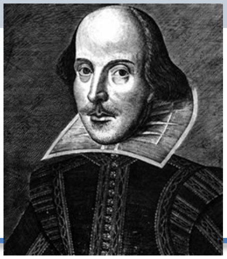
21st IEEE International Conference on Program Comprehension

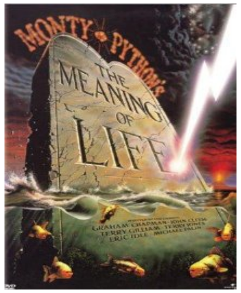
Improving Program Comprehension by Answering Questions

Brad A. Myers
Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
http://www.cs.cmu.edu/~bam
bam@cs.cmu.edu

Questions



- "To be or not to be?"
- "What is the meaning of life?"
- "Ask not what your country can do for you – ask what you can do for your country."
- "Which outfit should I wear?"
- "What does this code do?" "What just happened? …"

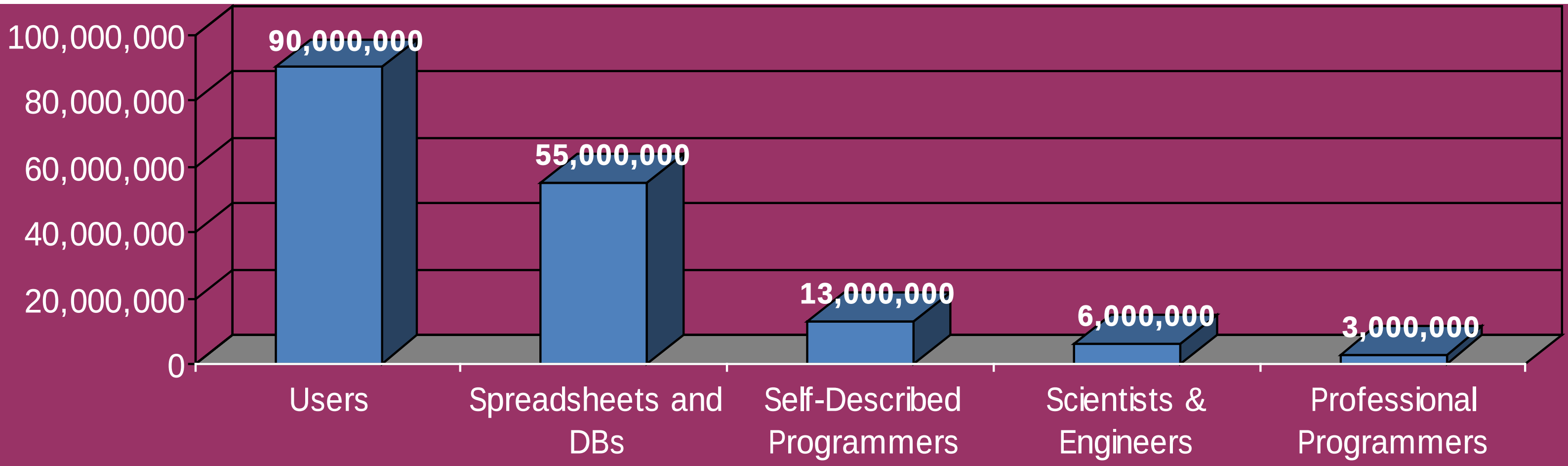


Natural Programming Project

- Researching better tools for programmers since 1978
- Natural Programming project started in 1995
- Make programming easier and more correct by making it more *natural*
  - Closer to the way that people think about their problems and solving their tasks
- Methodology – human-centered approach
  - Perform *studies* to inform design
    - Provide new knowledge about what people do and need to know
  - Guide the designs from the data
    - Design of *languages*, *environments* and *documentation*
  - Iteratively evaluate and improve
- Target novice, expert and end-user programmers

End User Programming

- People whose primary job is *not* programming
- In 2012, in USA at work: *— Scaffidi, Shaw and Myers 2005*
  - 3 million professional programmers
  - 6 million scientists & engineers
  - 13 million will describe themselves as programmers
  - 55 million will use spreadsheets or databases at work (and therefore may potentially program)
  - 90 million computer users at work in US
- We should make better tools for all of these people!

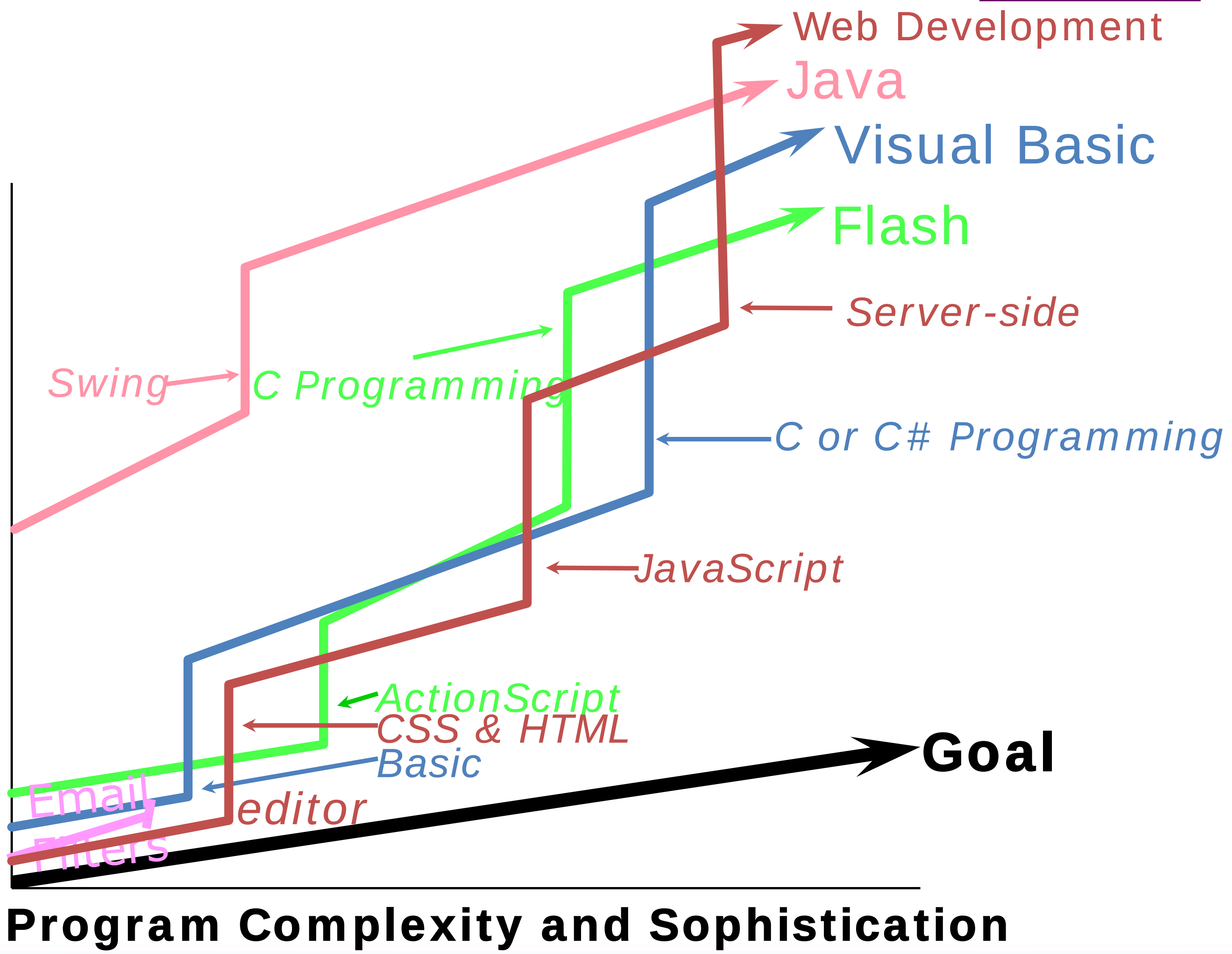


Debugging

- Study commissioned by NIST USA (2002) of 14 software vendors
  - Software errors cost ~$60 billion annually
  - Software engineers spend 70-80% of time testing and debugging
  - Time for 1 developer to fix 1 bug was ~17.4 hours
- Current debugging techniques *same as for last 70 years*
  - Same for end-user and professional environments

Goal: Gentle Slope Systems

High Ceiling



Low Threshold

Improve Developer Experience

- Use human centered approaches to:
  - Find out what developers *need to know*
  - Understand developers’ *barriers* that cause *wasted time*
  - Make developers *more effective*
  - *Reduce errors* in their understanding and in the resulting code
  - Insure that developer tools are *useful*

Why Would Being Natural be Good?

- Programmers are People Too
  - Take the human into account
- Language should be close to user's plan
  - "Programming is the process of transforming a mental plan into one that is compatible with the computer." *— Jean-Michel Hoc*
- *Closeness of mapping*
  - "The closer the programming world is to the problem world, the easier the problem-solving ought to be.... Conventional textual languages are a long way from that goal." *— Green and Petre*

Hard to understand

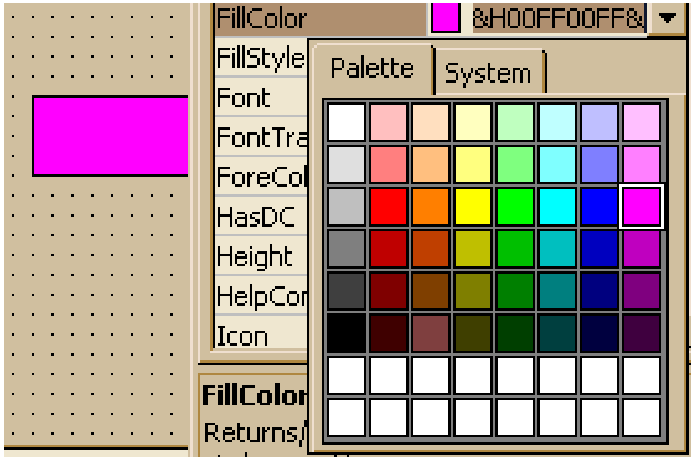
```
drawImage(img,10,20,30,40,11,21,31,41,red,obs);
```

– *8 ints*

```
item = new Item("C12","S123","S123","P123",
  "I123","","1,"2","3”,"4","5",1.0d,10.0d);
```

– *11 strings*

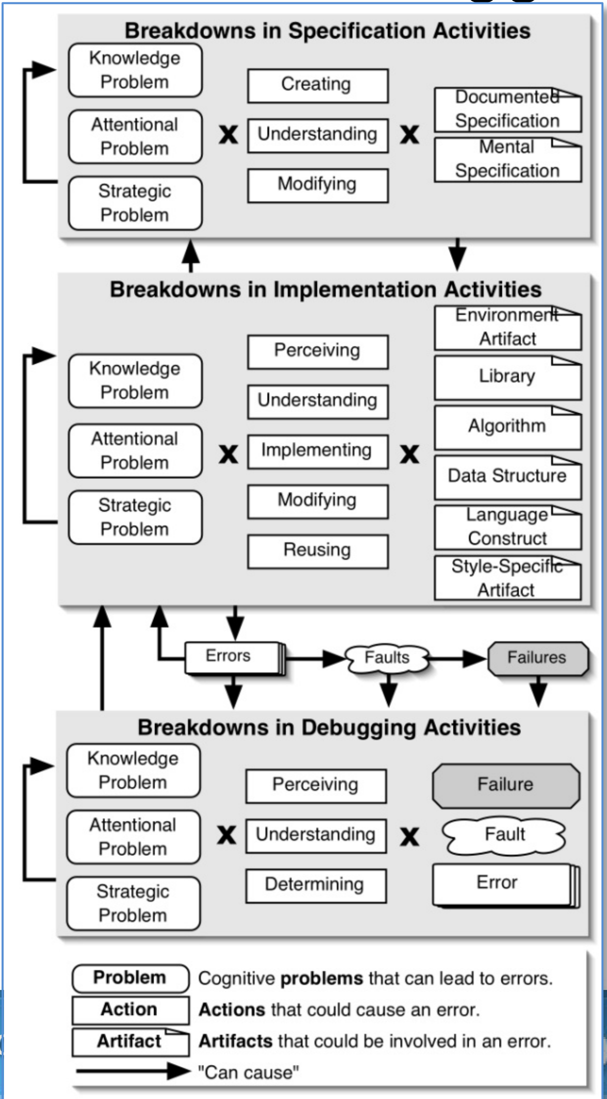
```
Let Shape1.FillColor
  = &H00FF00FF&
```

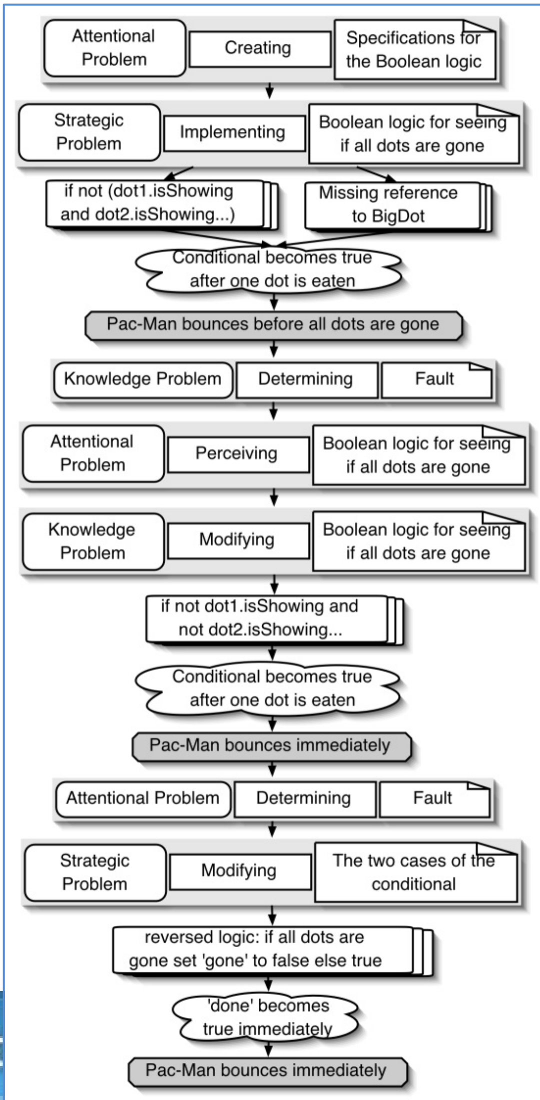

FillColor &H00FF00FF&

Study of Errors

- Study of novice errors and debugging
- Developed a model of problems and errors
  - Problems causing other problems

