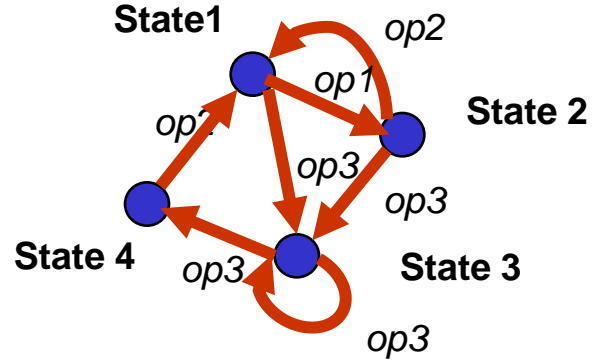
SCADE (Esterel Technologies)

Диаграмма последовательностей
(IBM/Rhapsody)



Виды моделей программ и НОТАЦИИ

FSM /
FA



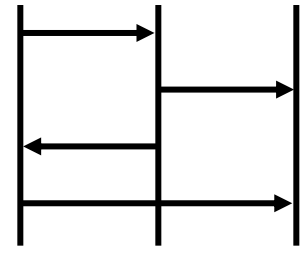
Грамматика

```
<assign> ::= <id> “:=” <expr>
<expr> ::= <intexp> | <boolexp>
```

Программный контракт

```
class ArrayList { public virtual void
  Insert(int index , object value)
  requires 0 <= index && index <= Count,
  requires !IsReadOnly && !IsFixedSize;
  ensures Count == old(Count) + 1;
  ensures value == this[index];
  ensures forall{int i in 0 : index ; old(this[i])
    == this[i]};
  ensures forall{int i in index : old(Count);
    old(this[i]) == this[i + 1]};
  { ... }
```

MSC



RSL

```
axiom  $\forall s : S, e : \text{Nat} :-$ 
  first(add(s, e))  $\equiv$ 
  if e > 10 then 2 * first(s)
  elseif e > 5 then first(s)
  else 0 end
```

Метрики для измерения качества/адекватности тестирования

- Покрытие кода важно для тестирования компонентов, но не эффективно для анализа надежности системы в целом
- Возможные основы для построения метрик:
 - Структура (иерархия) требований
 - Метод «скользящего окна» при анализе трассы взаимодействия компонентов

UniTESK (ИСПРАН)

Пример спецификации в JavaTESK (программный контракт):

```
specification int withdraw(int sum) {  
pre { return sum > 0; }  
post {   if (balance > 0)  
        mark "Withdrawal from account with positive balance";  
        else if (balance == 0)  
        mark "Withdrawal from empty account";  
        else  
        mark "Withdrawal from account with negative balance";  
if(balance < sum - maximumCredit)  
{           branch TooLargeSum;  
return balance == pre balance && withdraw == 0 ;      }  
else  
{           branch Normal ;  
return balance == pre balance – sum && withdraw == sum ; } } }
```

Forte for Java 4, Community Edition [Project Default] J@T Specification Method Coverage Report - Microsoft Internet Explorer

File Edit View Project Build Debug Versioning Tools File Edit View Favorites Tools Help Address C:\Documents and Settings\Administrator\Local Settings\Temp\jattReport37274\specification-

Editing GUI Editing Running Debugging

Explorer [Filesystems]

Filesystems

- D:\AT\examples\tests
 - ru
 - ispras
 - redverst
 - se
 - java
 - examples
 - account
 - failures
 - model
 - AccountMediator
 - AccountSpecifica
 - AccountTestScer
 - class Account
 - Test Results
 - Mon Aug
 - Mon Aug
 - Mon Aug
 - Mon Aug
 - Thu Aug
 - AccountTestScer
 - AccountTestScer
 - scenario-failures
 - scenario-states
 - scenarios
 - specification-branch
 - specification-disjuncts
 - specification-failures
 - specification-marks
 - specification-predicat
 - specifications
 - AccountTestScenario.1059988829290
 - AccountTestScenario.1059990198649
 - AccountTestScenario.1059990887891
 - index
 - jatt
 - jatt-report
 - intset
 - pqueue
 - queue
 - sqrt

