Process Mining Multiple Repositories for Software Defect Resolution from Control and Organizational Perspective

Monika Gupta, Ashish Sureka, Srinivas Padmanabhuni

Indraprastha Institute of Information Technology - Delhi
New Delhi, India
Research Motivation

4 P’s of Software Engineering

- PEOPLE
- PROCESS
- PRODUCT
- PROJECT

PROCESS

Plan

Measure

Analyze

Improve
Process is an integral part of an organization

“If one cannot measure it, one cannot improve it”

11/4/2014
Research Motivation

Process Spanning Multiple Information Systems

Issue Reporting and Resolution Process

- Issue Reported in **Issue Tracking System**
- Patch Submitted in **Peer Code Review System** for Review
- Patch Committed in **Version Control System**

Process Mining

Multiple Software Repositories
Ex: ITS, PCR and VCS

http://www.chromium.org/
http://source.android.com/
Research Motivation

What is Process Mining?
Research Motivation

Source: http://www.processmining.org/

11/4/2014
Process Mining:

- Extract knowledge from event logs recorded by information systems [1].

- Event logs (e.g. transaction logs) with four fields:

<table>
<thead>
<tr>
<th>CaseID</th>
<th>Event</th>
<th>Timestamp</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

- Tools and framework for process mining:
  - ProM ¹ (Open Source)
  - Disco ² (Commercial)

---

Software Repositories:

• Artifacts generated by the tools during software evolution and *archived* for future reference.

• *Rich data* available.

• Uncover interesting and *actionable information* for process improvement.

For example:
Issue Tracking System (ITS)
Peer Code Review System (PCR)
Version Control System (VCS)
Integrate three standalone repositories, ITS, PCR and VCS using regular expression based approaches for simultaneous process mining.

Apply state-of-the-art algorithms to discover runtime process model for the end-to-end bug resolution.

Define framework to process mine event log data from control flow perspective and Organizational perspective.

Conduct a case-study on popular open-source project Google Chromium: Google ITS, Rietveld PCR, Subversion VCS
## Related Work

### 1. Process Mining Software Repositories:

<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>Repository</th>
<th>Objectives</th>
</tr>
</thead>
</table>

### 2. Simultaneous Process Mining Multiple Workflow Management Systems:

<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>Repository</th>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poncin <em>et al.</em> [5]</td>
<td>2011</td>
<td>aMSN and GCC bug repositories, mail archives, SVM</td>
<td>Combined different repositories for analysis using a prototype, FRASR. Role classification and Bug life cycle construction using ProM.</td>
</tr>
<tr>
<td><em>Song et al.</em> [6]</td>
<td>2008</td>
<td>Sample source data</td>
<td>Common event log from 5 different information systems for behavior pattern mining.</td>
</tr>
</tbody>
</table>
3. Application of Social Network Analysis for Mining Software Repositories:

- Dittrich et al. model a software project as network using VCS data and identify the key authors and subject matter experts [7].
- Sureka et al. mine ITS to derive collaboration network and investigate it for the purpose of risk and vulnerability analysis [8].
- Sarma et al. developed Tesseract, an environment to enable study of complex socio-technical relationships between different project entities [9].
- Wolf et al. examined task-based communication and collaboration in software teams and identify causes of build failure [10].

Research Contributions

- Integrate three software repositories: ITS, PCR and VCS
  - Simultaneous process mining
- Control flow perspective
  - Discover run-time process lifecycle
  - Analyze activities, transitions, events and unique traces
  - Identify bottlenecks
  - Detect anti-patterns
- Organizational perspective
  - Extract team based interaction patterns and visualize metrics such as:
    - Joint cases,
    - Joint activities,
    - Handover of work,
    - Subcontracting
- Case-study on ITS, PCR and VCS for Google Chromium project
Research framework for process mining three repositories from Control flow and Organizational perspective
Data Extraction and Integration

