Turning the Lecture Course Inside Out

Cem Kaner, J.D., Ph.D.
Colloquium
Florida Institute of Technology
October 7, 2005
What's this course about?

Software testing is a process of technical investigation of the product under test conducted to provide stakeholders with quality-related information.
It's kind of like CSI

MANY tools, procedures, sources of evidence.

- Tools and procedures don't define an investigation or its goals.
- There is too much evidence to test, tools are often expensive, so investigators must exercise judgment.
- The investigator must pick what to study, and how, in order to reveal the most needed information.
Instructional Implications

- Software testing is a cognitively complex task
  - We might or might not consider it **as important** to teach as traditionally difficult science / math courses, e.g., Calculus
  - But it is **as difficult** to teach because the cognitive issues are similar
Complexity? What are We Facing in Testing?

• Many techniques / tools with different strengths and applications.
  – The tester must decide which technique or tool is most useful for the situation at hand.
• Infinite pool of possible tests
  – The tester must prioritize intensely because the vast majority of worthy tasks will not get done
  – Recommended procedures (e.g. IEEE Test Documentation standard 829) cost time and must be part of the tester's tradeoff. Spending time on X (and there are many testers-should-always-X's) means that time is unavailable for Y.
Some Other Testing Challenges

• The oracle problem
  – How do you decide whether the program passed or failed the test?
    • You can **assume this away** by assuming the existence of **complete, authoritative specifications**. But **assuming a problem away doesn't make it go away**. (Though it might make discussion of the problem politically incorrect.)

• The measurement problem
  – Broad disagreement about how to measure coverage, testing progress, tester productivity, etc.

• The agency problem
  – **Who speaks (accurately) for the customer and the other stakeholders? Are testers authorized, knowledgeable and appropriate for this?**
Bloom's taxonomy (1956) is the classic attempt to characterize cognitive levels of learning:

- Knowledge
- Comprehension
- Application
- Analysis
- Synthesis
- Evaluation
### Bloom's Ranking of Thinking Skills

<table>
<thead>
<tr>
<th>Knowledge</th>
<th>Knowledge</th>
<th>Comprehension</th>
<th>Comprehension</th>
<th>Application</th>
<th>Application</th>
<th>Analysis</th>
<th>Analysis</th>
<th>Synthesis</th>
<th>Synthesis</th>
<th>Evaluation</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>List</td>
<td>List</td>
<td>Summarize</td>
<td>Summarize</td>
<td>Solve</td>
<td>Solve</td>
<td>Analyze</td>
<td>Analyze</td>
<td>Design</td>
<td>Design</td>
<td>Evaluate</td>
<td>Evaluate</td>
</tr>
<tr>
<td>Name</td>
<td>Name</td>
<td>Explain</td>
<td>Explain</td>
<td>Illustrate</td>
<td>Illustrate</td>
<td>Organize</td>
<td>Organize</td>
<td>Hypothesize</td>
<td>Hypothesize</td>
<td>Choose</td>
<td>Choose</td>
</tr>
<tr>
<td>Identify</td>
<td>Identify</td>
<td>Interpret</td>
<td>Interpret</td>
<td>Calculate</td>
<td>Calculate</td>
<td>Deduce</td>
<td>Deduce</td>
<td>Support</td>
<td>Support</td>
<td>Estimate</td>
<td>Estimate</td>
</tr>
<tr>
<td>Show</td>
<td>Show</td>
<td>Describe</td>
<td>Describe</td>
<td>Use</td>
<td>Use</td>
<td>Contrast</td>
<td>Contrast</td>
<td>Schematize</td>
<td>Schematize</td>
<td>Judge</td>
<td>Judge</td>
</tr>
<tr>
<td>Define</td>
<td>Define</td>
<td>Compare</td>
<td>Compare</td>
<td>Interpret</td>
<td>Interpret</td>
<td>Compare</td>
<td>Compare</td>
<td>Write</td>
<td>Write</td>
<td>Defend</td>
<td>Defend</td>
</tr>
<tr>
<td>Recognize</td>
<td>Recognize</td>
<td>Paraphrase</td>
<td>Paraphrase</td>
<td>Relate</td>
<td>Relate</td>
<td>Distinguish</td>
<td>Distinguish</td>
<td>Report</td>
<td>Report</td>
<td>Justify</td>
<td>Justify</td>
</tr>
<tr>
<td>Recall</td>
<td>Recall</td>
<td>Differentiate</td>
<td>Differentiate</td>
<td>Manipulate</td>
<td>Manipulate</td>
<td>Discuss</td>
<td>Discuss</td>
<td>Justify</td>
<td>Justify</td>
<td></td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>State</td>
<td>Demonstrate</td>
<td>Demonstrate</td>
<td>Apply</td>
<td>Apply</td>
<td>Plan</td>
<td>Plan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Visualize</td>
<td>Visualize</td>
<td>Classify</td>
<td>Classify</td>
<td>Modify</td>
<td>Modify</td>
<td>Devise</td>
<td>Devise</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[http://www.umuc.edu/ugp/ewp/bloomtax.html](http://www.umuc.edu/ugp/ewp/bloomtax.html)