(EUP'03)





Attentional Problem
Creating
Specifications for the Boolean logic
Strategic Problem
Implementing
Boolean logic for seeing if all dots are gone
if not (dot1.isShowing and dot2.isShowing...)
Missing reference to BigDot
Conditional becomes true after one dot is eaten
Pac-Man bounces before all dots are gone
Knowledge Problem
Determining
Fault
Attentional Problem
Perceiving
Boolean logic for seeing if all dots are gone
Knowledge Problem
Modifying
Boolean logic for seeing if all dots are gone
if not dot1.isShowing and not dot2.isShowing...
Conditional becomes true after one dot is eaten
Pac-Man bounces immediately
Attentional Problem
Determining
Fault
Strategic Problem
Modifying
The two cases of the conditional
reversed logic: if all dots are gone set 'gone' to false else true
'done' becomes true immediately
Pac-Man bounces immediately

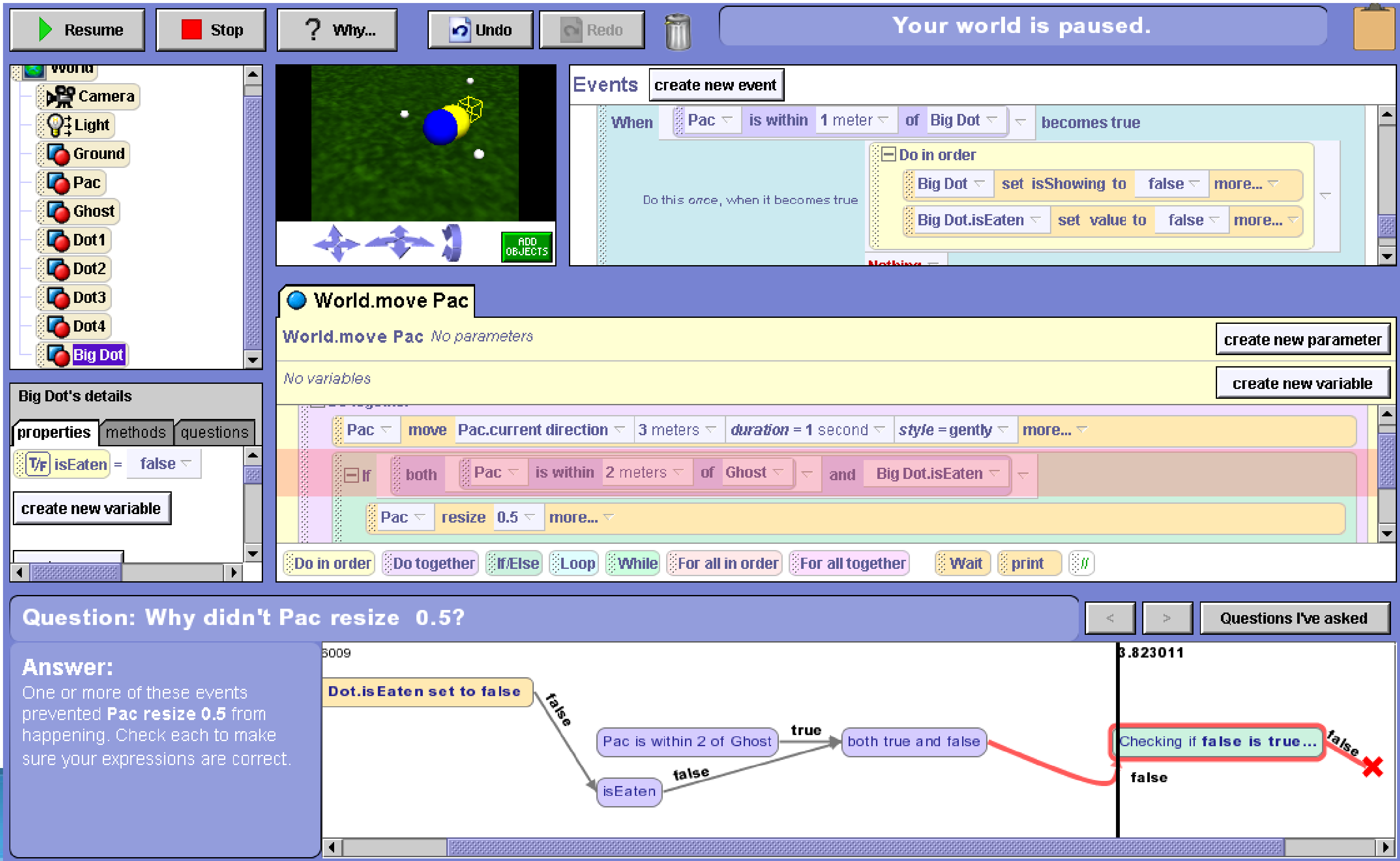
Study of Errors

- All of the observed debugging problems could be addressed by "Why" questions
  - 32% were "Why did"; 68% were "Why didn't"
- Current debugging techniques require user to *guess* where bug is or where to look
  - Most of initial guesses are *wrong*, even for experts

Original Design: Whyline for Alice

- Andy Ko, PhD 2008
- Answers as an elaborate visualization of control and dataflow

(CHI'04)



When Pac is within 1 meter of Big Dot becomes true

Do this once, when it becomes true

Do in order
Big Dot set isShowing to false more...
Big Dot.isEaten set value to false more...

World.move Pac *No parameters*

No variables

Pac move Pac.current direction 3 meters *duration* = 1 second *style* = gently more...
If both Pac is within 2 meters of Ghost and Big Dot.isEaten
Pac resize 0.5 more...

isEaten = false

Question: Why didn't Pac resize 0.5?

Answer:
One or more of these events prevented Pac resize 0.5 from happening. Check each to make sure your expressions are correct.

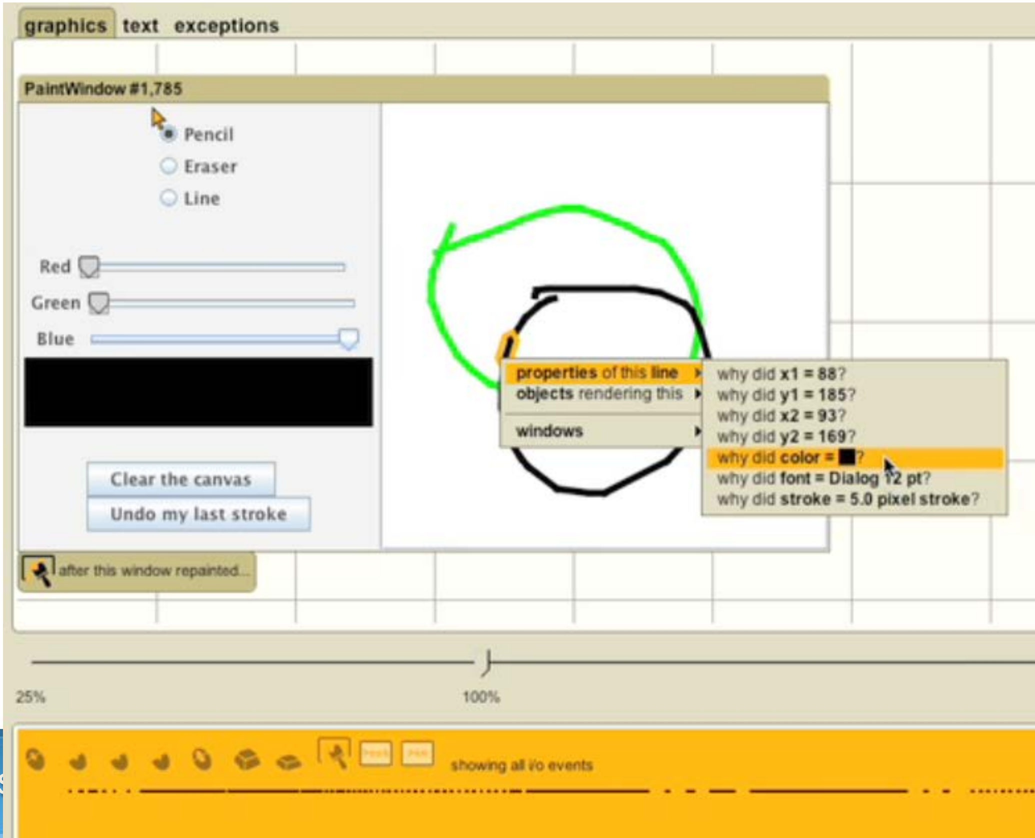
Dot.isEaten set to false → false → isEaten → false → both true and false
Pac is within 2 of Ghost → true → both true and false
both true and false → Checking if false is true ... → false
3.823011

Whyline for Java

- New algorithms
- New user interface design
  - Visualization primarily as navigation aide
  - Importance of search
- Not sufficient to just scrub through time

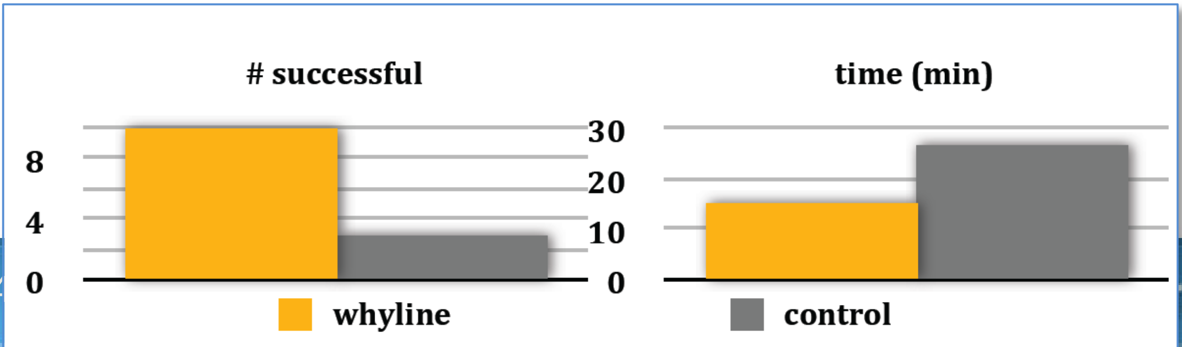
(ICSE'2008)

1:27



Whyline

- Whyline = Workspace that Helps You Link Instructions to Numbers and Events
- Initial study:
  - Whyline with novices outperformed experts with Eclipse
  - Factor of 2.5 times faster
    - (p < .05, Wilcoxon rank sums test)
- Formal study:
  - Experts attempting 2 difficult tasks
  - Whyline over 3 times as successful, in ½ of the time

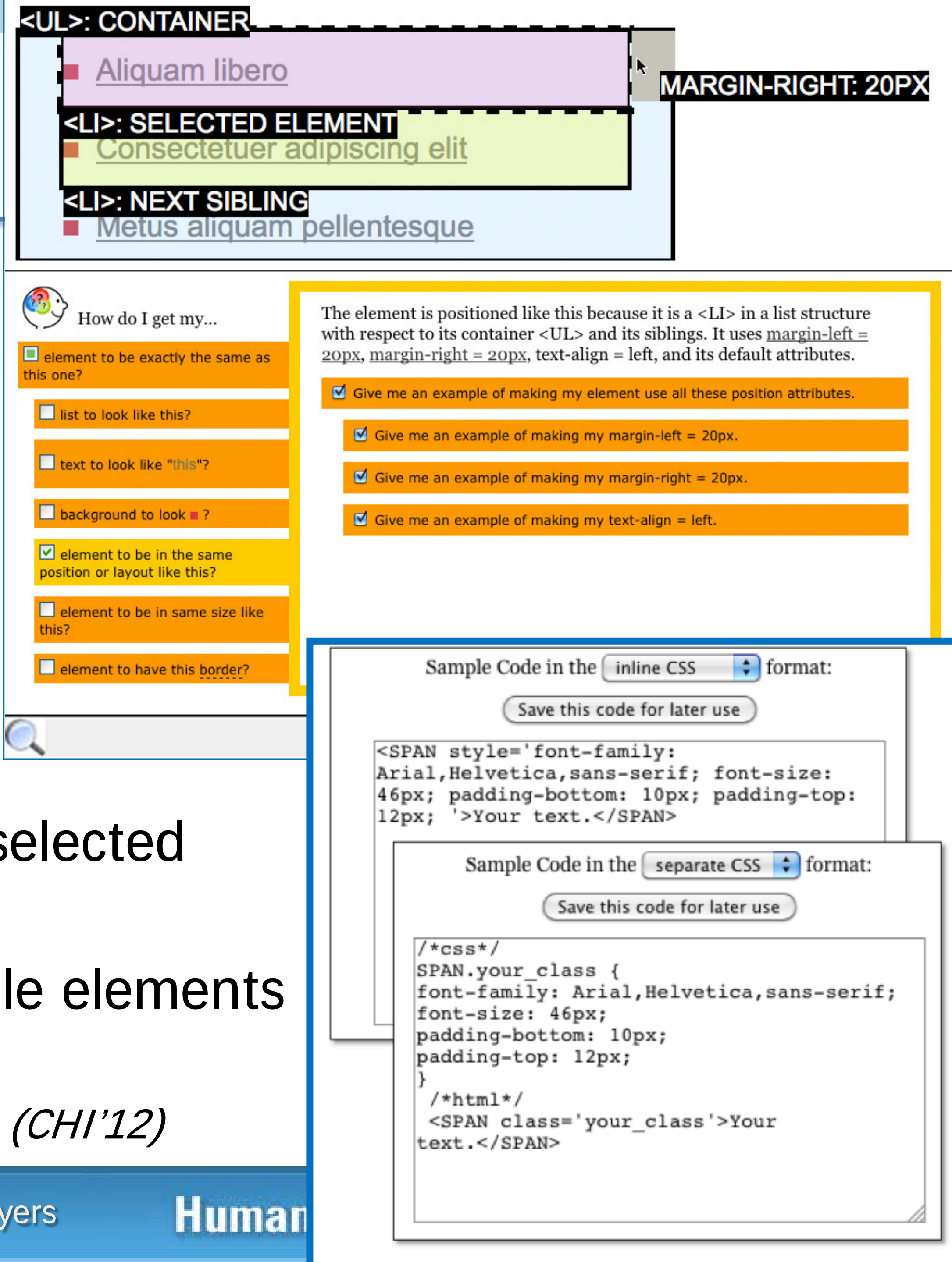


(CHI'09)

WebCrystal

- Investigate CSS and HTML responsible for example behaviors
- Navigate around HTML hierarchy
- Ask "how-do-I" questions about look, position and behavior
- Generates code in user-selected format
- Combine code for multiple elements

(CHI'12)



<UL>: CONTAINER
Aliquam libero
MARGIN-RIGHT: 20PX
<LI>: SELECTED ELEMENT
Consectetuer adipiscing elit
<LI>: NEXT SIBLING
Metus aliquam pellentesque
How do I get my...
element to be exactly the same as this one?
list to look like this?
text to look like "this"?
background to look ?
element to be in the same position or layout like this?
element to be in same size like this?
element to have this border?
Give me an example of making my element use all these position attributes.
Give me an example of making my margin-left = 20px.
Give me an example of making my margin-right = 20px.
Give me an example of making my text-align = left.
Save this code for later use

The element is positioned like this because it is a <LI> in a list structure with respect to its container <UL> and its siblings. It uses margin-left = 20px, margin-right = 20px, text-align = left, and its default attributes.

