Understanding software understanding

Andrew J. Ko

Information School
Computer Science & Engineering
University of Washington
We live in a time of incredible increases in productivity and discovery. Most of this is due to software.
Software fails

Create a Marketplace account

⚠️ Important: Your account couldn't be created at this time. The system is unavailable.

TRY AGAIN
57,000 software failures in the news stories since 1980

Ko et al. (2013) Thirty Years of Software Problems in the News, in review
The systems we create are incredibly difficult to understand

6-12 months to onboard a new developer to even basic productivity

Novice Software Developers All Over Again. A. Begel, B. Simon, ICER 2008

Months to understand why a stock market crashes, a Prius suddenly accelerates, or a plane nosedives
I’ve spent the past 10 years trying to understand why software is so difficult to understand and inventing technologies that make it (somewhat) easier.
Outline

dev

team

contributors

users
What makes software difficult for an individual developer to understand?

An Exploratory Study of How Developers Seek, Relate, and Collect Relevant Information during Software Maintenance Tasks (2006)
10 experienced Java programmers

1 program to maintain Eclipse 2.0 IDE

5 tasks, do in any order in 70 min

$10 for each task completed

Eclipse 2.0 IDE

the internet (Google) for documentation, examples, etc.
The Paint Application

508 lines of Java

Draw, erase, clear and undo colored strokes
We transcribed

User interface actions
Dependency navigations
  *Use to definition*
  *Method to invocation*
  *Class to superclass*

Reading code
Switching environments
Testing *Paint*
etc.
Division of Labor

- Reading code: 22%
- Editing code: 20%
- Navigating dependencies: 16%
- Searching for task-relevant words in code: 13%
- Reading task descriptions: 13%
- Switching applications: 5%
- Reading the Java API: 6%
- Testing Paint: 3%
Form working set of task-relevant code

Navigate dependencies in working set

Modify code in working set

task started → complete
Forming Working Sets

How does ____ work?

Searched for task-relevant words

Only 50% of searches led to relevant code :(

Why did(n’t) ____ happen?

Guessed which parts of the implementation might be causing the unexpected behavior

88% of guesses were incorrect

Developers struggled to map the observed behavior with the code that caused it.
Navigating within a Working Set

Navigated ~65 dependencies over 70 minutes

58% direct
declaration of . . .
use of . . .
called by . . .
definition of . . .

42% indirect
“The method that computes the value that is passed to this method and used in this expression”

Many (but not all) supported by Eclipse commands

Supported only by scroll bars, package explorer, tabs, and find
Information Foraging Theory (Burnett et al.)

Developers follow “scents” in source code, error messages, and online resources

Verbalizations of what scents they are following are highly predictive of where they will navigate to

**Identifiers** in source code are the single most important factor in making code comprehensible


Mylyn
Mylyn: a degree-of-interest model for IDEs Mik Kersten, Gail C Murphy. Proceedings of the 4th international conference on Aspect-oriented software development
Set-based editor

<table>
<thead>
<tr>
<th>Tasks</th>
<th>+</th>
<th>-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line tool</td>
<td>Thickness</td>
<td>Painting bug</td>
</tr>
</tbody>
</table>

Docked Info
- JSlider-class — how do I give it a range?
- SliderDemo.java from "How to Use Sliders"

