Experimentation and Diagnosis

Many approaches to test design are based on established methods of experimental design. An understanding of the underlying principles can help testers streamline their existing tests and introduce effective new approaches.

Good testers need to be able to go beyond simply logging a problem. A good tester will be able to investigate her findings, giving value to her stakeholders and smoothing integration with her development teams. Diagnostic skills help a tester to isolate genuine problems from a rash of symptoms, to work out what lies behind field reports, and to communicate her bugs effectively by describing plausible models.

The course uses lessons from established practice in medicine, physics and statistics to introduce principles of experimentation and diagnosis. Exercises and real-life examples are used throughout to help participants apply these principles to their own work in software testing.

Description

Deborah Mayo writes that "methodology is the avoidance of error". The course looks at different experimental methodologies and how they reflect their environment and use. Participants design and execute experiments, allowing direct experience. We apply the principles to software testing, giving perspective to a wide variety of software testing approaches, from performance testing to exploratory testing.

One key experimental approach within software testing is the diagnosis of recognised problems. The course uses a succession of practical exercises based on real problems including truncation, bottlenecks, boundaries and emergent behaviours. Participants will select test conditions to isolate and emphasise a bug, analyse data to reveal connections and populations, and work with logs and events to arrive at sequences that reveal potential cause and effect.

At the end of the workshop, participants will be able to design better tests, and will have an improved understanding of the techniques and principles of diagnosis that can be applied to issues found in their own systems.

The tutorial will have plenty of exercises, abstract and practical, and will stimulate everyone in the test team who designs, executes, or plans tests.

Who should attend

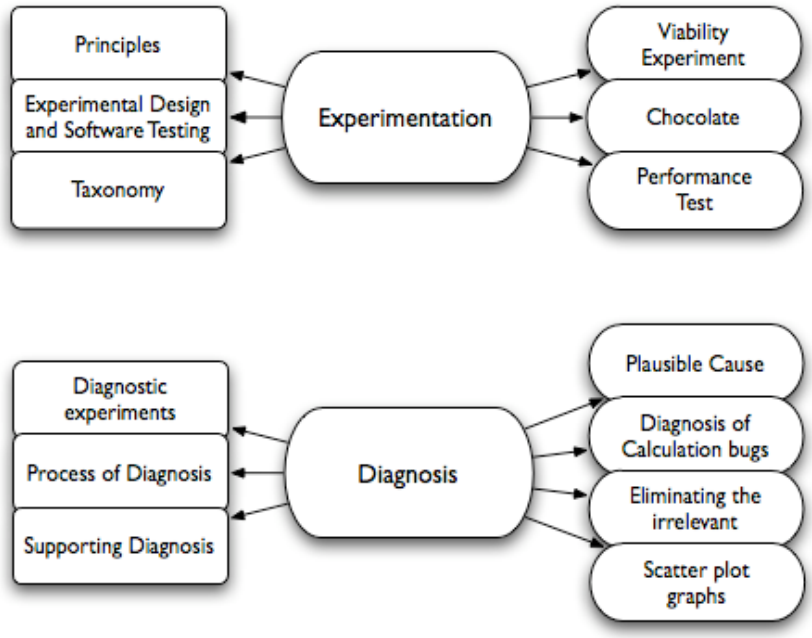
This tutorial will be of greatest use to test analysts and senior testers, but will also be immediately relevant to coders, especially those looking to enhance their skills in debugging. Direct experience of testing is not necessary, but delegates with one or more year's hands-on experience will get most from this course.

Learning Objectives

- Learn new test techniques
- Gain a fresh perspective on known techniques
- Understand how to use scientific principles can be used in testing
- Be able to justify diagnostic depth – how far should a tester go when diagnosing a bug
- Application of experimentation to testing in general, with special attention to exploratory testing
- Understand the process of diagnosis
- Understand the adaptability and suitability of experimental methods
- Be able to measure/quantify/analyse experimental effort in testing

Hands-on exercises and facilitated discussions will be used throughout the course. Each section will be concluded with a wrap-up session to summarise and reinforce key points. Participants' workbooks contain sections to record conclusions; participants are encouraged to review these after the course is completed.

Course map



Please note . . .

Many exercises are computer-based. Delegates are encouraged to download exercises before the tutorial, and bring their laptops to use and share. Exercises will use Workroom Productions' custom-built "Black Box" machines. These Flash-based interactives run in-browser, on most platforms, without installing executables or library files. The same platform is used for tester puzzles, which can be found at: http://www.workroom-productions.com/black_box_machines.html .

Presenter

James Lyndsay is an independent Test Strategist, based in London. He started testing in 1986, and has been the principal consultant at Workroom Productions since its formation in 1994. As a consultant, he's worked in a variety of businesses and project styles; from retail to telecommunications, from rapidly-evolving internet start-ups to more traditional large-scale enterprise. He's worked to technical requirements for companies that make and sell software, and to commercial requirements for companies that buy and use software.

James is a regular speaker at international test conferences, delivering invited keynote talks at STAREast, ASIAStar and EuroSTAR. He was an internal irritant to the ISEB exam process for five years, runs LEWT (the London Exploratory Workshop in Testing) and has won prizes for his papers.

See http://www.workroom-productions.com/ for more details.