Sample Code in the inline CSS format:

```
<SPAN style='font-family:
Arial,Helvetica,sans-serif; font-size:
46px; padding-bottom: 10px; padding-top:
12px; '>Your text.</SPAN>
```

Sample Code in the separate CSS format:

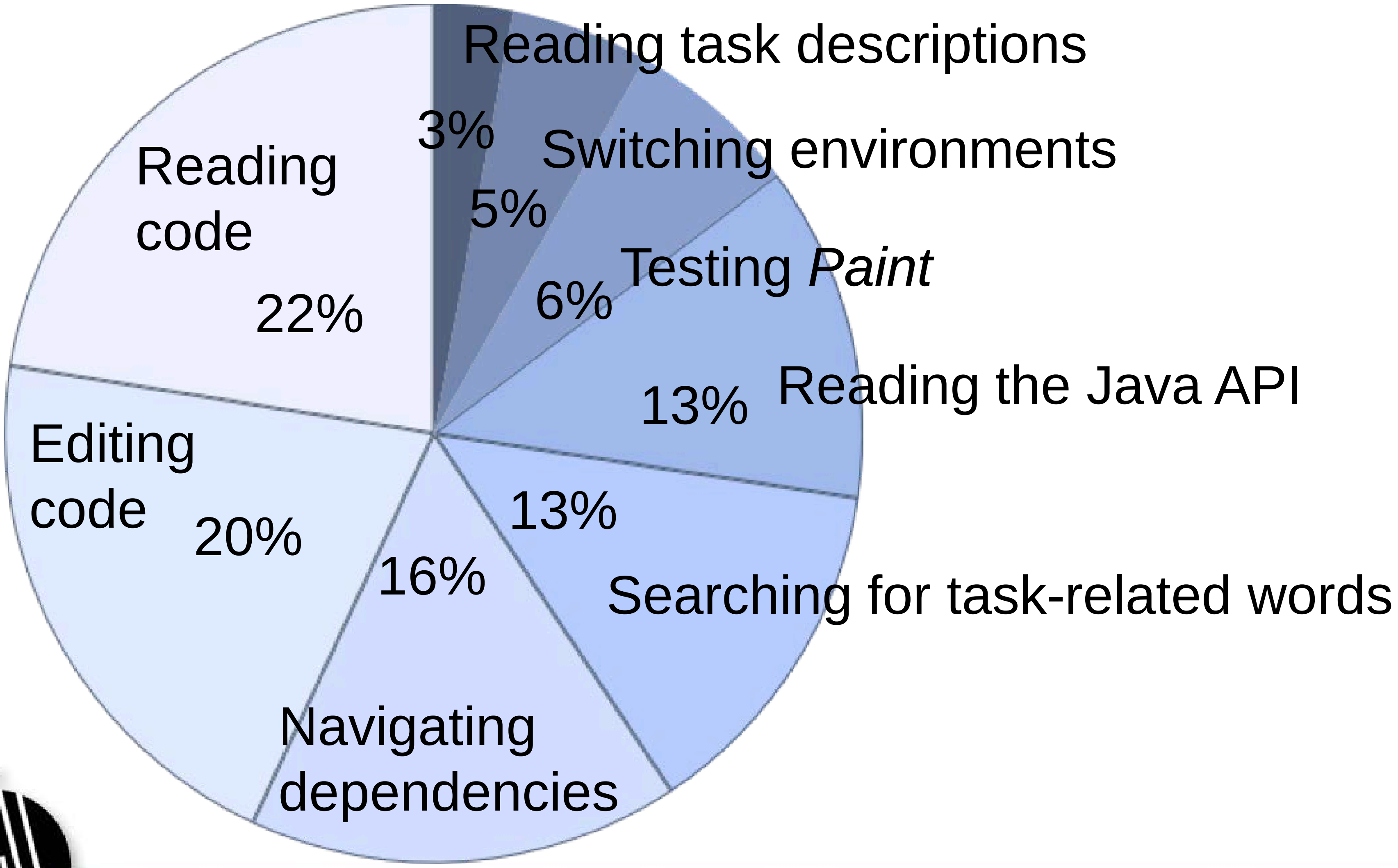
```
/*css*/
SPAN.your_class {
font-family: Arial,Helvetica,sans-serif;
font-size: 46px;
padding-bottom: 10px;
padding-top: 12px;
}
 /*html*/
 <SPAN class='your_class'>Your
text.</SPAN>
```

Study of Design Requirements for Maintenance-Oriented IDEs

- Studied expert use of Java Eclipse IDE in a lab setting (2004-2006)
- Focus on day-to-day maintenance tasks such as bug repairs and feature enhancements
- Lab study with detailed analysis
- Rich dataset → multiple papers

(ICSE'05)

Time Spent on Different Activities



Times for Bottlenecks

- Each instance of an interactive bottleneck cost only a few seconds, but . . .

Interactive Bottleneck	Overall Cost
Navigating to fragment in *same* file (*via scrolling*)	~ 11 minutes
Navigating to fragment in *different* file (*via tabs and explorer*)	~ 7 minutes
Recovering working set after returning to a task	~ 1 minute
Total Costs	~19 minutes

- = **35 %** **of uninterrupted work time!**

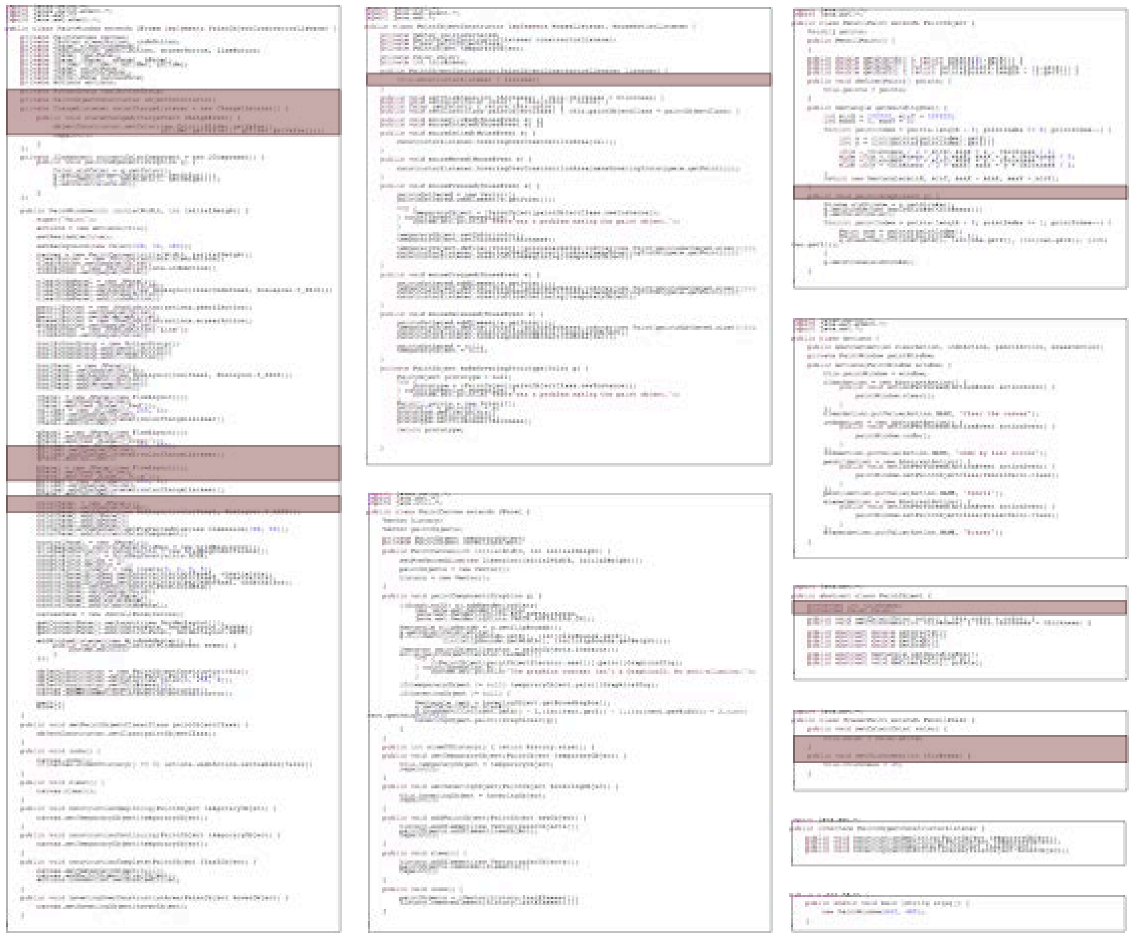
Forming **Working Sets**

- *How does ____ work?*
  - Searched for seemingly task-relevant words
  - Only 50% of searches led to relevant code
- *Why did(n't) ____ happen?*
  - Formed hypotheses about potential causes of unexpected behavior
  - 88% of hypotheses were false

Programmers had trouble relating the behavior they saw (or didn't see) to the code responsible for it.

A Programmer's Working Set

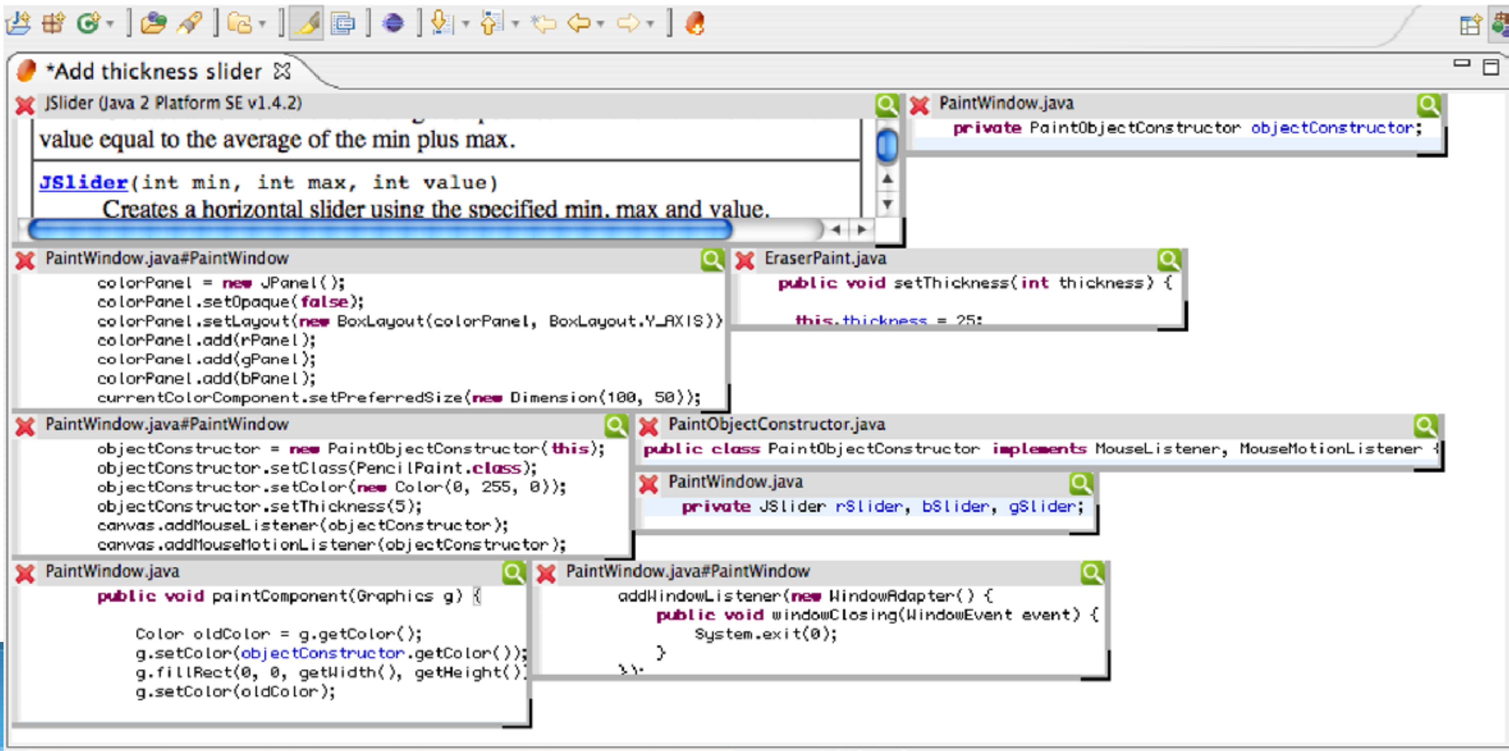
- A collection of task-relevant code fragments
- In modern software development, dependencies are distributed and non-local



Jasper: Working Set Tool

- Jasper = Java Aid with Sets of Pertinent Elements for Recall
- Allow programmers to grab arbitrary fragments of code to represent working sets
  - Allow programmers to view in one place, one screen
- Influenced Reiss *et. al's* Code Bubbles and DeLine's (Microsoft) Debugger Canvas in Visual Studio

(ETX'06)



*Add thickness slider

JSlider (Java 2 Platform SE v1.4.2)

value equal to the average of the min plus max.

```
JSlider(int min, int max, int value)
```
Creates a horizontal slider using the specified min, max and value.

