Using Version Control Data to Evaluate the Impact of Software Tools: A Case Study of the Version Editor

David L. Atkins, Thomas Ball, Todd L. Graves, Audris Mockus

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 2005 June

Jin Ung, Oh – 2006.11.08
Contents

- Introduction
- Background
- VE (Version-Sensitive Editor) Tool
- Hypothesis
- Impact of VE on Developer’s Effort
- Impact of VE on Feature Interval
- Conclusion
Introduction

• *To Help*,
  – Making decisions about the tool use more effectively.

• *Do*,
  – Quantify the effectiveness of software tools.
  – Assess impact of a version-sensitive text editor called VE.

• *Using*,
  – Software change history data.

• *Problem is*,
  – How to control for key sources of variations.
    • Developer work-style and experience.
    • Size of changes to software.
    • Type of changes (new feature, bug fix, code cleanup, code inspection).
Introduction

• Two observation
  – Major effect of a software tool
    • To help developer determine how to modify a software entity or,
    • Directly aid the developer in making modification.
  – Change history can be used to estimate effort

• Approach
  1. Record the tool usage.
  2. Relate monitoring information (1) to the modifications to software entity history in VCS (Version Control System).
  3. Using Effort Estimation Algorithm, analyze “similar” developers and modifications to estimate how the use/nonuse of the tool affected developer effort and overall interval.
Background

• Available Data

```java
if (!PreCheckRoute(route))
    return FAIL;

dest = GetDest(route);
...

// $Edited by VE 1996/…

// VE Signature
```

(VTypes of MR)
- NEW
- BUG
- CLEANUP
- INSPECT

< Developer >

Monthly Report

Report

MR Assigned

MR

Check out

Check in

VCS
Background

- # version Problem

```c
... if (!PreCheckRoute(route)) return FAIL;
    #version (4A)
    dest = GetDest(route);
    if (dest.port == 0) {
        return(RouteLocal(route));
    }
    #endversion (4A)
    DoRoute(route);
...
```

```c
... if (!PreCheckRoute(route)) return FAIL;
    #version (4A)
    dest = GetDest(route);
    #version (4A)
    if (dest.port == 0) {
        #endversion (4A)
        DoRoute(route);
    } else if (dest.port == 0 || dest.module == 0) {
        #endversion (4A)
        return(RouteLocal(route));
    }
    #endversion (4A)
    DoRoute(route);
...```
VE Tool: Introduction

• Version-Sensitive Editor (used in Bell Lab)
  – Display the extracted view of a certain version.
  – Leaves signature when editing.
  – Used optionally.

Release 5A view in VE