(c) Cem Kaner 2006
Characterizing Cognitive Complexity

- Anderson & Krathwohl provide a modern update to Bloom's taxonomy
Characterizing Cognitive Complexity

<table>
<thead>
<tr>
<th>Knowledge Dimension</th>
<th>Cognitive Process Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Remember</td>
</tr>
<tr>
<td>Factual</td>
<td></td>
</tr>
<tr>
<td>Conceptual</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td></td>
</tr>
<tr>
<td>Meta-Cognitive</td>
<td></td>
</tr>
</tbody>
</table>

Anderson & Krathwohl, 2001
Teaching Style & Complexity

• From 1983 to 1993, I developed tester-training materials for within-company use

• From 1993 to now, I've been polishing these and teaching commercial courses
  – Taught the course over 100 times
  – Highly profitable
  – Money-back guarantee
  – Good customer satisfaction

• I tried a lot of variations over those 100 teachings
Teaching Style & Complexity

- Primary communication style was lecture
  - Real-life examples
    - Motivating
    - Memorable
    - Illustrate applications
    - Illustrate complexity
- Lectures can be excellent for conveying basic knowledge, but they are weak for developing higher order cognitive skills
## Characterizing Cognitive Complexity

<table>
<thead>
<tr>
<th>Knowledge Dimension</th>
<th>Cognitive Process Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Remember</td>
</tr>
<tr>
<td>Factual</td>
<td>lecture</td>
</tr>
<tr>
<td>Conceptual</td>
<td>lecture</td>
</tr>
<tr>
<td>Procedural</td>
<td>lecture</td>
</tr>
<tr>
<td>Meta-Cognitive</td>
<td></td>
</tr>
</tbody>
</table>

Anderson & Krathwohl, 2001
As a Commercial Instructor ...

- Client expectations
  - Broad coverage
  - Familiarize the student with new ideas
  - One useful idea per day for each student

- Activities
  - Tradeoff between lecture time and activity time
  - Remarkable resistance to time-cost and coverage-cost of activities

- This is drive-by training
  - Not a way to foster mastery or even grow new difficult skills
What Were My Students Learning?

- Exposure to new ideas
- Memorable stories that *could* connect to their experience and yield applications
- Cultural attitudes
- Definitions / descriptions
- But not skills
  - Domain testing as a striking example
    - Basic technique easy to explain
    - Application gets complex
    - Students tell me early on that they get it, their body language tells me the same, their questions are often good, but when I give application tasks --> not so good

(c) Cem Kaner 2006
So I Became a University Instructor

• GOAL:

  Improve the state of the practice in software testing by improving the test-related education of software testers

• AS A UNIVERSITY INSTRUCTOR

  – Students exposed to material over a longer period of time
  – Students expect to be tested in detail (none of this 3-good-ideas per term)
  – Homework and exams to work through ideas and build skills