PaintWindow.java
```
private PaintObjectConstructor objectConstructor;
```

PaintWindow.java#PaintWindow
```
colorPanel = new JPanel();
colorPanel.setOpaque(false);
colorPanel.setLayout(new BoxLayout(colorPanel, BoxLayout.Y_AXIS))
colorPanel.add(rPanel);
colorPanel.add(gPanel);
colorPanel.add(bPanel);
currentColorComponent.setPreferredSize(new Dimension(100, 50));
```

EraserPaint.java
```
public void setThickness(int thickness) {

  this.thickness = 25;
```

PaintWindow.java#PaintWindow
```
objectConstructor = new PaintObjectConstructor(this);
objectConstructor.setClass(PencilPaint.class);
objectConstructor.setColor(new Color(0, 255, 0));
objectConstructor.setThickness(5);
canvas.addMouseListener(objectConstructor);
canvas.addMouseMotionListener(objectConstructor);
```

PaintObjectConstructor.java
```
public class PaintObjectConstructor implements MouseListener, MouseMotionListener {
```

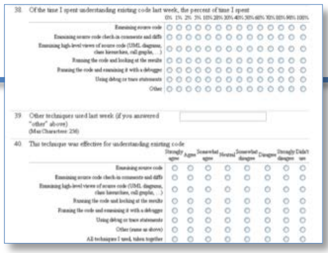
PaintWindow.java
```
private JSlider rSlider, bSlider, gSlider;
```

PaintWindow.java
```
public void paintComponent(Graphics g) {

    Color oldColor = g.getColor();
    g.setColor(objectConstructor.getColor());
    g.fillRect(0, 0, getWidth(), getHeight()
    g.setColor(oldColor);
```

PaintWindow.java#PaintWindow
```
addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent event) {
        System.exit(0);
    }
});
```

7 Studies of Understanding and Exploring Code



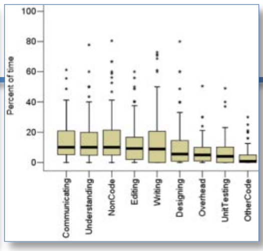
38 Of the time I spent understanding existing code last week, the percent of time I spent

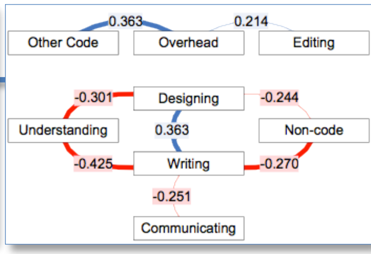
	0%	1%	2%	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Examining source code	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Examining source code check-in comments and diffs	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Examining high-level views of source code (UML diagrams, class hierarchies, call graphs, ...)	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Running the code and looking at the results	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Running the code and examining it with a debugger	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Using debug or trace statements	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Other	○	○	○	○	○	○	○	○	○	○	○	○	○	○

39 Other techniques used last week (if you answered "other" above)
(Max Characters: 250)

40 This technique was effective for understanding existing code

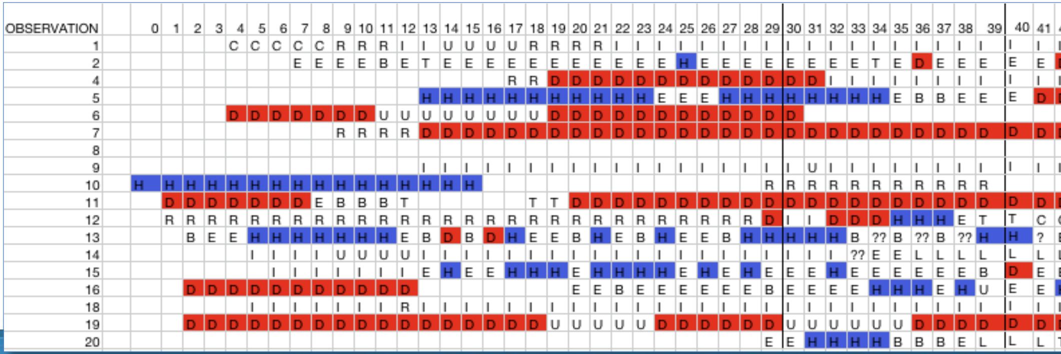
	Strongly agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Strongly disagree	Didn't use
Examining source code	○	○	○	○	○	○	○	○
Examining source code check-in comments and diffs	○	○	○	○	○	○	○	○
Examining high-level views of source code (UML diagrams, class hierarchies, call graphs, ...)	○	○	○	○	○	○	○	○
Running the code and looking at the results	○	○	○	○	○	○	○	○
Running the code and examining it with a debugger	○	○	○	○	○	○	○	○
Using debug or trace statements	○	○	○	○	○	○	○	○
Other (same as above)	○	○	○	○	○	○	○	○
All techniques I used, taken together	○	○	○	○	○	○	○	○





- PhD of Thomas LaToza (2012)
- Extensive investigations of how developers understand and explore code
  - **4 Surveys**: 740 developers at Microsoft
  - **Interviews**: 11 developers at Microsoft
  - **Controlled Experiment**: 16 students and staff at CMU x 3 hours
    - 11,821 lines of navigation events & 32 code changes
  - **Field Observations**: 17 developers at Microsoft x 90 minutes
    - 386 pages of transcripts
    - Minute by minute activity

(PLATEAU'2010)



OBSERVATION		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
1						C	C	C	C	C	R	R	R	I	I	U	U	U	U	R	R	R	R	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I
2									E	E	E	E	B	E	T	E	E	E	E	E	E	E	E	E	E	E	H	E	E	E	E	E	E	E	E	T	E	D	E	E	E	E	E
4																			R	R	D	D	D	D	D	D	D	D	D	D	D	D	D	I	I	I	I	I	I	I	I	I	I
5															H	H	H	H	H	H	H	H	H	H	H	E	E	E	H	H	H	H	H	H	H	H	E	B	B	E	E	E	D
6						D	D	D	D	D	D	D	U	U	U	U	U	U	U	U	D	D	D	D	D	D	D	D	D	D	D	D											
7											R	R	R	R	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
8																																											
9															I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	U	I	I	I	I	I	I	I	I	I	I
10		H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H														R	R	R	R	R	R	R	R	R	R	R		
11			D	D	D	D	D	D	D	E	B	B	B	T						T	T	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
12			R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	D	I	I	D	D	D	H	H	H	E	T	T	C
13				B	E	E	H	H	H	H	H	H	H	E	B	D	B	D	H	E	E	B	H	E	B	H	E	E	B	H	H	H	H	H	B	??	B	??	B	??	H	H	?
14							I	I	I	I	U	U	U	U	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	??	E	E	L	L	L	L	L	L
15								I	I	I	I	I	I	I	E	H	E	E	H	H	H	E	H	H	H	H	E	H	E	H	E	E	E	H	E	E	E	E	E	E	B	D	E
16				D	D	D	D	D	D	D	D	D	D	D								E	E	B	E	E	E	E	E	E	B	E	E	E	E	H	H	H	E	H	U	E	E
18							I	I	I	I	I	I	I	R	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I
19				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	U	U	U	U	U	D	D	D	D	D	D	U	U	U	U	U	U	D	D	D	D	D	D
20																															E	E	H	H	H	H	B	B	B	E	L	L	L

Many hard-to-answer questions about code

(PLATEAU' 2010)

Rationale (42)
Why was it done this way? (14) [15][7]
Why wasn't it done this other way? (15)
Was this intentional, accidental, or a hack? (9)[15]
How did this ever work? (4)

Debugging (26)
How did this runtime state occur? (12) [15]
What runtime state changed when this executed? (2)
Where was this variable last changed? (1)
How is this object different from that object? (1)
Why didn't this happen? (3)
How do I debug this bug in this environment? (3)
In what circumstances does this bug occur? (3) [15]
Which team's component caused this bug? (1)

Intent and Implementation (32)
What is the intent of this code? (12) [15]
What does this do (6) in this case (10)? (16) [24]
How does it implement this behavior? (4) [24]

Refactoring (25)
Is there functionality or code that could be refactored? (4)
Is the existing design a good design? (2)
Is it possible to refactor this? (9)
How can I refactor this (2) without breaking existing users(7)? (9)
Should I refactor this? (1)
Are the benefits of this refactoring worth the time investment? (3)

History (23)
When, how, by whom, and why was this code changed or inserted? (13)[7]
What else changed when this code was changed or inserted? (2)
How has it changed over time? (4)[7]
Has this code always been this way? (2)
What recent changes have been made? (1)[15][7]
Have changes in another branch been integrated into this branch? (1)

Implications (21)
What are the implications of this change for (5) API clients (5), security (3), concurrency (3), performance (2), platforms (1), tests (1), or obfuscation (1)? (21) [15][24]

Testing (20)
Is this code correct? (6) [15]
How can I test this code or functionality? (9)
Is this tested? (3)
Is the test or code responsible for this test failure? (1)
Is the documentation wrong, or is the code wrong? (1)

Implementing (19)
How do I implement this (8), given this constraint (2)? (10)
Which function or object should I pick? (2)
What's the best design for implementing this? (7)

Control flow (19)
In what situations or user scenarios is this called? (3) [15][24]
What parameter values does each situation pass to this method? (1)
What parameter values could lead to this case? (1)
What are the possible actual methods called by dynamic dispatch here? (6)
How do calls flow across process boundaries? (1)
How many recursive calls happen during this operation? (1)
Is this method or code path called frequently, or is it dead? (4)
What throws this exception? (1)
What is catching this exception? (1)

Contracts (17)
What assumptions about preconditions does this code make? (5)
What assumptions about pre(3)/post(2)conditions can be made?
What exceptions or errors can this method generate? (2)
What are the constraints on or normal values of this variable? (2)
What is the correct order for calling these methods or initializing these objects? (2)
What is responsible for updating this field? (1)

Performance (16)
What is the performance of this code (5) on a large, real dataset (3)? (8)
Which part of this code takes the most time? (4)
Can this method have high stack consumption from recursion? (1)
How big is this in memory? (2)
How many of these objects get created? (1)

Teammates (16)
Who is the owner or expert for this code? (3)[7]
How do I convince my teammates to do this the "right way"? (12)
Did my teammates do this? (1)

Policies (15)
What is the policy for doing this? (10) [24]
Is this the correct policy for doing this? (2) [15]
How is the allocation lifetime of this object maintained? (3)

Type relationships (15)
What are the composition, ownership, or usage relationships of this type? (5) [24]
What is this type's type hierarchy? (4) [24]
What implements this interface? (4) [24]
Where is this method overridden? (2)

Data flow (14)
What is the original source of this data? (2) [15]
What code directly or indirectly uses this data? (5)
Where is the data referenced by this variable modified? (2)
Where can this global variable be changed? (1)
Where is this data structure used (1) for this purpose (1)? (2) [24]
What parts of this data structure are modified by this code? (1) [24]
What resources is this code using? (1)

Location (13)
Where is this functionality implemented? (5) [24]
Is this functionality already implemented? (5) [15]
Where is this defined? (3)

Building and branching (11)
Should I branch or code against the main branch? (1)
How can I move this code to this branch? (1)
What do I need to include to build this? (3)
What includes are unnecessary? (2)
How do I build this without doing a full build? (1)
Why did the build break? (2)[59]
Which preprocessor definitions were active when this was built? (1)

Architecture (11)
How does this code interact with libraries? (4)
What is the architecture of the code base? (3)
How is this functionality organized into layers? (1)
Is our API understandable and flexible? (3)

Concurrency (9)
What threads reach this code (4) or data structure (2)? (6)
Is this class or method thread-safe? (2)
What members of this class does this lock protect? (1)

Dependencies (5)
What depends on this code or design decision? (4)[7]
What does this code depend on? (1)

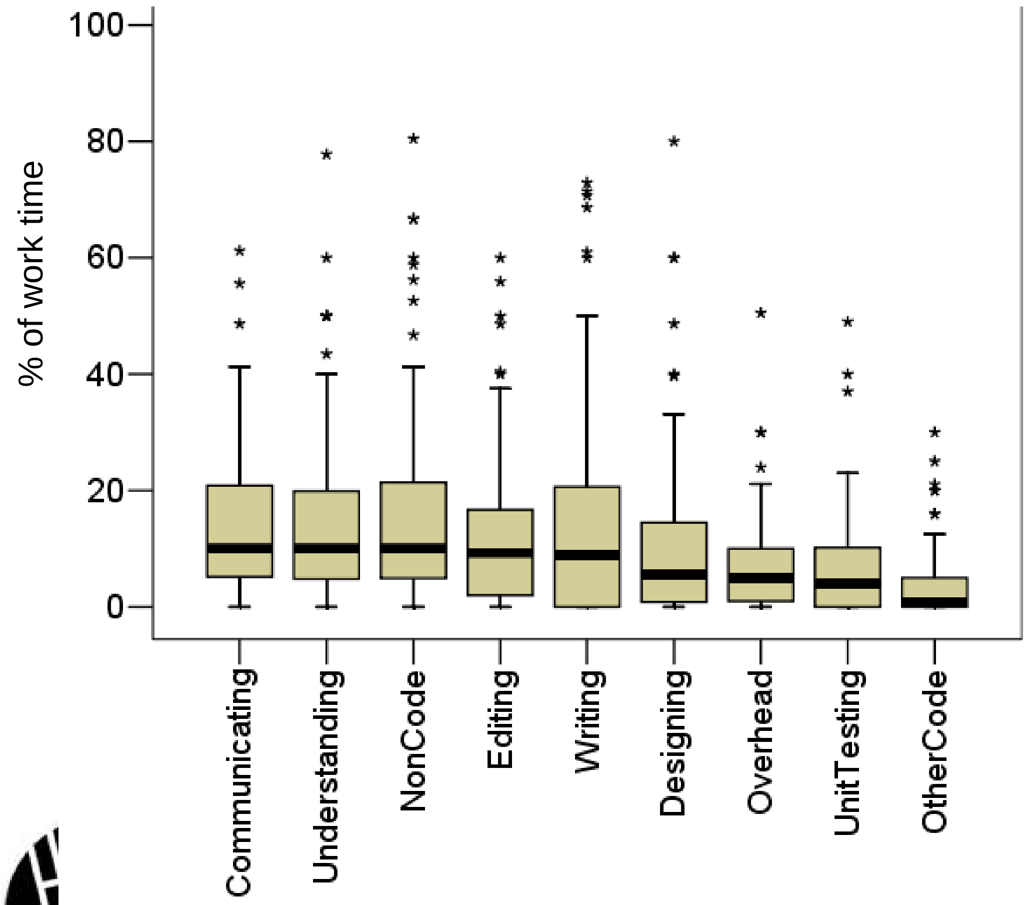
Method properties (2)
How big is this code? (1)
How overloaded are the parameters to this function? (1)

