Introduction

Dewayne E Perry
ENS 623
perry@mail.utexas.edu
Course Goals

- Create/evaluate empirical studies/data
  - Understand underlying scientific and theoretical bases
  - Understand fundamentals of experimental design
    - Independent and dependent variables
    - Variable manipulation/data gathering
    - Various issues of validity
    - Empirical logic and reasoning
    - Analytic tools for reasoning about the experiment
  - Understand and control confounding variables
    - Random and systematic biases
    - Alternative explanations
  - Understand and apply appropriate data analysis techniques
  - Understand and create the underlying logics of empirical studies in reasoning out conclusions
Experimenter/Evaluator Goals

- Responsible skepticism
  - Look for
    - Failures in experimental designs
    - Failures of observations
    - Gaps in reasoning
    - Alternative explanations
  - Compare new evidence against old
  - Raise counter objections/hypotheses
  - Question grounds for doubt as well
  - Accumulate weight of evidence
Good Research Practices

- Enthusiasm – an enjoyable endeavor
- Open-mindedness – keen, attentive, inquisitive
- Common sense – avoid looking under the lamppost
- Inventiveness – creative
  - Not only in experimental work
  - In resource management as well
- Confidence in one's own judgment
  - Despite detractors when right
  - Know when you are wrong
- Consistency and care about detail
  - No substitute for accuracy - keep complete records, organize and analyses accurately/carefully
- Ability to communicate
  - Writing is a superb, essential research technique
  - Make your discoveries known to others
- Honesty – integrity and scholarship
Review of SWE

- **Factors in software engineered products**
  - **Theory (basis for product)**
    - CS Core
    - Domain specific theory
  - **Experience (basis for judgment)**
    - Feedback
    - Engineering experiments (prototypes)
    - Empirical studies
      - Observations
      - Correlations
      - Causal connections
  - **Process (basis for production)**
    - Methods and techniques
    - Technology
    - Organizational structures
      - Teams
      - Projects
      - Cultures
Review of SWE

- **SW development processes**
  - **Phases**
    - Requirements
    - Architecture
    - Design
    - Construction
    - Deployment and maintenance
  - **Integral to all phases**
    - Documentation
    - Measurement & Analysis
    - Evolution
    - Teamwork
    - Management of artifacts
Daily Life of SWE: Decisions

- Determine what users want/need
- Make architecture, design and implementation decisions
- Evaluate/compare architecture and design choices
- Evaluate functional characteristics
- Evaluate non-functional properties
- Evaluate/compare technologies
  - ★ For supporting tools
  - ★ For product support
  - ★ For process support
- Determine what went wrong
- Determine good resource allocations
- etc
Some Illustrative Examples

- Most frequently stolen car
- Language/Solution comparisons
- Inspection reading techniques
- Exit Interviews
Most Frequently Stolen Car

- **Honda Accord**
  - Study presented in media
    - Honda most frequently stolen automobile
  - What does this study tell us?
    - Actually very little
  - What can we infer from this?
    - Shouldn’t buy a Honda?
    - Buy a Mercedes instead?
  - Very misleading

- More recently
  - Frequency relative to the total number of cars
    - Claims per 1000
  - Different story
    - Lincoln Navigator - 12.2; Cadillac Escalade - 10.3; etc
      - All Escalades recovered - GPS system
    - Honda and Camry didn’t make it anywhere near the top 10
      - Percentage of thefts claims is low compared to the number on road
Language/Solution Comparison

- From a summer project
- Comparing Java/C
- 2 different computers used
  - Pentium III 600MHz 128M
    - C 1.4 times faster than Java
  - AMD 1GHz 256M
    - Java 1.09 times faster than C
- Conflicting evidence
- How do we account for it?
- What are the differences?
- How do we resolve these differences?
- What can we conclude, if anything?
Preview of Confounding Variables
Inspection Reading Techniques

- **Experiment:** evaluate reading techniques for object oriented code inspections
- **Inconclusive results**
- **Possibilities:**
  - ★ There are no differences
  - ★ Poor experimental design
  - ★ Insufficient data
  - ★ Poor analyses and reasoning
Surveys

- Exit Polls
  - Used to predict outcome

- 2004 election
  - Significant disagreement with final vote
  - Early calls retracted

- Pollster’s response
  - Results with in margin of error
  - Data selectively reported

- Possible biases
  - Questions asked
    - slanted, non-uniform (ie general vs specific), etc
  - Time of polls
    - 7am - men on the way to work
    - 10am - soccer mom’s after kids are in school
  - Place of polls
    - East Austin versus Westlake
  - Nature of volunteers
Current State of SE Empirical Work

