# Empirical Software Engineering

# CSE 8340 — Spring 2014

Prof. Jeff Tian, tian@lyle.smu.edu
CSE, SMU, Dallas, TX 75275
(214) 768-2861; Fax: (214) 768-3085
www.lyle.smu.edu/~tian/class/8340.14s

## Module Ic: ESE Example

- ESE Study as an Example
- Hypothesis about FM
- Analysis and Results

# ESE Example and Guidelines

- *ESE Example:*
  1997 paper by Pfleeger and Hatton
  IEEE Computer 30(2):33-43.

- Use ESE Guidelines:
  2002 paper by Kitchenham, Pfleeger, Pickard, Jones, Hoaglin, Emam, Rosenberg
  (TSE 28(8):721-734).

- Context of our discussion:
  - ▷ Guideline applied to ESE study.
  - ▷ 6 steps (topic areas)
  - ▷ Focus on analysis (and conclusions)

# ESE Study on FM

- Hypothesis testing:
  - ▷ Can FM deliver?
  - ▷ Implicit hypothesis: Promises of FM.
  - ▷ Informal hypothesis testing.

- What is FM?
  - ▷ FM: formal methods.
    (formal spec. & formal verification)
  - ▷ Applied to software development (phases)
  - ▷ Basic idea in 7314 and 8317
  - ▷ Specifics in Pfleeger/Hatton

- Past work on same question:
  see insert by Fenton and Pfleeger.

# TA1: Context

- C1: Clearly specify industrial context
  - ▷ Company: Praxis
  - ▷ Product: air-traffic control IS
  - ▷ Customer: UK Civil Aviation Authority
  - ▷ Size: 200,000 LOC in C
  - ▷ observational studies/details below

- FM in requirement:
  - ▷ ER analysis
  - ▷ real-time Yourdon-Constantine SA
  - ▷ formal spec. language: VDM, CCS etc.

- FM in design:
  - ▷ VDM/CCS specs for code
  - ▷ FSM to define concurrency
  - ▷ pseudocode for UI

# TA1: Context

- C2: Hypothesis (if any)
  - ▷ Can FM deliver?
  - ▷ null and alternative hypothesis
  - ▷ basis: past work in FM
- C3: if exploratory research: No.
- C4: describe related research
  - ▷ insert by Fenton and Pfleeger.
  - ▷ much promises
  - ▷ no conclusive results

# TA2: Design

- Elements of experimental design:
  - ▷ population
  - ▷ sampling technique and rationale
  - ▷ treatment (or intervention)
  - ▷ bias and sample size

- In Pfleeger/Hatton study:
  - ▷ population: 1 product
  - ▷ observational case study
  - ▷ all fault data used
  - ▷ D1-D11 not formally addressed

# TA3: Data Collection

- Data collection: common guidelines.
  - ▷ DC1: define all measures fully.
  - ▷ DC2: properly treat subjective ones
  - ▷ DC3: accuracy/completeness of DC
  - ▷ DC4: resp. rate & representativeness
  - ▷ DC5: drop-outs? (for experiments)
  - ▷ DC6: other performance measures also

- In Pfleeger/Hatton:
  - ▷ DC1: measure definition
    – fault reports from in-house testing
  - ▷ in connection with data analysis
    (particularly: understanding data)
  - ▷ DC2–DC6 irrelevant.

# TA4: Analysis

- Analysis guidelines:
  - ▷ A1: careful with multiple testing ("torture/fishing" the same set of data?)
  - ▷ A2: consider using blind analysis (reduce subjective tendencies)
  - ▷ A3: perform sensitivity analysis
  - ▷ A4: match data with test
  - ▷ A5: verify the results

- In Pfleeger/Hatton:
  - ▷ in connection with analysis steps
  - ▷ 5 steps (details later)
  - ▷ fairly simple statistics
  - ▷ also include result presentation, interpretation and conclusions.