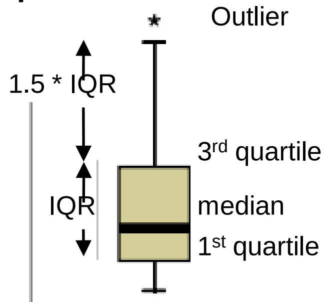
Many opportunities for better tools

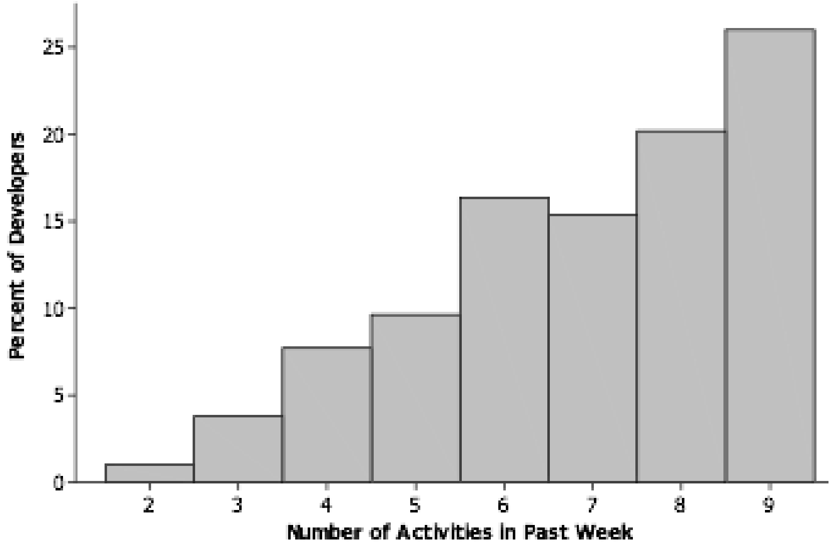
- Of all the reported questions
  - 34% addressed by commercial tools
  - 25% addressed by research tools
  - 41% unaddressed by any tools

No single activity dominates work

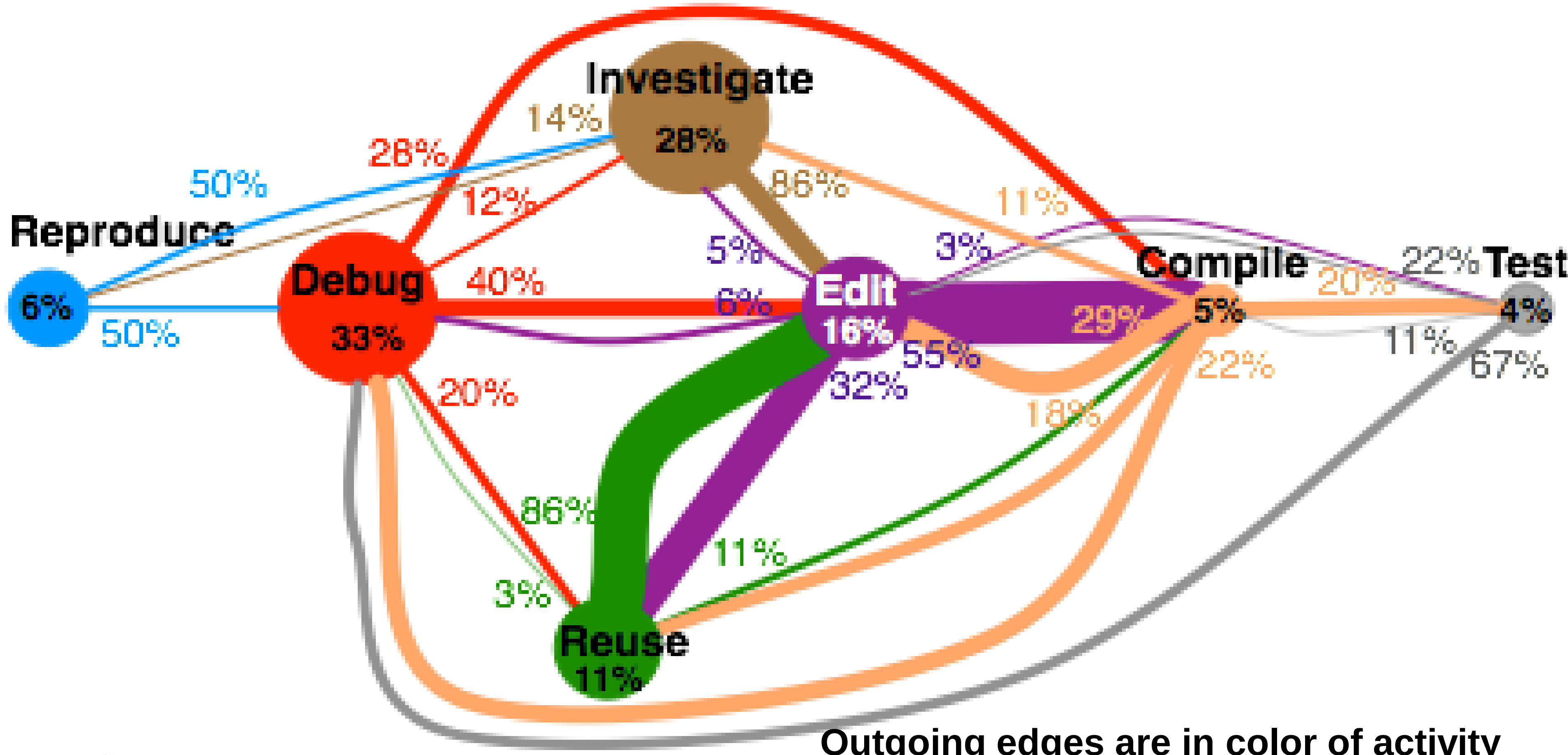
Prompt: Percent of work time last week that I spent







Most time debugging and investigating

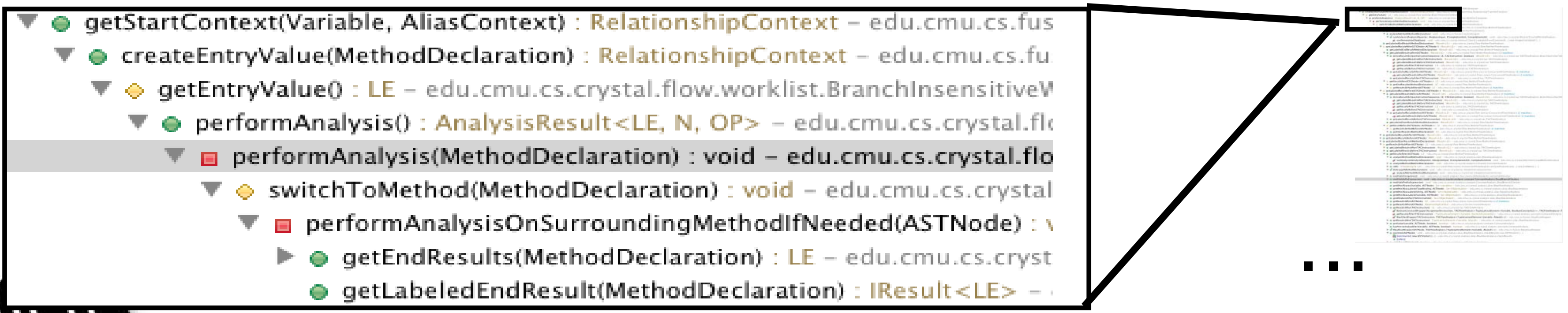


Outgoing edges are in color of activity
Circle size: % of coding activity time
Edge thickness: % of transitions observed

Frequent question: Reachability

- Programmers investigate *reachability questions*
  - How can this code *be reached*, either upstream or downstream
  - E.g., control flow from user scrolling → update status line
- Survey shows such control flow questions are difficult and important
- No easy way to discover with current tools
  - Call graphs are too general
  - Call hierarchy too deep

(ICSE'2010)

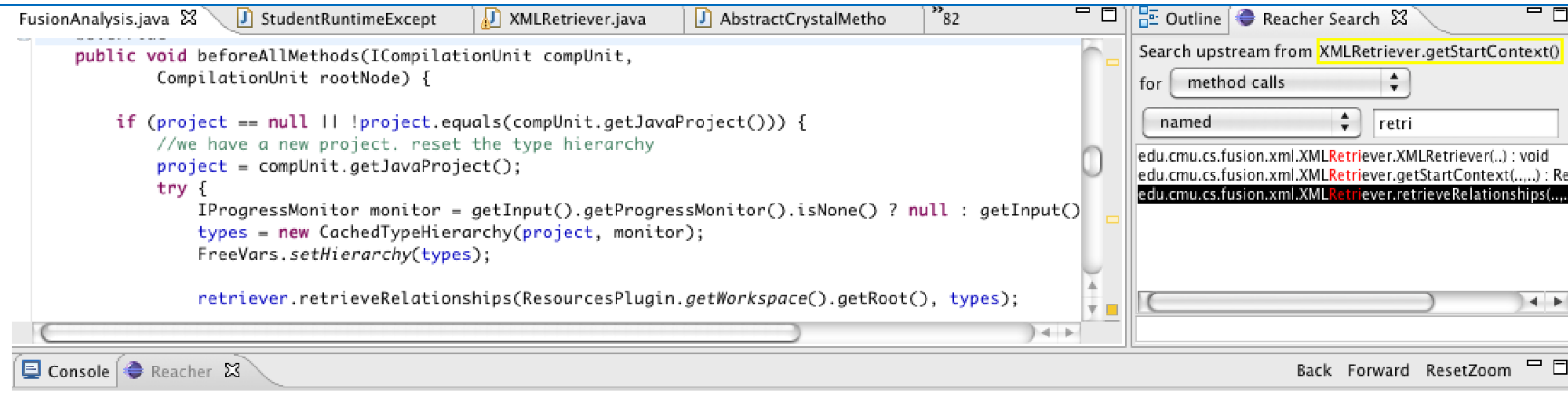


```
▼ ● getStartContext(Variable, AliasContext) : RelationshipContext – edu.cmu.cs.fus
  ▼ ● createEntryValue(MethodDeclaration) : RelationshipContext – edu.cmu.cs.fu
    ▼ ◆ getEntryValue() : LE – edu.cmu.cs.crystal.flow.worklist.BranchInsensitiveW
      ▼ ● performAnalysis() : AnalysisResult<LE, N, OP> – edu.cmu.cs.crystal.flo
        ▼ ■ performAnalysis(MethodDeclaration) : void – edu.cmu.cs.crystal.flo
          ▼ ◆ switchToMethod(MethodDeclaration) : void – edu.cmu.cs.crystal
            ▼ ■ performAnalysisOnSurroundingMethodIfNeeded(ASTNode) : v
              ▶ ● getEndResults(MethodDeclaration) : LE – edu.cmu.cs.cryst
                ● getLabeledEndResult(MethodDeclaration) : IResult<LE> –
```

...

REACHER

- Visualize exactly the paths of interest *(VL/HCC'11)*
- Search along the paths
- Focused questions and answers enable effective analysis of complex codebases
- Developers with Reacher 5.6 times more successful than those working with Eclipse only *0:53*



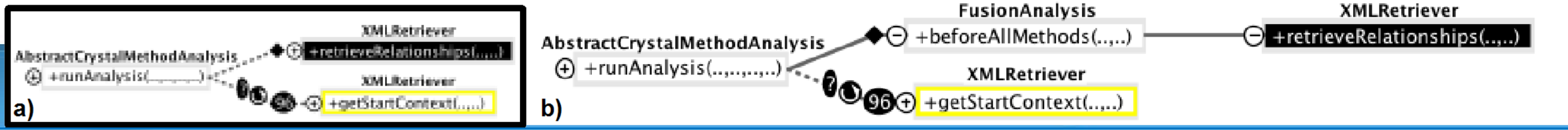
```
public void beforeAllMethods(ICompilationUnit compUnit,
        CompilationUnit rootNode) {

    if (project == null || !project.equals(compUnit.getJavaProject())) {
        //we have a new project. reset the type hierarchy
        project = compUnit.getJavaProject();
        try {
            IProgressMonitor monitor = getInput().getProgressMonitor().isNone() ? null : getInput()
            types = new CachedTypeHierarchy(project, monitor);
            FreeVars.setHierarchy(types);

            retriever.retrieveRelationships(ResourcesPlugin.getWorkspace().getRoot(), types);
```

Search upstream from XMLRetriever.getStartContext()
for method calls
named retri

```
edu.cmu.cs.fusion.xml.XMLRetriever.XMLRetriever(..) : void
edu.cmu.cs.fusion.xml.XMLRetriever.getStartContext(...,..) : Re
edu.cmu.cs.fusion.xml.XMLRetriever.retrieveRelationships(...,
```



AbstractCrystalMethodAnalysis
+runAnalysis(...,...,...)
XMLRetriever
+retrieveRelationships(...,..)
XMLRetriever
96
+getStartContext(...,..)
a)
AbstractCrystalMethodAnalysis
+runAnalysis(..,..,..,..,..)
FusionAnalysis
+beforeAllMethods(..,..)
XMLRetriever
+retrieveRelationships(..,..)
XMLRetriever
?
96
+getStartContext(..,..)
b)

Study of APIs

- Started as PhD work of Jeff Stylos, 2009
  - Inspired by Steven Clarke, Microsoft Visual Studio group
- Application Programming Interface
  - Libraries, frameworks, SDKs, ...
- Barriers to understanding of APIs
- Measures: learnability, errors, preferences
- Expert and novice programmers
- Studied:
  - Default parameters in constructors
  - Factory pattern
  - Object design
  - SAP's Web Services APIs