• I started with lectures

  – Students gave me happy feedback on the lectures
  – Students indicated they understood me (e.g. Domain tests)
  – But they got creamed by exams and assignments
Planning a Better Course

- At university, students expect me to assess their knowledge
- And to provide activities that help them prepare for the assessments
- I get to learn what they know
- Which tells me a lot about my teaching (and its failings)
Assessment Ties to Learning Objectives

- Angelo & Cross: Teaching Goals Inventory
  - [http://www.uiowa.edu/~centeach/tgi/background.html](http://www.uiowa.edu/~centeach/tgi/background.html)

- Their broad categories include:
  - "Higher-order thinking skills"
  - Basic academic success skills
  - Discipline-specific knowledge and skills
  - Liberal arts and academic values
  - Work and career preparation
  - Personal development"
So I Developed Learning Objectives

- Learn many test techniques well enough to know how, when, and why to use them
- Foster strategic thinking--prioritization, designing tests/reports for specific audiences, assess the requirements for complex testing tasks (such as test automation, test documentation)
- Apply (and further develop) communication skills (e.g. for bug reporting, status reporting, specification analysis)
- Improve and apply teamwork skills (peer reviews, paired testing, shared analysis of challenging problems)
- Gain (and document) experiences that can improve the student's chances of getting a job in testing
They're Not Objectives
If You Don't Assess Against Them

- This is the core difference between university and commercial teaching

- It enables an approach to course design:
  1. What do I want to achieve in the course?
  2. How will I assess the students' knowledge / progress?
  3. What learning experiences will better prepare them for the assessments?

- Instructional theory books often present these as sequential questions. I don't know how to do that, so I evolve all on three dimensions in parallel
Back to Bloom

• The main application of the taxonomy was to help teachers map assessments to cognitive level of the learning
  – Factual knowledge --> memory tests
  – Apply procedures --> skill demonstrations
  – Evaluate concepts --> compare / contrast / justify
First Shot

• Study Guides:
  – 100 questions, include all candidates for mid-term and final exam
  – Students prepare answers together, assess each other's work
  – I can require well-organized, thoughtful answers
  – Fosters strategic preparation
  – Reduces disadvantage of students whose native language is not English
  – Creates cooperative learning tasks that should help limited-English-proficiency students improve language skills
Study Guides

- Study guide results
  - Students inexperienced with these, often blow the first test
  - Make-up mid-terms
    - Replace grade, not average, not best 1 of 2 results
    - Students who take it improve more ($1^{st}$ test compared to final exam) than students who did not take it
      - Practice effect, motivation confound
    - Writing is better, answers are better, I have greater freedom to grade less forgivingly
  - Many students told me this was the most valuable learning experience in the course, and the most time-consuming

(c) Cem Kaner 2006
Application Under Test

• Like service learning, but not as heavy a commitment for the students or for me
  – Facilitates student learning (application level and above)
  – Facilitates student transfer of skills / knowledge to the workplace
  – We pick a well-known product
  – Students apply what they learn to that product
  – Typically, I use an open source product because it avoids NDA problems, students can show their work at interviews
Assignments

- Take-home assignments required students to apply what they learned to the application under test
- Can only give so many assignments
  - Grading burden becomes impossible
  - Student workload becomes impossible
- I discovered that students needed more practice than I could cover with assignments
  - Domain testing was a striking example. Students needed several assignments (and feedback) for skilled performance on exams
Practice, practice, practice?

- My starting vision for the next level of instructional support was to create a broad set of practice exercises.
- Assignments went slowly, covered insufficient material, but I packed more in each year. Eventually, we would get a broad set of practice stuff.
- Padmanabhan implemented this approach for her M.Sc. thesis.
Padmanabhan's Thesis: Domain Testing

• 15 classroom hours of lecture plus examples plus practice, practice, practice. Lots of procedural instruction and drill

• Students mastered every procedure

• Final exam
  - Applied what they knew to similar questions
    • They aced them
  - Applied what they knew to a problem that was beyond their practice (not beyond the lecture)
    • They all failed miserably

• Successful transfer of learning requires more than procedural training and practice
Dealing With the Transfer Problem

In science / math education, the transfer problem has driven fundamental change in the classroom.