Recommendations
- The method setThickness() in PaintObject Constructor seems related to this code.

```
PaintWindow.java
private JPanel rPanel, gPanel, bPanel;
private JSlider rSlider, gSlider, bSlider;
...

private ChangeListener colorChangeListener = new ChangeListener()
    public void stateChanged(ChangeEvent changeEvent) {
        objectConstructor.setColor(new Color(rSlider.getValue(),
            gSlider.getValue(), bSlider.getValue()));
    }
...

public PaintWindow(int initialWidth, int initialHeight) {
    ...
    rPanel = new JPanel(new FlowLayout());
    rPanel.setOpaque(false);
    rPanel.add(new JLabel("Red"));
    rSlider = new JSlider(0, 255, 0);
    rSlider.setOpaque(false);
    rSlider.addChangeListener(colorChangeListener);
    rPanel.add(rSlider);
    // Here's my code. I'm going to add a thickness slider
    tPanel = new JPanel(new FlowLayout());
    tPanel.add(new JLabel("Red"));
    tSlider = new JSlider(0, 255, 0);
    tSlider.setOpaque(false);
    tSlider.addChangeListener(colorChangeListener);
    tPanel.add(tSlider);
    ...
    objectConstructor = new PaintObjectConstructor(this);
    objectConstructor.setClass(PencilPaint.class);
    objectConstructor.setColor(new Color(0, 255, 0));
    objectConstructor.setThickness(1);
    ...
```

```
EraserPaint.java
public void setThickness(int thickness) {
    this.thickness = 25;
}
```

```
PaintObject.java
protected int thickness;
protected Color color;

public void setColor(Color color) {
    this.color = color;
}
public void setThickness(int thickness) {
    this.thickness = thickness;
}
```

```
ChangeListener doc (Java 2 Platform 1.4)
public interface ChangeListenerextends EventListener

Defines an object which listens for ChangeEvents.

Method Summary
void stateChanged(ChangeEvent e)
Invoked when the target of the listener has changed its state.
```

```
PencilPaint.java
public Rectangle getBoundingBox() {
    ...
    int x = (int)points[pointIndex].getX();
    int y = (int)points[pointIndex].getY();
    if (x - thickness / 2 < minX) minX = x - thickness / 2;
    else if (x + thickness / 2 > maxX) maxX = x + thickness / 2;
    if (y - thickness / 2 < minY) minY = y - thickness / 2;
    else if (y + thickness / 2 > maxY) maxY = y + thickness / 2;
    ...
```
Jasper

Code Bubbles

Debugger Canvas
Visual Studio Dev Labs, Rob DeLine
What makes software difficult for an individual developer to understand?

Developers must painstakingly build a mapping between input, output, and code, searching for causal relationships across program elements and executions that are highly disparate in time and space.
What makes software difficult for a team to understand?

Emailed 250 randomly selected Microsoft developers. Of these, 55 responded and we observed 17.

Developers varied by

- time with team
- time of observation
- developer/manager
- customer of code
- phase of development
- direction of team dependencies
Field observation methodology

25 hours of coding and bug fixing, in the role of “new hires”

357 pages of handwritten notes (other recording was too invasive)

4,231 events in an spreadsheet
# 10 most common needs

<table>
<thead>
<tr>
<th>Query</th>
<th># devs</th>
<th># of seeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>what have my coworkers been doing?</td>
<td>15</td>
<td>43</td>
</tr>
<tr>
<td>what code caused this program state?</td>
<td>11</td>
<td>31</td>
</tr>
<tr>
<td>have resources I depend on changed?</td>
<td>10</td>
<td>25</td>
</tr>
<tr>
<td>what code <em>could</em> have caused this behavior?</td>
<td>9</td>
<td>22</td>
</tr>
<tr>
<td>how do I use this data structure or function?</td>
<td>9</td>
<td>21</td>
</tr>
<tr>
<td>did I make any mistakes?</td>
<td>9</td>
<td>46</td>
</tr>
<tr>
<td>what is the program <em>supposed</em> to do?</td>
<td>7</td>
<td>28</td>
</tr>
<tr>
<td>in what situations does this failure occur?</td>
<td>7</td>
<td>27</td>
</tr>
<tr>
<td>is this problem worth fixing?</td>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td>why was this code implemented this way?</td>
<td>6</td>
<td>19</td>
</tr>
<tr>
<td>what’s statically related to this code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what are the implications of this change?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what does the failure look like?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>how can I coordinate this with the other code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>did I follow my team’s conventions?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what is the purpose of this code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is this a legitimate problem?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### 6 most common unsatisfied

<table>
<thead>
<tr>
<th>Question</th>
<th>%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>what have my coworkers been doing?</td>
<td>61%</td>
<td>21 min</td>
</tr>
<tr>
<td>what code caused this program state?</td>
<td>24%</td>
<td>9 min</td>
</tr>
<tr>
<td>have resources I depend on changed?</td>
<td>36%</td>
<td>17 min</td>
</tr>
<tr>
<td>what code <em>could</em> have caused this behavior?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>how do I use this data structure of function?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>did I make any mistakes?</td>
<td>15%</td>
<td>21 min</td>
</tr>
<tr>
<td>what is the program <em>supposed</em> to do?</td>
<td>41%</td>
<td>49 min</td>
</tr>
<tr>
<td>in what situations does this failure occur?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is this problem worth fixing?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>why was this code implemented this way?</td>
<td>44%</td>
<td>21 min</td>
</tr>
<tr>
<td>what’s statically related to this code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what are the implications of this change?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what does the failure look like?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>how can I coordinate this with the other code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>did I follow my team’s conventions?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what is the purpose of this code?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>is this a legitimate problem?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what changes are part of this submission?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>how difficult will this problem be to fix?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>what information was relevant to my task?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
how have my resources changed?