"Factory" Pattern

- Instead of "normal" creation: `Widget w = new Widget();`
- Objects must be created by *another* class:

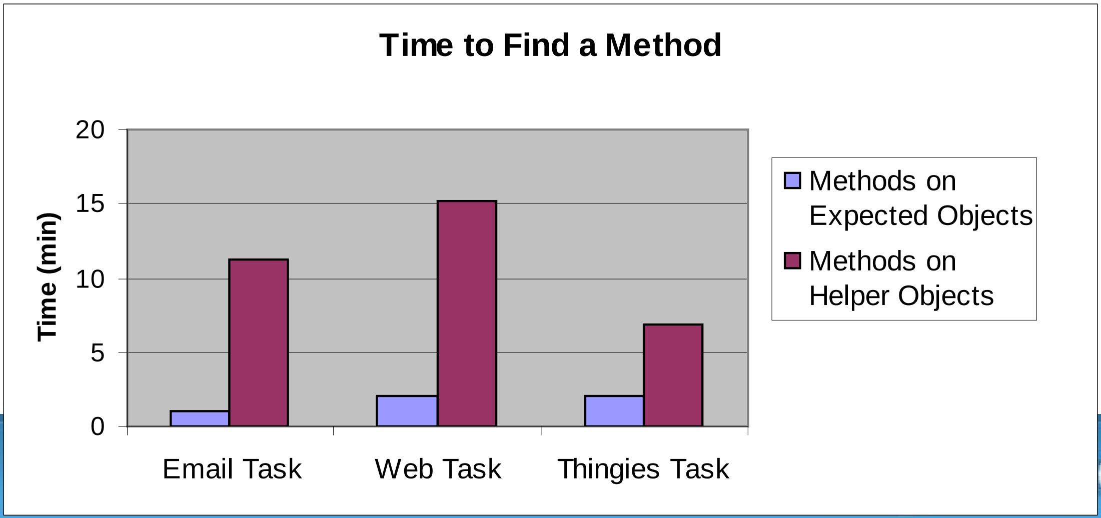
```
AbstractFactory f =  AbstractFactory.getDefault();
Widget w = f.createWidget();
```

- Used frequently in Java (>61) and .Net (>13) and SAP
- Results:
  - When asked to design on "blank paper", no one designed a factory
  - Time to develop using factories took 2.1 to 5.3 times longer compared to regular constructors (20:05 v 9:31, 7:10 v 1:20)
  - All subjects had difficulties understanding factories in APIs

(ICSE'2007)

Object Method Placement

- Where to put functions when doing object-oriented design of APIs when multiple classes work together
  - **`mail_Server.send( mail_Message )`**
    vs.
    `mail_Message.send( mail_Server  )`
- When desired method is on the class that they start with, users were between 2.4 and 11.2 times faster *($p < 0.05$)*
  - Initial intuition that class size correlated with difficulty was *wrong*
- Starting class can be predicted based on user's tasks

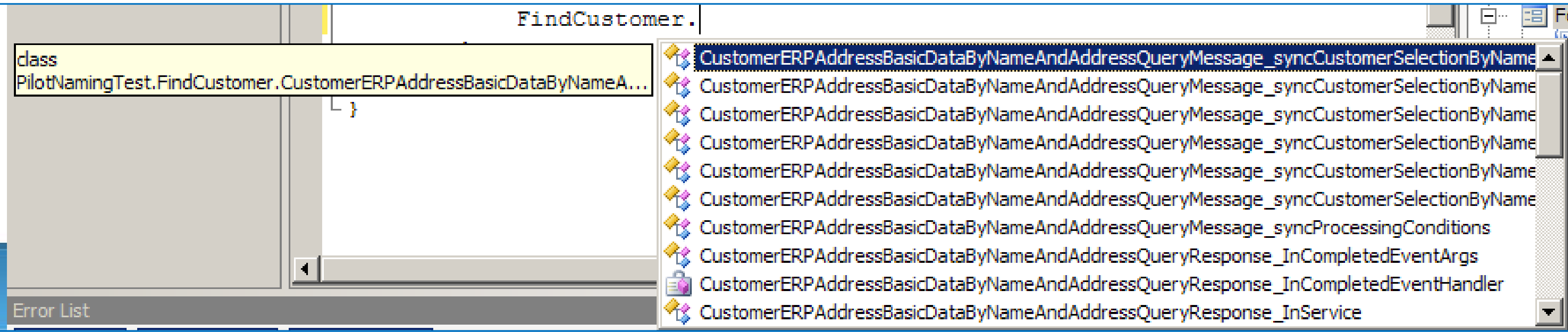


(FSE'2008)

Study of APIs for SAP

- Study APIs for Enterprise Service-Oriented Architectures ("Web Services")
- Naming problems:
  - Too long `MaterialSimpleByIDAndDescriptionQueryMessage_syncMaterialSimpleSelectionByIDAndDescriptionSelectionByMaterialDescriptio`
  - Not understandable
  - Differences in *middle* are frequently missed

CustomerAddressBasicDataByNameAndAddressRequestMessageCustomerSelectionCommonName
CustomerAddressBasicDataByNameAndAddressResponseMessageCustomerSelectionCommonName



```
FindCustomer.
```

```
class
PilotNamingTest.FindCustomer.CustomerERPAddressBasicDataByNameA...
```

```
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncCustomerSelectionByName
CustomerERPAddressBasicDataByNameAndAddressQueryMessage_syncProcessingConditions
CustomerERPAddressBasicDataByNameAndAddressQueryResponse_InCompletedEventArgs
CustomerERPAddressBasicDataByNameAndAddressQueryResponse_InCompletedEventHandler
CustomerERPAddressBasicDataByNameAndAddressQueryResponse_InService
```

eSOA Documentation Results

- Multiple paths: unclear which one to use
- Some paths were dead ends
- Inconsistent look and feel caused immediate abandonment of paths
- Hard to find required information
- Business background helped

(IS-EUD'2009)

SAP | ES WORKPLACE

ENTERPRISE SERVICES BY ENTERPRISE SER
ES BUNDLES

Enterprise Services Bundles group enterprise services according to business criteria. This
bundles.

Please note: Enterprise Services Bundles including enterprise services from various applica
several applications are listed more than once.
Browse the following Enterprise Services Bundles:

Enterprise Services Bundles providing enterprise services for SAP ERP

Financials

- Bank Communication Management
- Credit Management

SAP COMMUNITY NETWORK
Welcome Guest

SDN Community | BPX Community | Business Objects | Forums | Wiki | Blogs | Articles | Downlo

Member Login — Not a member?
User ID or Email
Password
☐ Remember me
Forgot your password? Log in

- FAQ
- Community Guidelines
- Contributors Corner

CONTRIBUTORS CORNER

Top Contributors
Top Companies
SAP Mentors
Contributors Corner Page
Recognition Program FAQ

My Home > Enterprise Services WIKI > ... > Custome
Business Objects > Sales Order

Sales Order

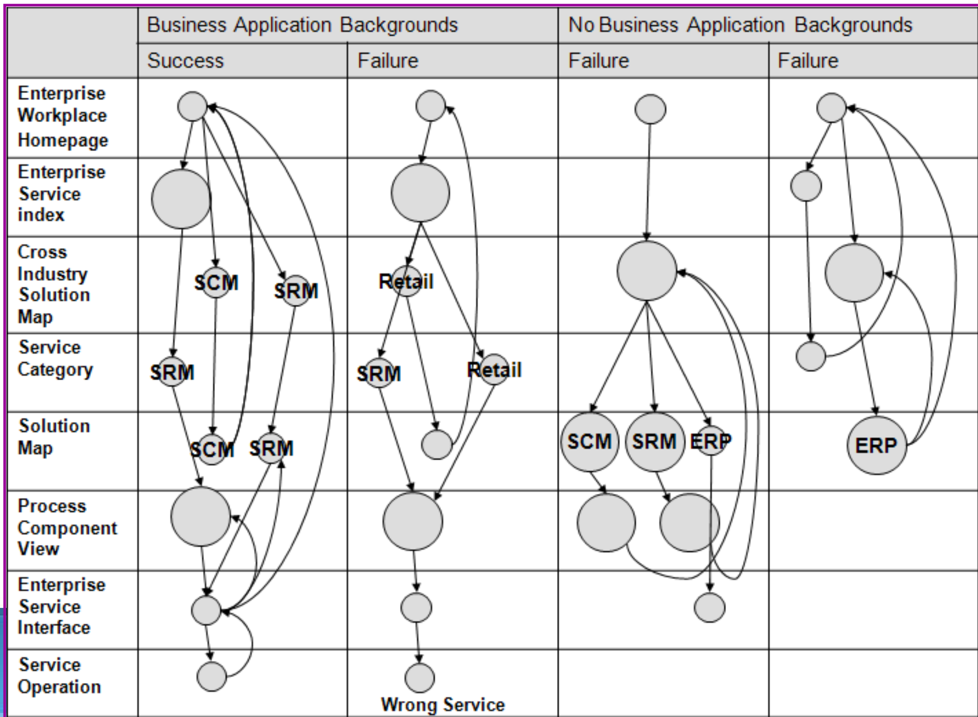
View | Comments (0) | Info

Added by Daniel Sass , last edited by Deborah G
Labels: business object, customer fact sheet,
logistics execution tms integration,
quote to order for configurable produc
order to cash for fashion, integration

A **Sales Order** is an agreement between a vend
are associated with these processes, on a specifi

Enterprise Service Operations

- Create Sales Order
- Create Sales Order V1
- Create Sales Order V2
- Check Sales Order Creation
- Read Sales Order V1
- Read Sales Order V2
- Read Sales Order V3

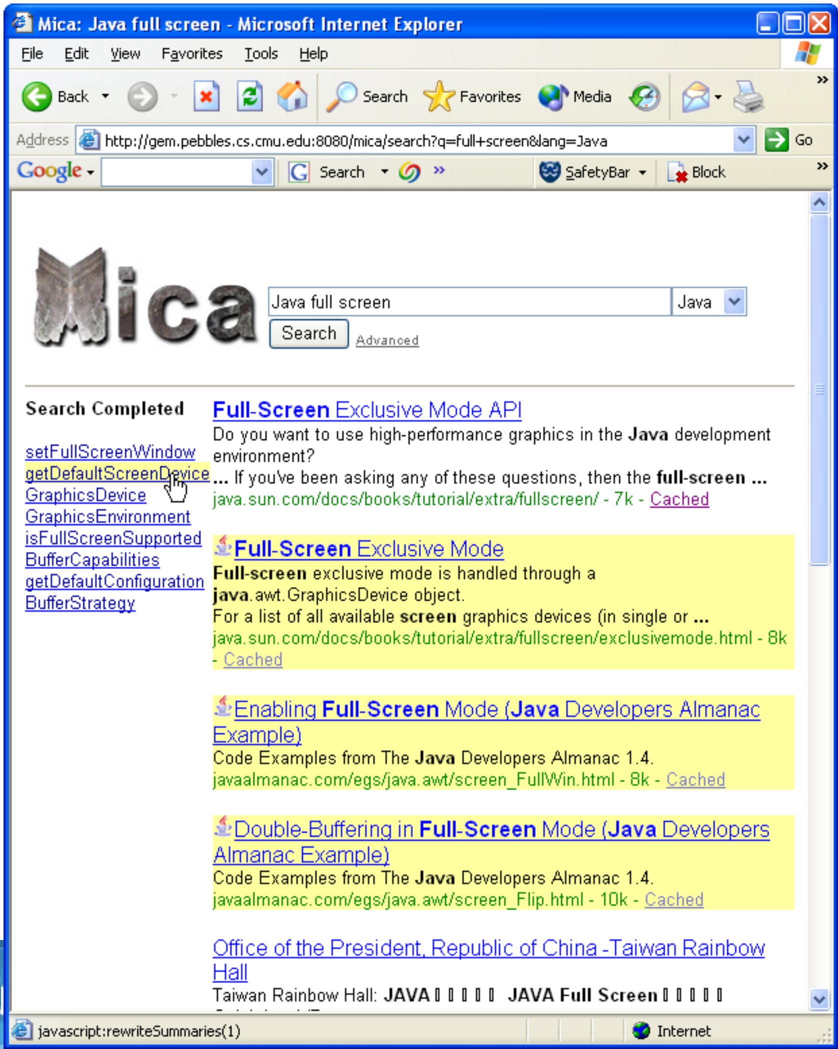


Our Tools to Help with APIs

- Mica
- Jadeite
- Calcite
- Euklas
- Graphite
- Apatite

Mica Tool to Help Find Examples

- Makes Interfaces Clear and Accessible
- Use Google to find relevant pages
- Match pages with Java keywords
- Also notes which pages contain example code or definitions


Mica: Java full screen - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address
Go
Google
Search
SafetyBar
Block
Java
Search
Advanced
Internet

http://gem.pebbles.cs.cmu.edu:8080/mica/search?q=full+screen&lang=Java

Java full screen

Search Completed

setFullScreenWindow
getDefaultScreenDevice
GraphicsDevice
GraphicsEnvironment
isFullScreenSupported
BufferCapabilities
getDefaultConfiguration
BufferStrategy

Full-Screen Exclusive Mode API
Do you want to use high-performance graphics in the **Java** development environment?
... If you've been asking any of these questions, then the **full-screen** ...
java.sun.com/docs/books/tutorial/extra/fullscreen/ - 7k - Cached

Full-Screen Exclusive Mode
Full-screen exclusive mode is handled through a **java**.awt.GraphicsDevice object.
For a list of all available **screen** graphics devices (in single or ...
java.sun.com/docs/books/tutorial/extra/fullscreen/exclusivemode.html - 8k - Cached

Enabling **Full-Screen** Mode (**Java** Developers Almanac Example)
Code Examples from The **Java** Developers Almanac 1.4.
javaalmanac.com/egs/java.awt/screen_FullWin.html - 8k - Cached

Double-Buffering in **Full-Screen** Mode (**Java** Developers Almanac Example)
Code Examples from The **Java** Developers Almanac 1.4.
javaalmanac.com/egs/java.awt/screen_Flip.html - 10k - Cached