- Implementation oriented
  - Fenton: poor statistical designs, don’t scale
  - Basili: differences in projects make comparisons difficult
  - Johnson: practitioners resist measurement

- Need to be more requirements oriented
  - Think hard about what experiments really are
  - How they can be most effectively used

- Core problem: conceptualizing and organizing a body of work as a scientific basis
Software Development

- Little hard evidence to inform decisions
  - Correlations suggestive but not sufficient in all cases
  - Many times don’t know exception cases
- Do not know fundamental mechanisms
  - Software tools
  - Methods and techniques
  - Processes
- Empirical studies are the key
  - Show mechanisms
  - Eliminate alternative explanations
  - Empirical validation is standard in some fields
  - Quality of empirical studies in SE is rising
  - Funding agencies recognizing value of empirical studies
  - Increasing number of tutorials, panels, SOTAs, papers, etc
  - Key consciousness raising papers (Tichy et al, Zelkowitz)
  - Several key organizations: SEL, ISERN
Systematic Problems

- Research ideas are not empirically validated
  - Should retroactively validate
  - Should proactively directed
- Search for perfect study
  - Instead of focus on credibility
- Study the obvious
  - OK, but we need deeper insights
- Lots of data
  - Not enough – should answer important questions
- Lack useful hypotheses
- Lack conclusions from data
Empirical Software Engineering Studies

- Individual programmer studies have credibility due to well understood techniques from psychology and statistics.
- Large software development studies with the addition of large population social factors are not well established or credible.
- General goal:
  - Establish a spectrum of empirical techniques that are robust to large variances from social factors present.
Challenges

❖ Create better empirical studies
   ★ Establish principles that are
     ➢ Causal: correlated, temporally ordered, testable theory
     ➢ Actionable: causal agent effectively controllable
     ➢ General: widely applicable
   ★ Answer important questions
   ★ Family of focused studies – illuminate related aspects
   ★ Cost effective and reproducible

❖ Credible interpretations
   ★ Degree of confidence we have in conclusions
     ➢ Eliminate alternative explanations
     ➢ Provide a compelling logic in the discussion
   ★ Validity is critical: construct, internal, external
   ★ Hypothesis is critical: ask important questions
   ★ Resolutions appropriate to the intent of the study
   ★ Make the data public
Some Concrete Steps

- **Designing studies**
  - **Ask significant questions**
    - Knight-Leveson, N-version programming
  - **Families of studies**
    - Schneiderman et al, on the value of flowcharts
  - **Build partnerships**
    - Takes time; multi-person effort; interdisciplinary, industry
  - **Long running in vivo/situ experiments**
    - Subparts; subject rights; know when to stop

- **Collecting data**
  - **Retrospective artifact analysis**
    - Eg, version management systems
  - **Simulation and modeling**
    - Eg, integration studies of Solheim and Rowland

- **Involving others**
  - **Meta-analysis - eg Porter and Johnson**
  - **Educational laboratories**
    - Teach empirical studies basics
    - Populate lab with appropriate data/designs/equipment
Goals for SWE Empirical Studies

- **Some help**
  - Look to other empirical disciplines
  - Adapt what is useful here

- **Goals**
  - Perform better empirical studies
  - Focus on causal mechanisms
  - Generate theories
  - Iterate and improve

- **Good empirical studies enable us to**
  - Encode knowledge more rapidly
  - Prune low payoff ideas rapidly
  - Recognize and value high payoff ideas
  - Exploit important practical ideas
What is Critical

- **Fundamentals**
  - Credible interpretations
  - Repeatability
  - Understanding validity limits
  - Identifying underlying mechanism
  - Practical significance

- **Non-fundamentals**
  - Whether qualitative or quantitative
    - Both have their place and usefulness
  - Identical results
    - Want congruent results
  - Correlation studies
    - Important precursor, but not the goal
  - Opportunistic studies
Structure of an Empirical Study

- Research context
  - Problem definition
  - Research review
- Hypothesis
  - Abstract - about the world
  - Concrete - about the design
- Experimental design
  - Variables - independent and dependent
  - Plan to systematically manipulate variables
  - Control operational context
- Threats to validity: construct, internal, external
- Data analysis and presentation
  - Quantitative: hypothesis testing, power analysis
  - Qualitative
- Results and conclusions
  - Limits, influences
  - Explain how answered question and its practical significance
  - Sufficient information for repeatability