Verification and Validation for Concurrent Programs

A State-of-Practice Questionnaire

Margaret A. Wojcicki
School of Information Technology and Electrical Engineering, University of Queensland
e-mail: wojcicki@itese.uq.edu.au

Overview
- Background
- Related Work
- Survey Methodology
- Survey Results
- Survey Analysis
- Conclusions and Future Work

Background
- Verification: does a computer program conform to its specification?
- Validation: does a computer program meet the requirements of the client?
- Verification and validation (V&V): dynamic or static
- Cost-effectiveness factors: types of defects, cost of defect detection and false-positives (mostly static)

Related Work

<table>
<thead>
<tr>
<th>Survey</th>
<th>Methodology</th>
<th>Principal Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: V&amp;V Australia (04)</td>
<td>Online, telephone and interview</td>
<td>Low automation</td>
</tr>
<tr>
<td>2: V&amp;V Sweden, US and Multinational (05)</td>
<td>Online</td>
<td>Low automation, low unit testing</td>
</tr>
<tr>
<td>3: V&amp;V Canada (04)</td>
<td>N/A</td>
<td>Low automation, low unit testing</td>
</tr>
<tr>
<td>4: V&amp;V Sweden (05)</td>
<td>Workshops and interviews</td>
<td>Larger companies more systematic and use commercial tools</td>
</tr>
<tr>
<td>5: Software development NZ (05)</td>
<td>Telephone and in person</td>
<td>Larger companies more systematic</td>
</tr>
<tr>
<td>6: Software Testing Institute (02)</td>
<td>Online</td>
<td>Online source of V&amp;V technology information</td>
</tr>
<tr>
<td>7: Quality Assurance Institute (03)</td>
<td>Online, Conference on Software Testing</td>
<td>Cost reviews are most common static test</td>
</tr>
</tbody>
</table>

Survey Methodology
- Objectives:
  1. Examine current V&V technology use and context of use
  2. Examine decision process for V&V selection
- Survey sections:
  1. Context of V&V Technology
  2. Source of information
  3. Case Study Aspects
  4. Cost-effectiveness factors
Survey Methodology

- Method:
  self-recruited + cross-sectional = non-systematic sampling
- Sample selection (convenience):
  1. Concurrency-interest mailing list
  2. IBM developerWorks Multithreaded Java Programming forum
  3. Snowballing
  4. 5% response rate adequate
     (Lethbridge et al)

Survey Results - Demographics

- 35 responses
- Majority have at least 3 years of V&V experience

Survey Results – Programming Languages

- Predominantly Java, but also C++ (smaller numbers using Ada and C)
- average of 7.3 years experience with Java
- currently using version 1.5 or more recent

Survey Results – Scope and Defects

- Frequencies of Scope of V&V
- Frequencies of Concurrency Defects

Survey Results – V&V

- Code inspection/code walkthrough predominantly done ad-hoc and individually
- 60% use open source technology: mostly JUnit, FindBugs
Survey Results – Decision Process

Importance of Source of Information in Determining V&V Technology to Use

- History of use in other companies
- History of use in current company
- Marketing information
- Search engine results on V&V technology
- New groups/forums
- Research
- Literature

Number of Respondents

Survey Results – Case Studies

Importance of Case Study Factors for V&V Selection

- Case study in similar application domain
- Case study with real defects
- Case study with large number of participants
- Case study in industry

Number of Respondents

Survey Results – Cost-effectiveness

Ranking of Cost-Effectiveness Factors

- Type of Defects
- False Positives
- Cost of Using Tool

Survey Analysis – Needs

- V&V technology:
  - Java 1.5, C++
  - Code inspection/code walkthrough (automation)
  - Systematic inspections (groups)
  - V&V technologies (coverage tools and model-checkers)

- Case studies:
  1. Real defects (bug repositories, benchmarks)
  2. Measurable results

- Decision Process:
  - History in Industry – Research Literature

Survey Analysis – Needs

- Cost-effectiveness:
  - Willing to deal with false-positives (quality reports)
  - Also: code conversion, implementation/maintenance of tool, purchase cost relative to defects caught

- Previous surveys:
  - Low Automation
  - Unit testing versus acceptance (surveys 2 & 3)
Survey Analysis

- Threats to Validity:
  - Low internal, but high external validity
  - Generalise to software developers, but no control on background/knowledge
  - Non-response bias (difficult to calculate)
  - Convenience sampling
- Other issues:
  - Definition of V&V (correctness)
  - Sources of Information question biased to companies
  - Ranking question (defaults to 1)

Conclusions

- V&V mostly manual, aided with development testing tool automation (JUnit)
- Importance of real defects in case studies
- Many aspects of cost-effectiveness
- Qualitative information improves understanding of context

Future Work

- Examine locking mechanism used by practitioners
- Focus on performance issues rather than correctness and debugging tools
- In depth examination of C and C++ (memory management)
- Modification of TestCon and schema

Questions and Answers