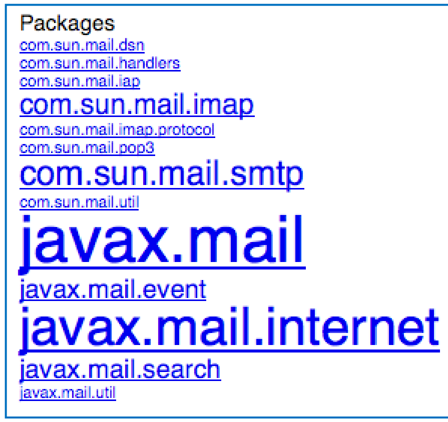
Office of the President, Republic of China -Taiwan Rainbow Hall
Taiwan Rainbow Hall: **JAVA** ▯▯▯▯▯ **JAVA Full Screen** ▯▯▯▯▯

javascript:rewriteSummaries(1)

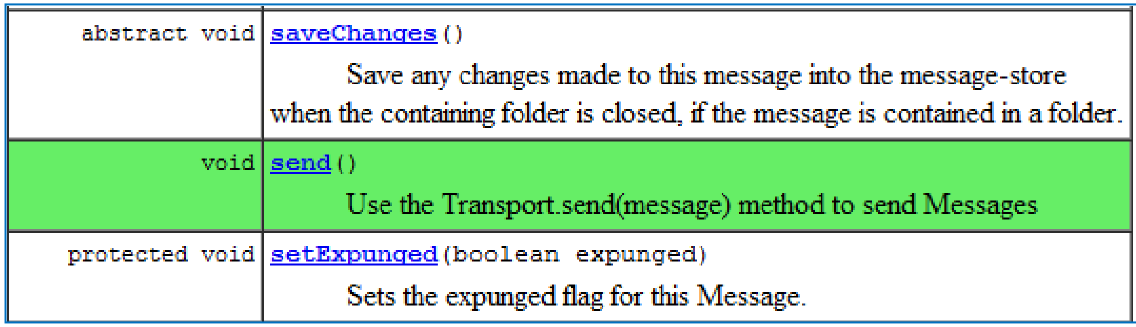
(VL/HCC'06)

Jadeite: Improved JavaDoc

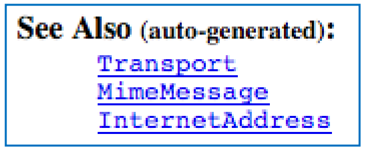
- Jadeite: Java API Documentation with Extra Information Tacked-on for Emphasis
  http://www.cs.cmu.edu/~jadeite
- Fix JavaDoc to help address problems
  - Focus attention on most popular packages and classes using font size
  - "Placeholders" for methods that users want to exist
  - Automatically extracted code examples for how to create classes



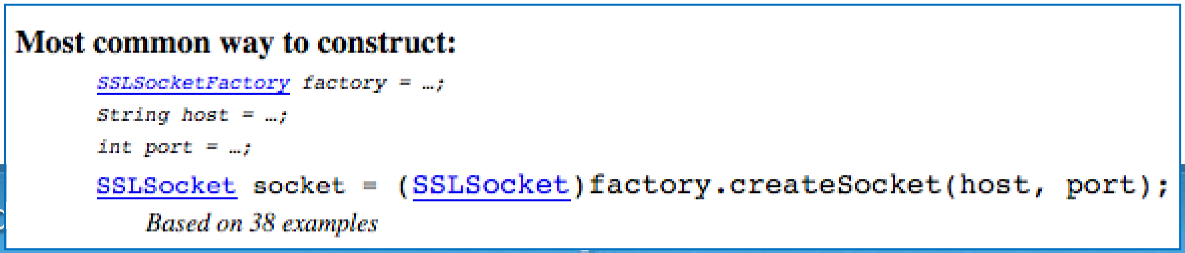
Packages
com.sun.mail.dsn
com.sun.mail.handlers
com.sun.mail.iap
com.sun.mail.imap
com.sun.mail.imap.protocol
com.sun.mail.pop3
com.sun.mail.smtp
com.sun.mail.util
javax.mail
javax.mail.event
javax.mail.internet
javax.mail.search
javax.mail.util



abstract void	saveChanges() Save any changes made to this message into the message-store when the containing folder is closed, if the message is contained in a folder.
void	send() Use the Transport.send(message) method to send Messages
protected void	setExpunged(boolean expunged) Sets the expunged flag for this Message.



See Also (auto-generated):
Transport
MimeMessage
InternetAddress



Most common way to construct:

```
SSLSocketFactory factory = …;
String host = …;
int port = …;
SSLSocket socket = (SSLSocket)factory.createSocket(host, port);
```

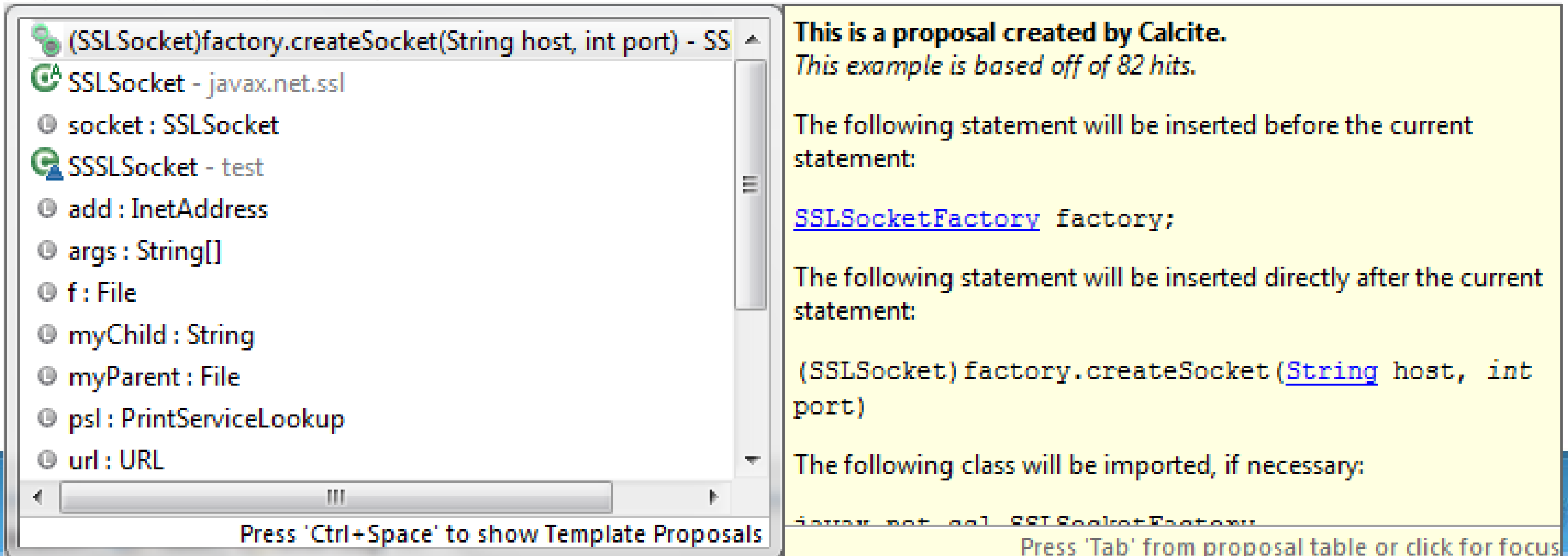
Based on 38 examples

(VL/HCC'09)

Calcite: Eclipse Plugin for Java

- Calcite: Construction And Language Completion Integrated Throughout
  http://www.cs.cmu.edu/~calcite
- Code completion in Eclipse augmented with Jadeite's information
  - How to create objects of specific classes
    `SSLSocket s = ???`

(VL/HCC'10)



```
(SSLSocket)factory.createSocket(String host, int port) - SS
SSLSocket - javax.net.ssl
socket : SSLSocket
SSSLSocket - test
add : InetAddress
args : String[]
f : File
myChild : String
myParent : File
psl : PrintServiceLookup
url : URL
```

Press 'Ctrl+Space' to show Template Proposals

This is a proposal created by Calcite.
This example is based off of 82 hits.

The following statement will be inserted before the current statement:

```
SSLSocketFactory factory;
```

The following statement will be inserted directly after the current statement:

```
(SSLSocket)factory.createSocket(String host, int port)
```

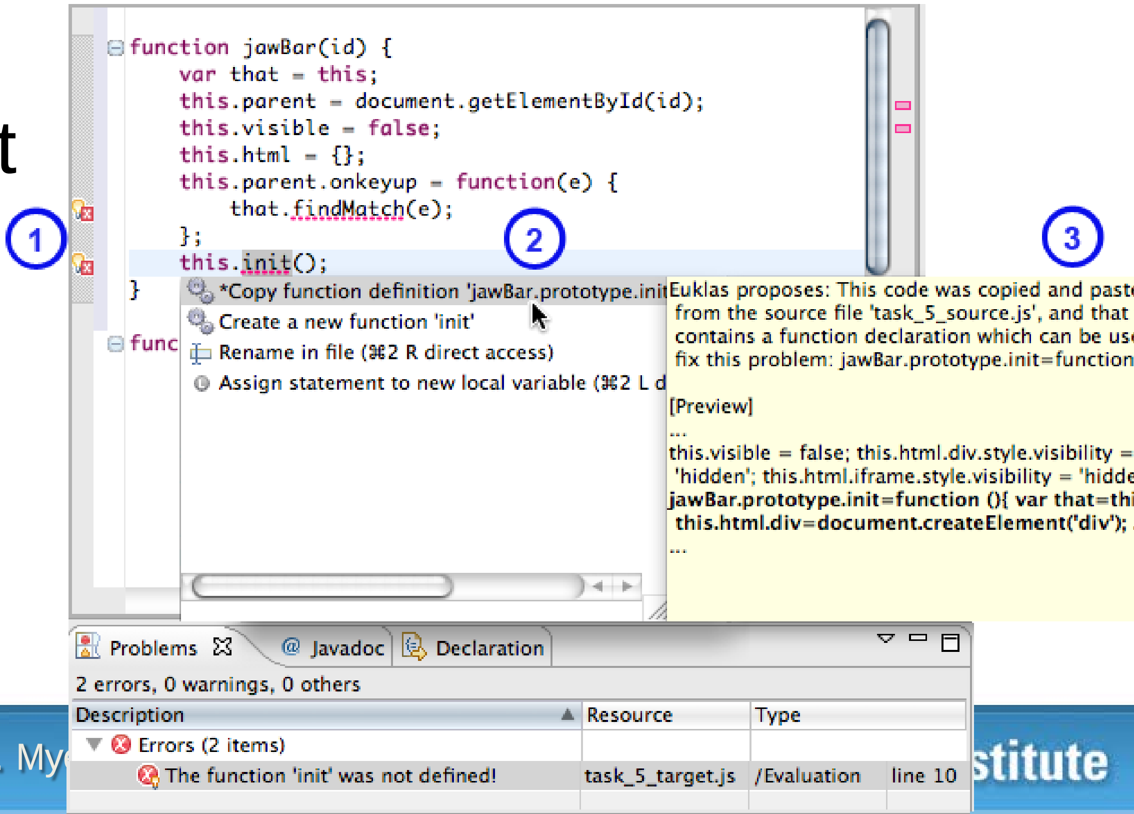
The following class will be imported, if necessary:

```
javax.net.ssl.SSLSocketFactory;
```

Press 'Tab' from proposal table or click for focus

Euklas: Eclipse Plugin for JavaScript

- Euklas: Eclipse Users’ Keystrokes Lessened by Attaching from Samples
  http://www.cs.cmu.edu/~euklas
- Brings Java-like analysis to JavaScript
- Auto-correct uses copy source context for errors due to copy & paste



```
function jawBar(id) {
    var that = this;
    this.parent = document.getElementById(id);
    this.visible = false;
    this.html = {};
    this.parent.onkeyup = function(e) {
        that.findMatch(e);
    };
    this.init();
}

func
```

1
2
3
*Copy function definition 'jawBar.prototype.init
Create a new function 'init'
Rename in file (⌘2 R direct access)
Assign statement to new local variable (⌘2 L d

Euklas proposes: This code was copied and paste from the source file 'task_5_source.js', and that contains a function declaration which can be use fix this problem: jawBar.prototype.init=function

[Preview]

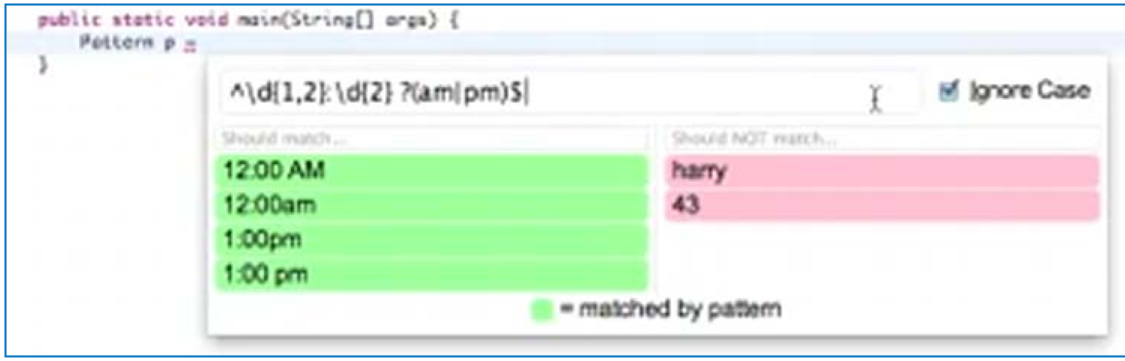
...
this.visible = false; this.html.div.style.visibility = 'hidden'; this.html.iframe.style.visibility = 'hidde
jawBar.prototype.init=function (){ var that=thi this.html.div=document.createElement('div');
...