D:\AT\examples\gen

J@T Specification Method Coverage Report
generated: 28.08.2003 15:02:15

Report Overview
[All Failures](#)
[Specifications Coverage](#)
[Failures](#)
[Branches](#)
[Marks](#)
[Predicates](#)
[Disjuncts](#)
[Scenarios Coverage](#)

Packages Overview
[ru.ispras.redverst.se.java.examples.account.model](#)
[AccountSpecification](#)
[deposit\(int\)](#)
[withdraw\(int\)](#)

| withdraw(int) | | | | | default context | total |
|---------------|-----------|------------|-----------|------------|-----------------|-------|
| branches | marks | predicates | disjuncts | hits/fails | hits/fails | |
| 100% (2/2) | 83% (5/6) | 83% (5/6) | 83% (5/6) | 59 | 59 | |

| branches | marks | predicates | disjuncts | | | | | | | default context | total | | | |
|----------------------------|---|------------|-----------|----|----|----|----|----|----|-----------------|-------|----|----|-----|
| | | | f1 | f2 | f3 | f4 | f5 | f6 | f7 | | | f8 | f9 | f10 |
| Successful withdrawal | Withdrawal from account with negative balance; Successful withdrawal | predicate1 | + | - | - | - | * | * | * | * | * | * | 5 | 5 |
| | Withdrawal from empty account; Successful withdrawal | predicate2 | + | + | - | - | * | * | * | * | * | * | 3 | 3 |
| | Withdrawal from account with positive balance; Successful withdrawal | predicate3 | + | + | * | - | * | * | * | * | * | * | 41 | 41 |
| Withdrawn sum is too large | Withdrawal from account with negative balance; Withdrawn sum is too large | predicate4 | + | - | + | * | * | * | * | * | * | * | 9 | 9 |
| | Withdrawal from empty account; Withdrawn sum is too large | predicate5 | + | - | + | * | * | * | * | * | * | * | 1 | 1 |
| | Withdrawal from account with positive balance; Withdrawn sum is too large | predicate6 | + | + | * | * | * | * | * | * | * | * | 0 | 0 |

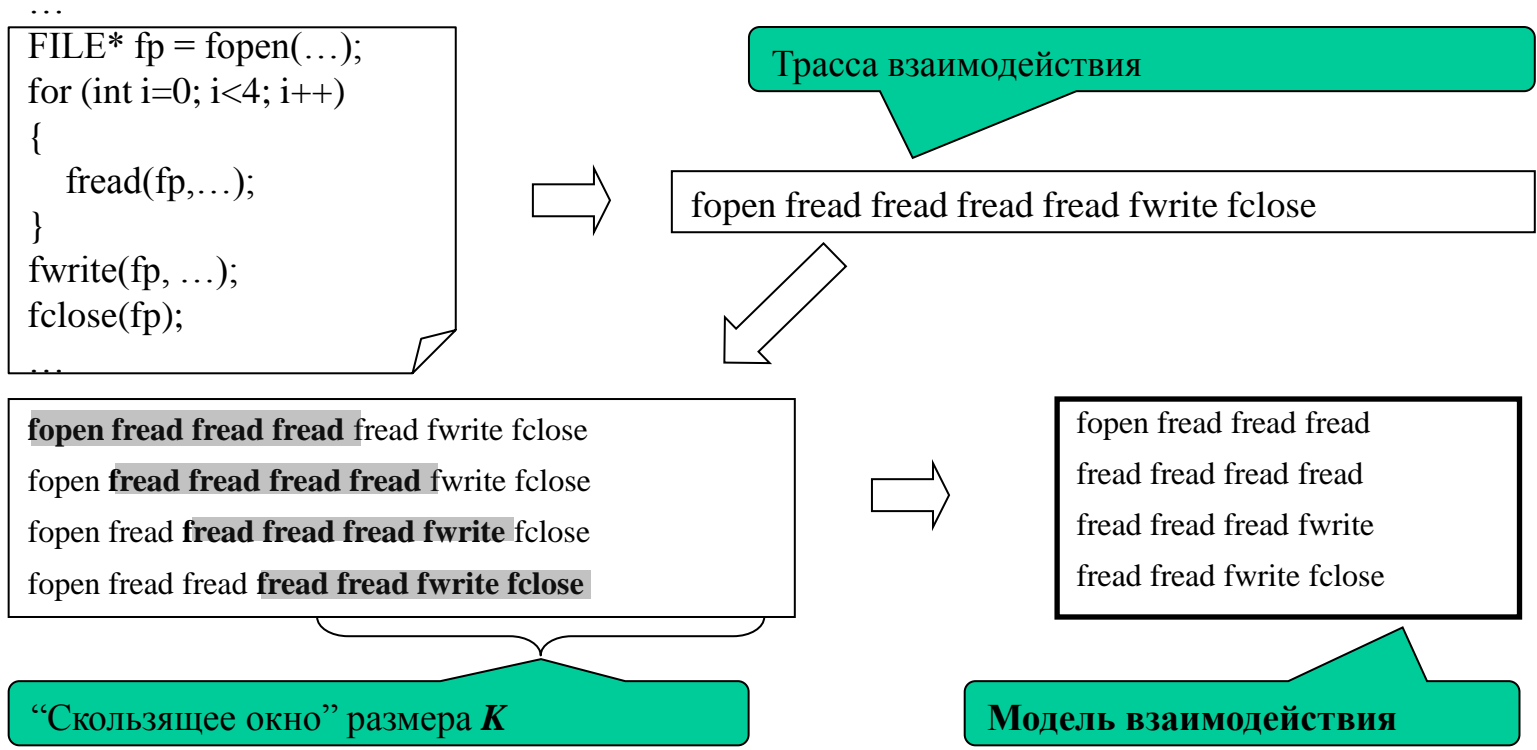

```

specification public int withdraw(int sum)
  reads sum, maximumCredit
  updates balance
{
  pre { return sum > 0; }
  post
  {
    if(balance > 0)
      mark "Withdrawal from account with positive balance";
    else if(balance == 0)
      mark "Withdrawal from empty account";
  }
}
    
```