10 seeks ≤9 minutes 24% unsatisfied

How are my builds doing?
Is my test suite done running?
What code has changed?
Has that bug been fixed yet?

A developer is blocked on a task, and decides to figure out how code and other resources have been changing.

sources use tools, check e-mail, ask a coworker...
in what situations does this occur?

7 seeks ≤49 minutes 41% unsatisfied

What reproduction step is missing?
What’s different about my configuration and the report author’s?

A developer gets a bug report and needs to reproduce the problem, but the reproduction steps do not work.

sources read the reproduction steps, talk to the report author, start twiddling bits, give up...
what could have caused this behavior?

9 seeks \( \leq 17 \text{ minutes} \quad 36\% \text{ unsatisfied} \)

Could it be widget X or perhaps sprocket Z? Maybe its getting into bad state Y...

A developer has just started trying to understand a failure, but needs to guess what’s causing it in order to use a debugger.

sources guess; look for hints in the report; ask a coworker; make it not fail and find the difference.
what actually caused this state?

11 seeks  ≤21 minutes  61% unsatisfied

Why did this variable get set to 5?
How did we reach this method?
Why are we in this thread?

A developer is trying to fix a bug, but to know what to fix, has to find the parts of the program that were involved.

sources set up the debugger, set some breakpoints and try to trace it back manually...
why was this code written this way?

6 seeks  \leq 21 minutes  44\% unsatisfied

What was the author’s intent?
Was this overlooked intentionally or by mistake?

A developer figures out what’s causing a failure, but to avoid breaking other dependencies, needs to know why the code was written the way it is:

sources it’s written down in a report or a notebook somewhere, or it’s in the author’s head.
what’s the program supposed to do?

What did we decide at that last design meeting?
Did I write this down somewhere?
Is the intended behavior in the design spec?

A developer starts on a bug report, figures out what the program is doing, but doesn’t know the intended behavior.

sources is there a spec I can trust? I’ll ask the PM; we might have to have a meeting about this...
What makes software difficult for a team to understand?

Difficulty observing program execution, and:

1. Knowing precisely how a program was being executed in the wild
2. Knowing why code was written in a particular way
3. Having confidence that the intended behavior is the right behavior
Does having “many eyes” help with these problems?

Obtained all 496,766 reports except for private security patches since 2009

152,877 unique e-mail addresses

64% addresses only authored, attached to, or commented on 1 report

Studied 15 years of Mozilla bug reports
who was behind these 152,877 addresses?

1% **CORE** developers, drivers, super reviewers, module owners, peers
who was behind these 152,877 addresses?

1% **CORE** developers, drivers, super reviewers, module owners, peers

1% **ACTIVE** developers assigned bug reports
who was behind these 152,877 addresses?

1% **CORE** developers, drivers, super reviewers, module owners, peers

1% **ACTIVE** developers assigned bug reports

80% **REPORTERS**

reported and commented on bug reports

responsible for 54% of reports
who was behind these 152,877 addresses?

1% **CORE** developers, drivers, super reviewers, module owners, peers

1% **ACTIVE** developers assigned bug reports

80% **REPORTERS** reported and commented on bug reports

18% **USERS** only commented on bug reports

who was behind these 152,877 addresses?
# of active contributors by type, per 6 month period

Reporters and users fluctuate spike before a release.
runtime violations

errors, warnings, assertion violations, crashes, hangs, and **language-defined invalid states**

“...scary deadlock assertions exiting mozilla after referencing nsInstallTrigger...”
specification violations

an agreed upon **functional requirement** among the application developers

“There's an incorrectly placed PR_MAX in the code for pref width distribution of colspanning cells.”

application language
standards violations

**industry-wide** functional specifications, reaching beyond the application’s developer community

“'codebase' attribute of the HTML 4.0 OBJECT element is not supported...”
violations of a reporter expectations

a reporter’s **personal perspective** about what the system should do

“Every time I Sort By Name by Bookmarks Firefox sorts and closes my Bookmark menu... Why does it do this??”
violation of a community’s expectations

a reporter’s belief about a “typical” user’s expectations

“The preference to not show the tab bar when only one tab is open could be set to false by default. This would at least alert a new user to the possibility that tabs exist) The old tabbed browsing preferences could be returned.”
violation of genre conventions

inconsistency with the behavior of a similar application

“Firefox does not limit the slideshow horizontal size to the window width. The same source works correctly in IE.”
inconsistency with prior behavior

community expectation that behavior of previous versions would be preserved

“The latest version of Firefox only imports one certificate from each file. I used to import all certificates previously.”
most **REPORTER** reports were duplicate, worksforme, or invalid
were the duplicates useful?

