The Florida A&M University Software TestLab: Perspective, Direction and Opportunities

Dr. Edward L. Jones
CIS Department
Florida A&M University
Tallahassee, FL

Part I
INTRODUCTION
My Institution

- 12,000 students total
- 600+ CIS majors
- 15 Faculty
- Heavy teaching
- Graduate program (20 students)
- Increasing research

My Experience

- 13 years Harris Corporation
- Software engineering technology
- $200M+ NASA project
  - methodology expert
  - software inspections, testing
  - process training
- 20+ years university teaching
- 3 years focusing on testing
The Vision

- Quality Awareness in Curriculum
- Solid Framework
- Enhanced Student Skill Set
- Expanded Student Opportunities
- Industry Partnerships
- Research & Dissemination

The Software TestLab

- Environment for discovery & learning
  - Basic testing skills
  - Mentoring / Competency based training
- Evolving Laboratory
  - Tools & tutorials
  - Staffed by students and faculty
  - Problems/solutions test bed
- Dissemination Strategy
My Perspective

- Testing is not just for testers!
- In ideal world, fewer testers required
  - Verification vs testing
  - More skilled developers
- No silver bullet … just bricks
  - Simple things provide leverage
  - Testing *in-the-small* -- classroom the lab
- Technology transfer crucial
SPRAE Testing Framework

- Specification – basis for testing
- Remediation – follow a process
- Repeatability – tester independence
- Accountability – documentation
- Economy – cost effectiveness

Why SPRAE?

- A value system for testing practice
- Explains the “what” and “why”
- Sets goals for test experiences
- Each experience reinforces values
A Test Life Cycle

- Specification (Y/N)
- Analysis
- Design
- Implementation
- Execution
- Evaluation

- Test Plan/Strategy
- Test Script, Data, Driver
- Defect Data Problem Reports
- Test Results, Log
- Test Cases

Part II

EDUCATION MISSION

(What can Academics Do Besides Teaching A Course?)
Caught and Taught

- Caught
  - Attitudes
  - Respect for consequence of not testing
- Taught
  - Techniques for specific kinds of testing
  - Basic analytical and programming skills
- Strategy must contain both elements

A Holistic Approach

SPRAE Testing Framework

Software Test Lab

Elective Testing Course

Core Course Experiences

Testing In Action Experiences
What Is Meant By “Holistic”?  
- Testing an integral part of curriculum 
- Multiple test experiences 
- Experiences in each course 
- Repetition and reinforcement 
- Accumulation of experiences 
- Experiences cover test lifecycle 
- Holistic, NOT Exhaustive!

A Software Testing Course  
- 80% practice, 20% theory 
- 2/3 fine-grained testing (functions) 
- 1/3 object and application testing 
- Test cases, drivers, and scripts 
- Decision tables the "formalism" 
- Function, boundary, white-box testing 
- Effectiveness: Coverage & error seeding
Course -- Lessons Learned

- **Advantages**
  - A "taste" of how hard testing really is
  - Preparation for advanced study
  - Rounds out analytical & programming skills

- **Deficiencies**
  - Course not available to all students
  - Students compartmentalize knowledge

Technology Transfer -- Program Grading Services

- Provided by TestLab students
  - Experience designing automated tests
  - Shell programming

- Instructor plays active part
  - Refines specification
  - Creates grading plan

- **Benefits**
  - Faculty more inclined to incorporate testing “module” into course
Technology Transfer -- Plagiarism Deterrent

- Mutation analysis/error seeding applied to suspicious student program
- Student must debug modified program
- TestLab project – tunable error seeding

Benefits
- Tool for teaching students to debug
- Less plagiarism … or … more skilled programmers

TestLab Student Mentorship Model

- Managed skill development
- Clear achievement goals
- Key Practices x Competency Levels
- Progress certification
- Student-Student mentoring
- Recognition Program
Key Practices

- Practitioner -- performs defined test
- Builder -- constructs test “machinery”
- Designer -- designs test cases
- Analyst -- sets test goals, strategy
- Inspector -- verifies process/results
- Environmentalist -- maintains test tools & environment
- Specialist -- performs test life cycle.

Specialist I - Competencies

| Role                      | 1 | 2 | 3 | 4 | 5 | ...
|---------------------------|---|---|---|---|---|-----
| Test Practitioner         |   | 2 | 3 | 4 | 5 |     
| Test Builder              |   | 2 | 3 | 4 | 5 |     
| Test Designer             |   | 2 | 3 | 4 | 5 |     
| Test Analyst              |   | 2 | 3 | 4 | 5 |     
| Test Inspector            |   | 2 | 3 | 4 | 5 |     
| Test Environmentalist     |   | 2 | 3 | 4 | 5 |     
| Test SPECIALIST           |   | 2 | 3 | 4 | 5 |     

4/25/2002

FAMU TestLab
Problems/Solutions Testbed

- Repository of testing problems
- Repository of student test artifacts
- *Best in class* → solutions testbed
- *Deficient* solutions → "almost" testbed
- Testbed used for tester certification