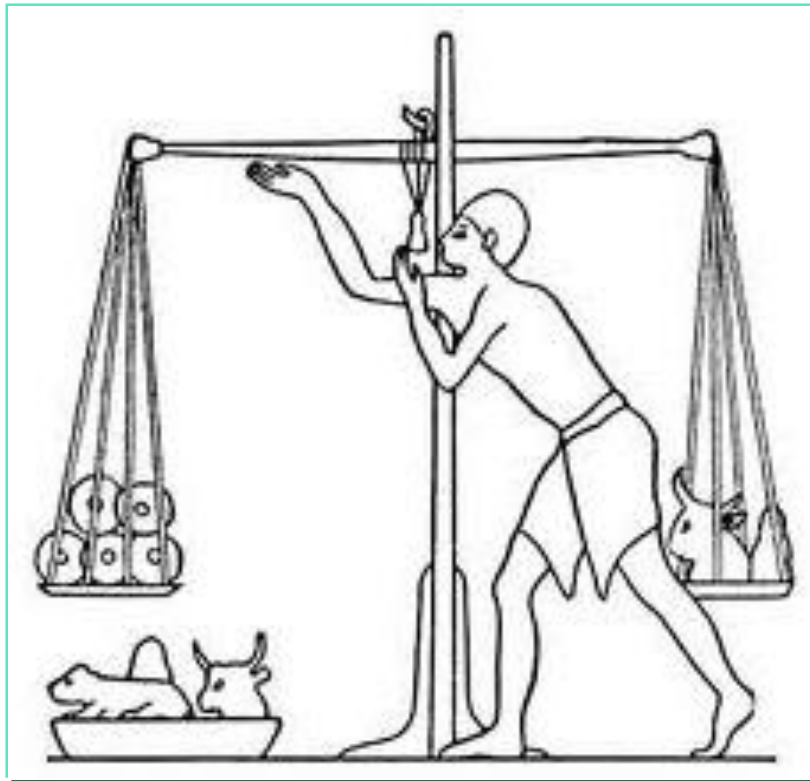
1:1 INS My Computer

Метод «скользящего окна» при анализе трассы взаимодействия компонентов

Модель поведения взаимодействия – множество последовательностей вызовов интерфейсных функций длины K , выполненных во время тестирования взаимодействия.



Соотношение мер технических и организационных



Спасибо!

А.К.Петренко

petrenko@ispras.ru

<http://ispras.ru/~petrenko>