73% of REPORTERs’ duplicates referred to fixed reports

70% of REPORTERs’ duplicates referred to issues known for > 1 month

66% of REPORTER duplicates of fixed reports were created after a patch was attached

most REPORTER reports identified issues that were

already known

already patched

invalid, wontfix, worksforme
Does having “many eyes” help with program understanding?

Not really. Finding and fixing bugs requires expertise and information that most contributors simply did not have.

Also didn’t teach devs about whether they were building the right thing
If not through openness, how can developers gain confidence that they’re building the right software?

How does one design software for 1 billion users?

Summer 2011
17 one hour interviews with senior engineers, managers, testers, designers, and data scientists

Asked about

- Process for making design decisions when launching a new product or feature
- How they gather information and insights about users
- How user information was used in design and engineering efforts
Engineering is the easy part

“... It’s just not that hard to build a particular functionality... we try to do UX and iterate on various designs versus just getting something out there, make sure it’s reasonable, and then iterate on the design based on how people are using it... not just the design but the how the whole interaction flows...
"...design is hard...really hard. Just doing our best with very smart people, we screw up plenty... design and consistency takes time...we really work to make each experience as good as it can be... unfortunately, design and simplicity are often after thoughts..."
You know we’re trying to make a universal product and we think some very core pieces of Facebook are universal, so we feel like there is a solution that works for everybody for the very key things... like how you manage identity... the profile should look basically the same for everyone... but once you get farther away from the core products, it’s not necessarily obvious to us that there is a magic way that a feature can work and everyone can find value in it...
The user is not like me

“...when I started here the demographic and socio-economic makeup of the company was very similar to the user base... our own feelings were an excellent proxy for users... what we thought was cool, many users would agree was cool... we’re now many moons from that time... when we make a new photo upload button, it needs to be equally intuitive to a 90 year Mongolian grandmother to a 14 year old Brazilian soccer player...
I have watched a series of live user studies where we had users in the room... we could watch how they move their mouse... it was eye-opening to see how many of your assumptions are wrong and to see all the things you take for granted because you’re an engineer... To watch somebody who just has you know who is nowhere near there and not see a link or a button or start typing in the wrong place... I’m talking about totally smart people who can’t upload a photo because they’re not in the mind-set of I know there’s a way to do it and I just need to find a button...
The limits and lack of data

“I would want data but it’s pretty hard to always get data, particularly for this look and feel kind of stuff... it’s slow right... if I make a change, it may change your impression of the product over months perhaps... not a short period of time... so it’s hard to get data that corroborates that..."

“It’s certainly the case that debates either before or after product launches, people throw out these hypotheses and a lot of time the data just isn’t there to back it up or and it would take a lot of instrumentation to do it..."
Why is software is so difficult to understand?

dev  team  contributors  users
Why is software so difficult to understand?
Why is software so difficult to understand?

1) Programs are difficult to read
Why is software so difficult to understand?

1) Programs are difficult to read

2) Program execution is difficult to observe
Why is software so difficult to understand?

1) Programs are difficult to read

2) Program execution is difficult to observe

3) Program execution by millions of end users’ is even more difficult to observe
Why is software so difficult to understand?

1) Programs are difficult to read

2) Program execution is difficult to observe

3) Program execution by millions of end users’ is even more difficult to observe

4) Even if reading code and observing it execute were easy, knowing whether an execution is “good” is not scalable
We need more facile ways of controlling and observing program execution
e.g., Timelapse seamlessly captures precise replayable recordings of web applications
We need tools that make causality immediately apparent in the IDE
e.g., Cleanroom, FeedLack, and Whyline all help developers directly observe the link between source and its impact on program output


We need easier and more scalable ways for end users to precisely report when something is confusing or broken.

e.g., LemonAid allows to seek answers to questions by selecting parts of the user interface, allowing developers to see exactly what part of an interface is confusing.

Questions?