The Test Arcade

- Fun & Games approach
- Players compete solving testing problem
  - Scored on time and accuracy
  - Ranked list of players by score
  - HELP facility provides a "teaching" mode
- Supports testing contests, certification
- NSF Funding requested
Part III

RESEARCH ACTIVITIES

Seminal Projects

- Decision-based test methods
- Reliability testing training simulator
- Test arcade
- Testing via design mutation
- Test patterns/verification agents
- Other Ideas
Decision Based Testing (DBT)

- Decision Table
  - Logic model, lightweight formality
  - Simple to teach
  - A state-based model
- Systematic test condition generation
  - Column => behavioral equivalence class
  - Conditions => basis for boundary analysis
- Complication
  - DT variables computed from inputs

Reliability Simulation

- Training in reliability modeling
- Published ISSRE-2001; Graduate course
- Simulation for experimentation
  - Imperfect defect removal
  - Operational testing
- Thesis topic(s)
  - Model validation vs. published results
  - Management decision-support via simulation
Test Arcade

**Problem:** Testbed of problem templates + problem/answer generator + presentation + response manager + score manager.

**Approach:** Identify problem templates and develop knowledge base for representation and generation. Target problems to specific skill levels.

**Research:** Refine competency levels x test instances. Accumulate problem set per (level, instance) pairs.

**Tool:** database + web access.

Design Mutation

- Reflects design misinterpretations
- Mutant killed when $\text{Fail}_k$ is nonempty

![Diagram of Design Mutation process]

Design → Mutate → Gen Test Sets → Run Tests → Fail $\text{Fail}_1$, $\text{Fail}_2$, ..., $\text{Fail}_n$
Design Mutation -- Prospectus

- Application
  - Requirements/design models
  - Guided inspection -- search for likely faults, i.e., kill mutants
  - Testing: creation of BAD test cases
- Practical if automated
  - Facilitated by formal representation

Test Patterns (Example)

- GUI input controls
- Properties determine required testing
- Derive black-box test patterns for GUI input controls
- Propose automated test agents knowledgeable about specific test pattern
Test Patterns (Continued)

- Test patterns for design patterns
- Empirical test patterns reflecting organizational practices
  - Reverse engineer practices into patterns
  - Forward engineer patterns into practice

Test Agents

- Properties of Agents
  - Responsibility
  - Knowledge
  - Autonomy
- Scope of Agents
  - Functional Unit (F)
  - Architectural Component (A)
  - System (S)
Test Agents -- Prospectus

- Intelligence
  - Generator of checklist / procedure for human tester
  - Watch-dog and status reporter
  - Reminder to complete test action
  - Performer of test action
  - Coordinator of other test agents

Seminal Projects

- Decision-based test methods
- Reliability testing training simulator
- Test arcade
- Testing via design mutation
- Test patterns / verification agents
- Other Ideas
Study: State of Testing Practice

**Problem:** Develop a method for characterizing the state of practice in software testing.

**Approach:** Use frameworks like the Test Maturity Model and SPRAE to characterize/assess state of practice.

**Research:** Relate attributes of testing practice to qualitative and quantitative indicators of effectiveness and satisfaction. Devise easy to use evaluation system that identifies areas needing improvement, and which provide insightful measures of performance.

Adaptive Operational Testing

**Problem:** Perfecting software by OT is biased by expected feature usage. Even when errors are flushed out for one feature, the test emphasis remains the same. OT leads to slow rate of error discovery in other features.

**Approach:** Given features A, B, C with probabilities \( p_A, p_B, p_C \), and MTBF of \( t_A, t_B, t_C \). Shift feature probability as feature reliability increases.

**Research:** Determine criteria for shifting p’s so that feature starvation does not occur.

**Tool:** Reliability simulator to prototype solutions.
Testing and Reverse Engineering

- Testing answers the questions
  - “What do we have here?” (exploratory)
  - “Is this what we wanted?” (acceptance)
  - “Is this what we expected?” (scripted)
- Testing is the last chance to document the as-built system
- Exploratory testing -- can it be sooner?

Testability via Self-Diagnostics in Objects

- **Problem:** Lack of observability [Voas] in O-O software complicates testability. Encapsulation prevents error propagation to outside.
- **Approach:** Design in self diagnostics along with means of propagation to outside.
- **Research:** Research observability problem in object-based testing. Apply and extend frameworks/methods like JUnit to implement self-diagnostics.
Part IV

CONCLUSION

Support Acknowledgment

- Lucent Technologies
- Dell Computer
- 3M
- Telcordia
- Students
- NSF (EIA-9906590)
- Lockheed Martin
- Abbott Laboratories
- Eli Lilly and Company
- Software Research, Inc.
Future

- Evolve TestLab Mentorship Model
- Transfer to Selected Courses
  - TestLab students = transfer agents
- Disseminate Results
  - Web site, newsletter, conferences
- Procure Funding for Research
- Find Research Collaborators

Opportunities to Partner

- Sponsor Grad/Undergrad Student
- Student Internship
- Faculty Internship
- M.S. Thesis project
- Shadow Projects
- Design study of specific test technique
TestLab Website
www.cis.famu.edu/~testlab

Questions?
Comments?
ejones@cis.famu.edu

Questions?

Questions?

Questions?