# TA4: Analysis

- Step 1: Understand the data
  - ▷ DC1: define all measures fully (previous guideline topic area)
  - ▷ fault reports are actually failures
  - ▷ severity 1, 2, 3: all failure related
  - ▷ around 3000 fault reports
  - ▷ 1990 to June 1992 (delivery)
  - ▷ traced to modules (which is changed?) but little root cause analysis

- Step 2: Looking for diff. in #changes
  - ▷ module changes from fault reports
  - ▷ quantitative questions regarding:
    - – FM quantitatively affect code quality?
    - – Was one FM superior to another?
  - ▷ results presented in Tables 1 and 2
  - ▷ related interpretation/discussions
  - ▷ conclusion: no sig. differences

# TA4: Analysis

- Step 3: Look for trends
  - ▷ one question (no sig. diff. in avg) leads to another (over time diff.?)
  - ▷ results in Fig. 2
  - ▷ related discussions:
    - – onset of testing in qt.4
    - – possible size/complexity diff.
  - ▷ comment: uncontrolled factors

- Step 4: Conduct a code audit
  - ▷ try to explain Step 3/Fig. 2 above
  - ▷ potential faults remaining per module
  - ▷ complexity analysis
  - ▷ results: Fig. 3, high quality
    - – simple design, loose coupling
  - ▷ but not attributed to design methods

# TA4: Analysis

- Step 5: Examine the results of unit testing
  - ▷ easy to test (and early)?
  - ▷ overall faults distribution:
    - insp.: 340, UT: 725, ST/AT: 2200
    - different from prev. studies
  - ▷ UT results: Table 3
    - formal lower than informal (UT pb.)
    - implications: formal better/cleanroom?
  - ▷ postdelivery $\Rightarrow$ next question

- Step 6: Evaluate postdelivery changes
  - ▷ results: Table 4
  - ▷ formal better than informal
  - ▷ indistinguishable within different FM
  - ▷ comparison: Tables 5 and 6
  - ▷ direct & indirect effect of FM:
    - conformance to req. (direct)
    - highly testable system (indirect)

# TA5: Result Presentation

- Presentation guidelines:
  - ▷ P1: describe/ref. for stat. procedures
  - ▷ P2: statistical package used
  - ▷ P3: enough details (sig. level etc.)
  - ▷ P4: raw data whenever possible
  - ▷ P5: appropriate descriptive statistics
  - ▷ P6: make appropriate use of graphics

- In Pfleeger/Hatton:
  - ▷ simple statistics: no need to explain
  - ▷ most of Px's irrelevant
  - ▷ in connection with data analysis
  - ▷ good use of tables/graphics

# TA6: Result Interpretation

- Interpretation guidelines:
  - ▷ I1: describe inferential statistics or predictive models
  - ▷ I2: stat. sig. $\neq$ practical importance
  - ▷ I3: define the type of study
  - ▷ I4: specify study limitations

- In Pfleeger/Hatton:
  - ▷ simple statistics/interpretation
  - ▷ most of Ix's irrelevant
  - ▷ in connection with data analysis
  - ▷ summarized in lessons learned section

# TA6: Result Interpretation

- Lessons about formal methods:
  - ▷ pre-delivery similar
  - ▷ UT and post-delivery: FM better
  - ▷ high-quality audit profile:
    – simple, independent components
  - ▷ FM in concert with other SE initiatives

- Lessons about empirical investigation:
  - ▷ data availability issue:
    expr./size data, other projects, etc.
  - ▷ data consistency: fault vs failure
  - ▷ separate pre-/post-delivery data
  - ▷ other limitations

# TA6: Result Interpretation

- Overall: inconclusive, but some indications

- Recommendation to practitioners:
  - ▷ data defn/coll in planning to evaluate task effectiveness and product quality
  - ▷ trend and relationship identification
  - ▷ Be skeptical: quantitative evidence?

- Comments by Tian:
  - ▷ focus: data analysis
  - ▷ simple statistics/interpretation
  - ▷ good ESE example
  - ▷ good ESE guideline test/example
    – relate to hw#2&3 analysis/critique