- Students learn (and transfer) better when they discover concepts, rather than by being told them.
Andragogy

- Pedagogy: study of teaching / learning of children
- Andragogy: study of teaching / learning of adults
- University undergrads are in a middle ground between the teacher-directed child and the fully-self-directed adult
- Both groups, but especially adults, benefit from activity-based and discovery-based styles
Now What?

- Practice is important (without it, my students make boneheaded mistakes)
- Good lectures are important (they present material that isn't available elsewhere, and in I-think-better ways than textbooks)
- But lectures and practice are insufficient
- So where do we go from here?
Back to the Instructional Basics

• What do I want students to be able to do?

• Here's an example:
  
  – We have a bunch of techniques (see next slide)
  
  – We have a bunch of attributes of techniques (see following slide)
  
  – I want students to be able to
    
    • Articulate the differences among the techniques in terms of strengths, weaknesses, and what they're good for
    
    • This is foundational for selecting the right technique for the context
    
    • And for creating new techniques

(c) Cem Kaner 2006
What’s a test technique?
Ten dominating techniques

- Function testing
- Specification-based testing
- Domain testing
- Risk-based testing
- Scenario testing
- Regression testing
- Stress testing
- User testing
- State-model based testing
- High volume automated testing

These are 10 common Examples.

There are many Others.

http://www.testingeducation.org/BBST/BBST--IntroductiontoTestDesign.html

(c) Cem Kaner 2006
Test attributes

To different degrees, good tests have these attributes:

- **Power.** When a problem exists, the test will reveal it.
- **Valid.** When the test reveals a problem, it is a genuine problem.
- **Value.** It reveals things your clients want to know about the product or project.
- **Credible.** Your client will believe that people will do the things that are done in this test.
- **Representative** of events most likely to be encountered by the user. (xref. Musa's *Software Reliability Engineering*).
- **Non-redundant.** This test represents a larger group that address the same risk.
- **Motivating.** Your client will want to fix the problem exposed by this test.
- **Performable.** It can be performed as designed.
- **Maintainable.** Easy to revise in the face of product changes.
- **Repeatable.** It is easy and inexpensive to reuse the test.
- **Pop.** *(short for Karl Popper)* It reveal things about our basic or critical assumptions.
- **Coverage.** It exercises the product in a way that isn't already taken care of by other tests.
- **Easy to evaluate.**
- **Supports troubleshooting.** Provides useful information for the debugging programmer.
- ** Appropriately complex.** As the program gets more stable, you can hit it with more complex tests and more closely simulate use by experienced users.
- **Accountable.** You can explain, justify, and prove you ran it.
- **Cost.** This includes time and effort, as well as direct costs.
- **Opportunity Cost.** Developing and performing this test prevents you from doing other work.
The Task

- See activity file
Classroom Support for Tasks Like This

- Students reported that their most valuable learning experiences were their out-of-class collaborations
- So let's bring the collaborations into the room, where we can coach them
- Drive 'every' class (well, most classes) with a coached, group activity that encourages students to think about what they're learning (apply / evaluate / create)
- So how do we make room for lectures?
Lectures On-Line

- [http://www.testingeducation.org/BBST](http://www.testingeducation.org/BBST)
- Video lectures
  - Students watch them before class
  - Take simple quiz that checks that they watched the video and paid attention
  - Then we do in-class activities
- The results seem good. I'm still evaluating (grad) student performance from last term compared to prior years, but my informal assessment at this point is that students did well
- Undergrads are struggling with this, but at a higher level than before (I think)
Nuts and Bolts

- I tape these myself and edit them myself
- Adobe Premiere (Final Cut Pro had too many bugs)
- 2-10 hours of tape for an hour of edited lecture
- Several hours of slide formatting / preparation for integration with video
- Several hours of preparation of lecture notes (tradeoff from scripted versus unscripted: spontaneous and active but hours more tape)
- Several hours editing, for a total of about 35 hours work per hour of taped class