**ITS BUG ID**

- **Issue 88294**: "default printing settings are always "two sided"
  - Status: Fixed
  - Owner: kmadhusu@chromium.org
  - Closed: Jul 2011
  - Type: Bug
  - Platform: Windows
  - Severity: Minor
  - Summary: default printing settings are always "two sided"
  - Description:
    - Can't Edit
    - Can't Publish+Mail
    - Start Review
    - Created: 2 years, 10 months ago by kmadhusu
    - Modified: 2 years, 10 months ago
    - Reviewers: Lei Zhang, I have the power (commit-bot)
    - CC: kmadhusu@chromium.org
  - The following revision refers to: [Revision 92154](http://src.chromium.org/viewvc/chrome?view=rev&revision=92154)
  - Commited: [Patch Set 1](http://src.chromium.org/viewvc/chrome?view=rev&revision=92154)
  - Total comments: 2
  - Patch Set 2: Fixed nit

**PCR ISSUE ID**

- **Issue 7285039**: PrintPreview: [WIN] Fix the default duplex print setting. (Closed)
  - Description:
    - PrintPreview: [WIN] Fix the default duplex print setting.
    - TEST: Please refer to bug report.
    - Committed: [Patch Set 1](http://src.chromium.org/viewvc/chrome?view=rev&revision=92154)
  - Review URL: [http://codereview.chromium.org/7285039](http://codereview.chromium.org/7285039)

**PEER CODE REVIEW SYSTEM**

- codereview.chromium.org
- chromiumcodereview.appspot.com

**VERSION CONTROL SYSTEM**

- Revision 92154
  - Jump to revision: 92154
  - Author: kmadhusu@chromium.org
  - Date: Tue Jul 12 12:55:14 2011 UTC (2 years, 10 months ago)
  - Changed paths: 1
  - Log Message:
    - PrintPreview: [WIN] Fix the default duplex print setting.
    - BUG=88924
    - TEST: Please refer to bug report.

**PCR ISSUE ID**

11/4/2014
Data Extraction and Integration

Distribution of number of CR issues for ITS issues

PCR Issues with link to VCS

PCR ISSUES COMMITTED

19952

19422

Percentage of ITS issues

Number of CR issues

11/4/2014
Experimental Dataset Details for Google Chromium Browser*

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>First ITS issue creation date</td>
<td>1 July 2011</td>
</tr>
<tr>
<td>Reporting First issue</td>
<td>30 June 2012</td>
</tr>
<tr>
<td>Total extracted closed ITS issues</td>
<td>35,035</td>
</tr>
<tr>
<td>ITS issues with patches in PCR</td>
<td>10,110</td>
</tr>
<tr>
<td>Accessible ITS Issues with patches</td>
<td>10,000</td>
</tr>
<tr>
<td>Total PCR issues for above ITS issues</td>
<td>19,952</td>
</tr>
<tr>
<td>Unique PCR issues for above ITS issues</td>
<td>17,979</td>
</tr>
<tr>
<td>Total VCS commit</td>
<td>19,422</td>
</tr>
</tbody>
</table>

* Final log publicly available: https://github.com/Mining-repos-data/experimental_dataset
Research framework for process mining three repositories from Control flow and Organizational perspective
One of the major challenges is to produce a log conforming to input format of process mining tool.

1. Event log generation

2. State Minimization

3. Data cleaning
# Data Transformation

## Event log generation

**CaseID** | **Event** | **Timestamp** | **Resource**
---|---|---|---
ITS issue ID | Time of event | Actor |

## List of activities captured as state

### Information System
- **Issue Tracking System**
  - I_Creation
  - I_Open
  - I_Fixed
  - I_Invalid
  - I_Duplicate
  - I_WontFix
  - I_Verified
  - I_Closed

### Code Review System
- C_Creation
- P_Creation
- C_Reviewed

### Version Control System
- V_Commit

---

Research framework for process mining three repositories from Control flow and Organizational perspective
Process Discovery

"What is really happening?"