```c
if (!PreCheckRoute(route))
    return FAIL;
dest = GetDest(route);
if (dest.port == 0 || dest.module == 0) {
    return(RouteLocal(route));
}
DoRoute(route);
```

MR 12467 by dla, 97/9/21, assigned [Local routing]
Versioning: 5A inside 4A
"route.c" [modified] line 67 of 241
VE-Tool: Usage Pattern

Size of files and fraction of `#version` lines in on subsystem.
VE-Tool: Usage Pattern

Fraction of developers using VE over time >

< VE usage over time.
Hypothesis

- VE tool reduces the effort needed to make changes involving preprocessor directives (#version lines)

- Usage of VE would lead to shorter development intervals.

- Usage of VE would lead to better modification quality.
Impact of VE on Developer’s Effort

• **Experiment Scenario**
  – **Generate Effort Estimation Model**
    • Using statistical method, (Expectation-Maximization)
    • Using well-selected data,
    • Generate effort estimation model.
  – **Using generated model,**
    • Analyze coefficient of each factors.
    • Choose primary effort driver,
    • Regenerate simpler effort estimation model.

• If tool usage is chosen to be statistically significant **effect driver**, we can validate our hypothesis.
Impact of VE on Developer’s Effort

- **Effort Estimation Model**
  - “Inferring Change Effort from Configuration Management Data”, Metrics 98, pp.267-273, Nov. 1998

\[
E(\text{effort}) = \alpha_{\text{DEV}} \times \beta_{\text{TYPE}} \times \text{Size}^\gamma \times \theta_{\text{TOOL}}
\]

- DEV : developer identity.
- TYPE : type of change (NEW, BUG, CLEANUP, INSPECT).
- Size : # of deltas. (or # of lines)
- TOOL : use or nonuse of VE. (VE, HAND, NONE)

- To fit the above regression model,

  *We should measure effect for each change completed by developers.*
Impact of VE on Developer’s Effort

- Estimation Sequence

<table>
<thead>
<tr>
<th>TABLE 1</th>
<th>Data Available in Effort Estimation Problem, for a Single Developer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort for MR A</td>
<td>Jan</td>
</tr>
<tr>
<td>Effort for MR C</td>
<td>0</td>
</tr>
<tr>
<td>reported effort</td>
<td>1.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLE 2</th>
<th>Initialization of Effort Modeling Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort for MR A</td>
<td>Jan</td>
</tr>
<tr>
<td>Effort for MR B</td>
<td>1.0</td>
</tr>
<tr>
<td>Effort for MR C</td>
<td>0</td>
</tr>
<tr>
<td>reported effort</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Impact of VE on Developer’s Effort

• Estimation Sequence (Cont’d)

<table>
<thead>
<tr>
<th>TABLE 3</th>
<th>Rescaling Developers’ Monthly MR Efforts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jan</td>
</tr>
<tr>
<td>Effort for MR A</td>
<td>0.8</td>
</tr>
<tr>
<td>Effort for MR B</td>
<td>0</td>
</tr>
<tr>
<td>Effort for MR C</td>
<td>0</td>
</tr>
<tr>
<td>reported effort</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Using generalized linear model.

<table>
<thead>
<tr>
<th>TABLE 4</th>
<th>Rescaling Developers’ Monthly MR Efforts So that in Each Month the Developer Spent the Correct Amount of Total Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Jan</td>
</tr>
<tr>
<td>Effort for MR A</td>
<td>1.0</td>
</tr>
<tr>
<td>Effort for MR B</td>
<td>0</td>
</tr>
<tr>
<td>Effort for MR C</td>
<td>0</td>
</tr>
<tr>
<td>reported effort</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Impact of VE on Developer’s Effort

• Developer Selection
  – Chose developers having “similar” work profile.
  – 9 Developers who,
    • Made 300 – 500 MRs in 6 years. (1990 – 1995)
    • Similar # of MRs with and without VE. (# > 40)
Impact of VE on Developer’s Effort

• Generated Model

\[ E(\text{effort}) = \#\text{delta}^{\alpha_1} \times \#\text{lines added}^{\alpha_2} \times \]
\[ \beta_{\text{BUG}} \times \beta_{\text{CLEANUP}} \times \beta_{\text{INSPECT}} \times \]
\[ \gamma_{\text{HAND}} \times \gamma_{\text{NONE}} \times \prod_i \delta_{\text{Developer}_i} \]

\[ E(\text{effort}) = \beta_{\text{BUG}} \times \gamma_{\text{HAND}} \times \gamma_{\text{NONE}} \times \prod_i \delta_{\text{Developer}_i} . \]

• Result

<table>
<thead>
<tr>
<th>Model</th>
<th>Coefficient</th>
<th>Estimate</th>
<th>p-val</th>
<th>95% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \alpha_1 )</td>
<td>0.15</td>
<td>0.4</td>
<td>([-2, 0.5])</td>
</tr>
<tr>
<td></td>
<td>( \alpha_2 )</td>
<td>-0.08</td>
<td>0.3</td>
<td>([-2, 0.1])</td>
</tr>
<tr>
<td></td>
<td>( \beta_{\text{BUG}} )</td>
<td>1.44</td>
<td>0.01</td>
<td>[1.1, 1.8]</td>
</tr>
<tr>
<td></td>
<td>( \beta_{\text{CLEANUP}} )</td>
<td>0.6</td>
<td>0.4</td>
<td>[0.2, 2]</td>
</tr>
<tr>
<td></td>
<td>( \beta_{\text{INSPECTION}} )</td>
<td>0.7</td>
<td>0.8</td>
<td>[0.1, 7.6]</td>
</tr>
<tr>
<td></td>
<td>( \gamma_{\text{HAND}} )</td>
<td>1.46</td>
<td>0.04</td>
<td>[1.01, 2.1]</td>
</tr>
<tr>
<td></td>
<td>( \gamma_{\text{NONE}} )</td>
<td>0.7</td>
<td>0.3</td>
<td>[0.4, 1.3]</td>
</tr>
<tr>
<td>Full</td>
<td>( \beta_{\text{BUG}} )</td>
<td>1.5</td>
<td>0.00</td>
<td>[1.2, 2]</td>
</tr>
<tr>
<td>Minimal</td>
<td>( \gamma_{\text{HAND}} )</td>
<td>1.5</td>
<td>0.03</td>
<td>[1.04, 2.2]</td>
</tr>
<tr>
<td></td>
<td>( \gamma_{\text{NONE}} )</td>
<td>0.8</td>
<td>0.3</td>
<td>[0.4, 1.4]</td>
</tr>
</tbody>
</table>
Impact of VE on Developer’s Effort

• Validation for interaction between factors.
  – Developer <-> VE usage
    • Minimized by selecting similar developers.
  – Size <-> VE usage
    • Insignificant.
  – Type of change <-> VE usage
    • Validated with the model without factor for type of change.

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Estimate</th>
<th>p-val</th>
<th>95%CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma_{HAND}$</td>
<td>1.5</td>
<td>.04</td>
<td>[1.01, 2.2]</td>
</tr>
<tr>
<td>$\gamma_{NONE}$</td>
<td>0.8</td>
<td>0.37</td>
<td>[0.5, 1.4]</td>
</tr>
</tbody>
</table>

• Does not affect tool effect.
  – Time <-> VE usage
    • Remaining problem.
Impact of VE on Feature Interval

• Mapping Feature & Interval to Available Data
  – Feature
    • Set of MRs and deltas.
  – Feature Interval
    • Time between first and the last delta for that feature.
  – Validation
    • Actual data from process step information.
    • Median ratio of MR-derived interval : 0.6

• Hypothesis
  – Given that VE reduces effort for individual MRs
  – VE would reduce MR interval, and possibly feature interval.
Impact of VE on Feature Interval

• Regression Model

\[
\log \text{Interval} = \beta_1 + \beta_2 \log \text{Size} + \beta_3 \text{VE} + \text{error},
\]

- log (for Interval and Size) used for make their distribution closer to a Gaussian distribution.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Estimate</th>
<th>Std. Error</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept  ($\beta_1$)</td>
<td>2.94</td>
<td>0.06</td>
<td>&lt; 0.01</td>
</tr>
<tr>
<td>$\log \text{Size}$ ($\beta_2$)</td>
<td>0.58</td>
<td>0.01</td>
<td>&lt; 0.01</td>
</tr>
<tr>
<td>VE ($\beta_3$)</td>
<td>-0.46</td>
<td>0.06</td>
<td>&lt; 0.01</td>
</tr>
</tbody>
</table>

- ANOVA result shows highly significant impact of the VE tool.
- $R$ value : 0.62
Conclusion

- **Using**
  - Version Control Data,
  - Appropriate tool usage signature,
  - Statistical Method,

- **We can provide**
  - “witness” data for effectiveness of tools,

- **And**
  - The methodology presented here can be applied to other settings