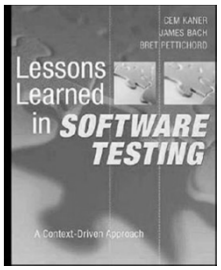
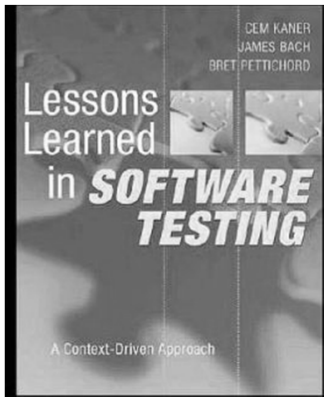


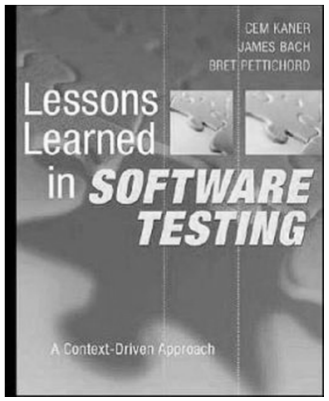
Theme 3: Testing Techniques

Testing Techniques



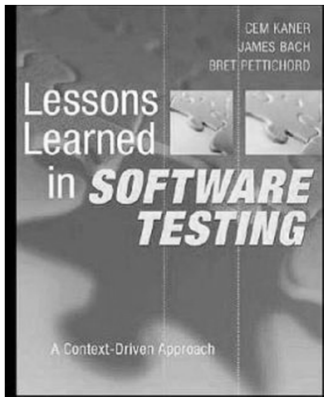
- Lesson 48: "Testing combines techniques that focus on testers, coverage, potential problems, activities, and evaluation"
  - Can be "about":
    - *Who* does the testing (e.g. user testing)
    - *What* gets tested (e.g. function testing)
    - *Why* you're testing (e.g. extreme value testing)
    - *How* you test (e.g. exploratory testing)
    - *How to tell pass/fail* (e.g. comparison to known good result)

Testing Techniques



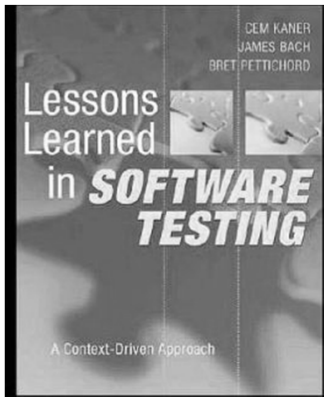
- Lesson 49: "People-based techniques focus on who does the testing"
  - User testing, obviously
  - Subject-expert testing
    - Designing a medical diagnosis system? You probably want some good doctors to evaluate it
  - "Eat your own dogfood"
    - Many companies release tools internally, without "testing" as a goal – just to see if their engineers can find bugs

Testing Techniques



- Lesson 50: "Coverage-based techniques focus on what gets tested"
  - Function testing
    - Cover every function of the program
  - Menu tour
  - Our coverage metrics discussed previously
    - Try covering all lines, branches, logical combinations…

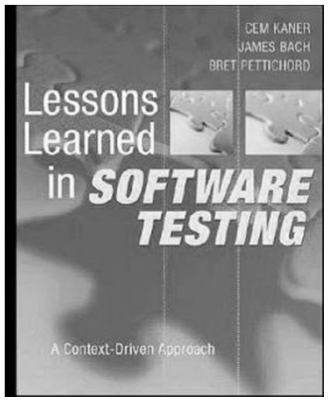
Testing Techniques



- Lesson 51: "Problems-based techniques focus on why you're testing (the risks you're testing for)"
  - Input constraints
  - Output constraints
  - Computation constraints
  - Storage (or data) constraints

  - Race conditions and timing issues are especially critical to look at here

Testing Techniques



- Lesson 52: "Activity-based techniques focus on how you test"
  - Regression testing
  - Scripted testing
  - Smoke testing
  - Exploratory testing
  - Guerrila testing
  - Installation testing
  - Load testing
  - Performance testing