- Obtain *runtime process map*
- Use *Disco* miner based on proven framework of Fuzzy Miner

Control Flow Perspective

Identify activities and the order of their execution

I_Creation
I_Open
C_Creation
P_Creation
S_Commmit
C_Reviewed
I_Fixed
I_Closed
• Event log for 9744 cases (with <7 code review issues)

• Total 32 states

• High percentage (89%) are marked Fixed

• Code review issues are created sequentially for the same issue
Event Frequency

- Number of events per case in lifecycle ranges from 5 to 91
- Majority cases with 6 to 13 events

Unique Traces

- Complete sequence of events in the lifecycle
- Total 4453 traces
- 50% cases covered with only top 6% traces
Bottleneck Analysis

- Identify *most time consuming* transitions of process
- *Cause of delay* in end-to-end process

- **Significant delay in passing control from ITS to PCR and vice-versa.**
  For example:
  - Reporting first PCR issue after last ITS activity takes around 8 days
  - 8 days to mark an issue *Fixed* after the completion of patch review

- **Verification of *Fixed* issues taking around 50.4 days**
Anti-patterns


**Basic Anti-patterns:**
Restricted to *one information system* like loops, back-forth transitions.

*For example:* Multiple patch submission for same issue with few cases having >30 subsequent patches

**Composite Anti-patterns:**
Involves triggering and information flow between two or more information systems.

*For example:*

[![Diagram](chart.png)](chart.png)

---

Joint Cases

- Individuals working together on same case will have a stronger relation than the people rarely working together [12]

\[
\text{Metric } M1 = \frac{\sum_{i=1}^{N} (w_i)}{d*N}
\]

where,

\( w_i = \) Weighted degree of vertex \( i \) having degree \( d \), and

\( N = \) Number of vertices with degree \( d \)

- \( M1 \) measures \textbf{average strength of interaction} for vertices

Organizational Perspective

Joint Cases

- We calculate relative working together metric for performer $p1$ and $p2$ as:

$$p1 \bowtie p2 = \frac{\text{Number of cases } p1 \text{ and } p2 \text{ worked together}}{\text{Number of cases } p1 \text{ participated}}$$

<table>
<thead>
<tr>
<th>p1</th>
<th>p2</th>
<th>Cases together</th>
<th>$p1 \bowtie p2$</th>
<th>$p2 \bowtie p1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>358</td>
<td>0.094</td>
<td>0.053</td>
</tr>
<tr>
<td>C</td>
<td>B</td>
<td>337</td>
<td>0.093</td>
<td>0.050</td>
</tr>
<tr>
<td>D</td>
<td>B</td>
<td>314</td>
<td>0.080</td>
<td>0.047</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>256</td>
<td>0.070</td>
<td>0.067</td>
</tr>
<tr>
<td>D</td>
<td>C</td>
<td>249</td>
<td>0.063</td>
<td>0.068</td>
</tr>
</tbody>
</table>
Organizational Perspective

Joint Activities

*Relation between the actors and activities they perform*

- Total unique performer: 2160
- Total ITS issue reporters: 1236
- Total patch reviewers: 915
- **Specialist and generalist identification**
- **Efficient role assignment**

Social network representing relation between activity and performers
Handover of work

- An activity performed by performer $p_1$ is consecutively followed by an activity executed by performer $p_2$
- Two performers are related if case is passed from one performer to another
- Only direct succession
- Total unique handovers: 988
- Self loops: 429
  - Same performer performs subsequent activities

High outdegree (big size), High indegree (dark color)
Subcontracting of work

- Cases with only one activity in-between two activities performed by the same performer, direct subcontract
- Two performers are related if direct subcontract
- Total unique subcontract instances: 577
- Self loops
  - More than 2 subsequent activities
Conclusion

• Runtime *process map discovered* issues with lifecycle spanning three IS
• Analyzed distribution of activities, transitions, events and unique traces
• Resolution process efficient with high chances of issues getting *Fixed*
• Basic and Composite *anti-patterns* like loops, and information flow detected
• *Bottlenecks* are identified such as control transfer between ITS and PCR
• More social performers are more *active*
• Joint activities helps to identify *generalists and specialists*
• Same performer performs *multiple subsequent activities*
Acknowledgement

• Supported by Prime Minister’s fellowship for PhD Students (awarded to first author) by SERB, CII and industry sponsor, Infosys Limited.

• Thanks Dr. Srinivas Padmanabhuni (industry mentor) for his valuable inputs.
THANK YOU!

QUESTIONS ...
References

1. https://bugzilla.mozilla.org