2 errors, 0 warnings, 0 others

Description	Resource	Type	
Errors (2 items)			
The function 'init' was not defined!	task_5_target.js	/Evaluation	line 10

Graphite: Eclipse Plugin for Literals

- Graphite: GRAphical Palettes Help Instantiate Types in the Editor.
- Pop up a custom palette for specialized constants (literals) in Eclipse
  - Color palettes
  - Regular expression strings
- Customizable



```
public static void main(String[] args) {
    Pattern p =
}
```

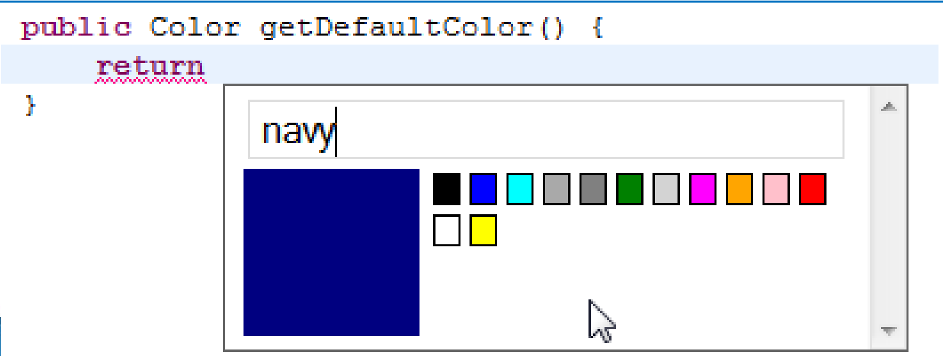
^\d{1,2}:\d{2} ?(am|pm)$

☑ Ignore Case

Should match...	Should NOT match...
12:00 AM	harry
12:00am	43
1:00pm	
1:00 pm	

🟩 = matched by pattern

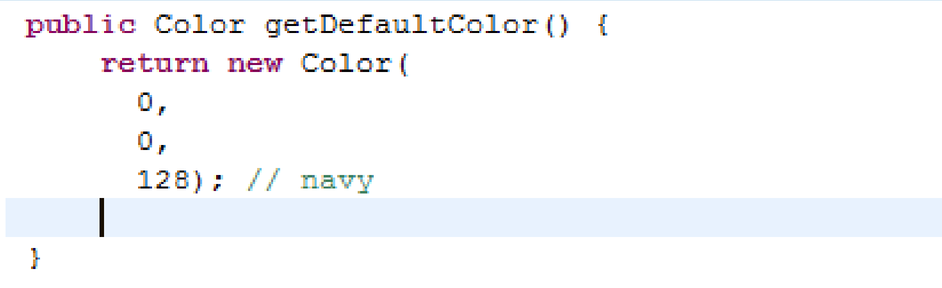
(ICSE'2012)



```
public Color getDefaultColor() {
    return
}
```

navy

(a)



```
public Color getDefaultColor() {
    return new Color(
      0,
      0,
      128); // navy

}
```

(b)

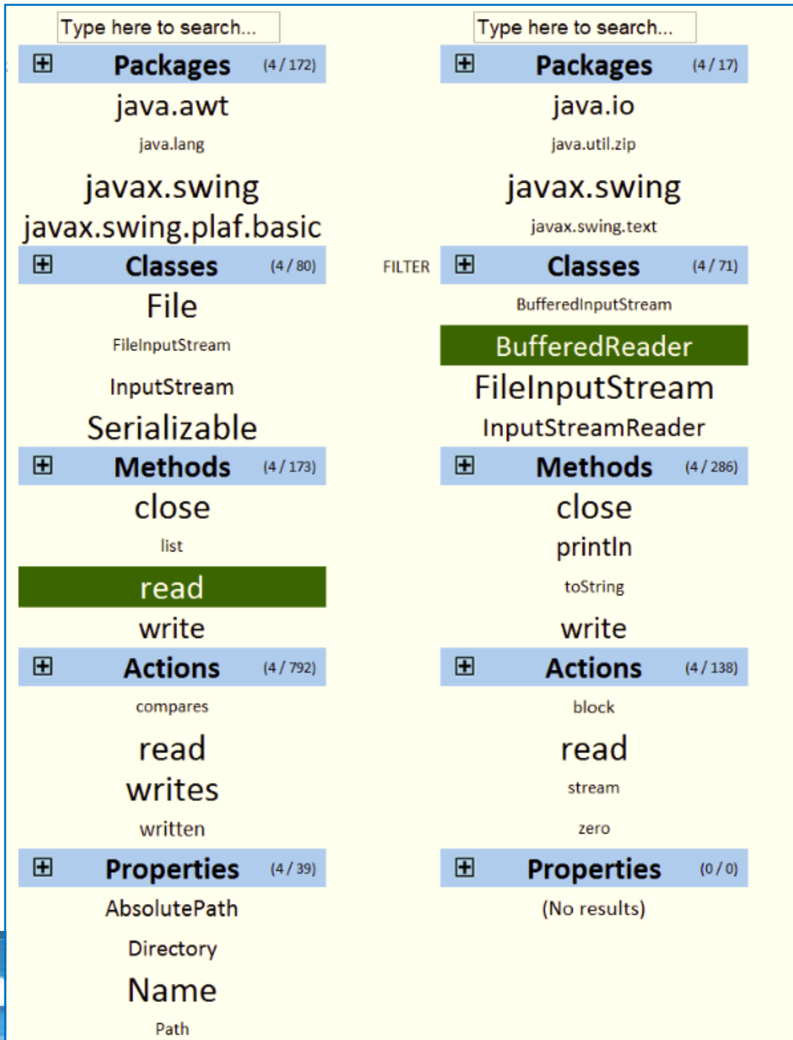
Apatite Documentation Tool

- Apatite: Associative Perusing of APIs That Identifies Targets Easily

  http://www.cs.cmu.edu/~apatite

- Start with verbs (actions) and properties and find what classes implement them
- Find associated items
  - E.g., classes that are often used together
  - Classes that implement or are used by a method

(VL/HCC'10)



Type here to search...

⊞ Packages (4 / 172)
java.awt
java.lang
javax.swing
javax.swing.plaf.basic

⊞ Classes (4 / 80)
File
FileInputStream
InputStream
Serializable

⊞ Methods (4 / 173)
close
list
read
write

⊞ Actions (4 / 792)
compares
read
writes
written

⊞ Properties (4 / 39)
AbsolutePath
Directory
Name
Path

Type here to search...

⊞ Packages (4 / 17)
java.io
java.util.zip
javax.swing
javax.swing.text

FILTER ⊞ Classes (4 / 71)
BufferedInputStream
BufferedReader
FileInputStream
InputStreamReader

⊞ Methods (4 / 286)
close
println
toString
write

⊞ Actions (4 / 138)
block
read
stream
zero

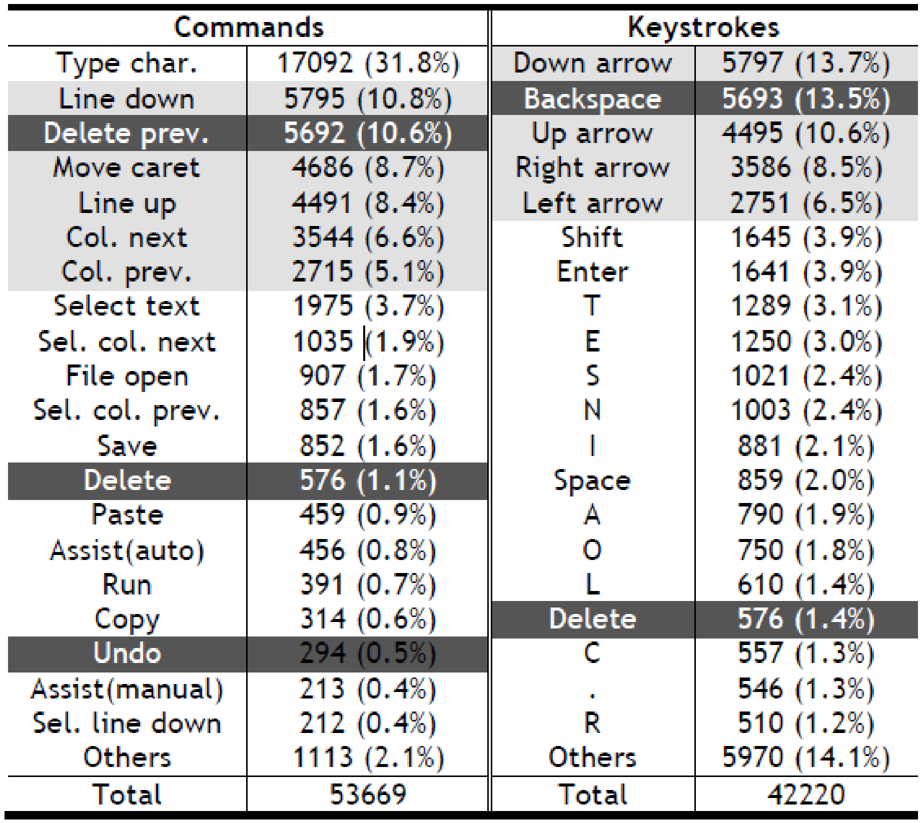
⊞ Properties (0 / 0)
(No results)

Exploratory Programming and Understanding

- PhD work of YoungSeok Yoon (in progress)
- Explorations
  - When trying different approaches
  - When trying to understand an API
  - When trying out different fixes
  - ...

Fluorite Logger

- Fluorite: Full of Low-level User Operations Recorded In The Editor http://www.cs.cmu.edu/~fluorite
- Logger for *all* keystrokes & events in Eclipse
- Analyzes frequencies and patterns
- Deleting is a high percent of all the keystrokes
- Also surveyed >100 developers



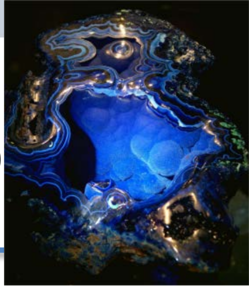
Commands		Keystrokes	
Type char.	17092 (31.8%)	Down arrow	5797 (13.7%)
Line down	5795 (10.8%)	Backspace	5693 (13.5%)
Delete prev.	5692 (10.6%)	Up arrow	4495 (10.6%)
Move caret	4686 (8.7%)	Right arrow	3586 (8.5%)
Line up	4491 (8.4%)	Left arrow	2751 (6.5%)
Col. next	3544 (6.6%)	Shift	1645 (3.9%)
Col. prev.	2715 (5.1%)	Enter	1641 (3.9%)
Select text	1975 (3.7%)	T	1289 (3.1%)
Sel. col. next	1035 (1.9%)	E	1250 (3.0%)
File open	907 (1.7%)	S	1021 (2.4%)
Sel. col. prev.	857 (1.6%)	N	1003 (2.4%)
Save	852 (1.6%)	I	881 (2.1%)
Delete	576 (1.1%)	Space	859 (2.0%)
Paste	459 (0.9%)	A	790 (1.9%)
Assist(auto)	456 (0.8%)	O	750 (1.8%)
Run	391 (0.7%)	L	610 (1.4%)
Copy	314 (0.6%)	Delete	576 (1.4%)
Undo	294 (0.5%)	C	557 (1.3%)
Assist(manual)	213 (0.4%)	.	546 (1.3%)
Sel. line down	212 (0.4%)	R	510 (1.2%)
Others	1113 (2.1%)	Others	5970 (14.1%)
Total	53669	Total	42220

(CHASE'12)

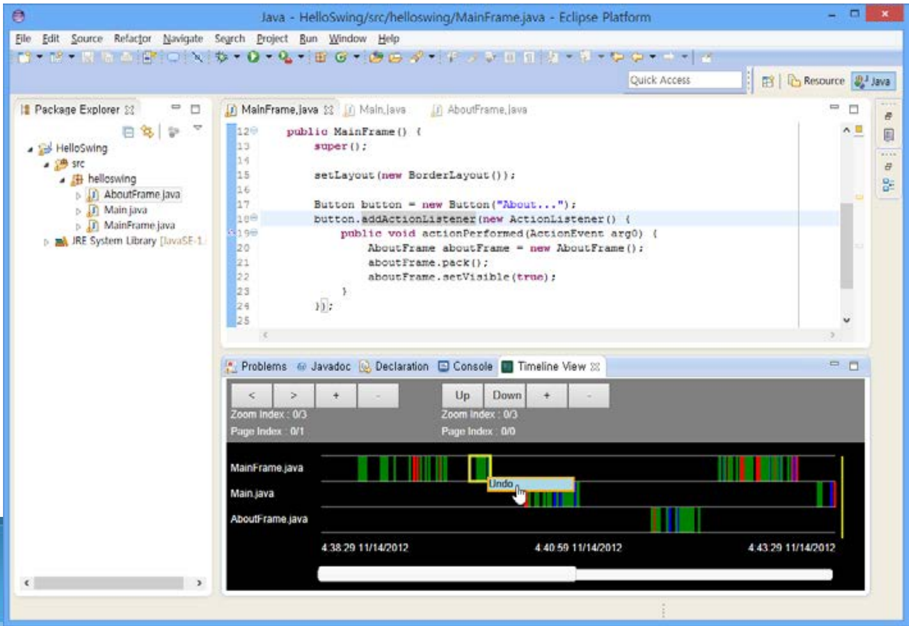
Backtracking Results

- All developers *backtrack* for many reasons
  - Explorations, investigations, iterative design
- Undo not used for exploration, just typo fixing
- People use comments to remove code, so they can restore it if necessary
  - But difficult to comment & uncomment correctly
  - Often non-local changes
- Current work: new tool to help developers backtrack

Azurite: Eclipse Plugin for Selective Undo



- PhD work of YoungSeok Yoon (in progress)
- Azurite: Adding Zest to Undoing and Restoring Improves Textual Exploration http://www.cs.cmu.edu/~azurite
- Work out semantics of selective undo for code
  - Conflicting edits of same code must be shown to user
- Time-line visualization of all past operations
- Side-by-side view of current and past code
- Search through history (time) to find appropriate points



```
    public MainFrame() {
        super();

        setLayout(new BorderLayout());

        Button button = new Button("About...");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                AboutFrame aboutFrame = new AboutFrame();
                aboutFrame.pack();
                aboutFrame.setVisible(true);
            }
        });
```

Summary of Insights

- Field and lab studies can reveal developer's real questions
  - Answering these questions creates tools that are actually useful
- Researcher's intuitions about what might be useful are often wrong
- Our experience highlights:
  - Developers often have specific questions in mind, which can be exploited in tools
  - Code views are central
  - Visualizations are often useful as navigation aides for code
  - Ability to search is key
    - Not just through code, but also through dynamic and static call-graphs, through time, etc.

There are lots of Gemstones!

- *And acronyms are fun!*

Fluorite:
Full of
Low-level
User
Operations
Recorded In
The
Editor

Azurite:
Adding
Zest to
Undoing and
Restoring
Improves
Textual
Exploration

Euklas
Eclipse
Users'
Keystrokes
Lessened by
Attaching from
Samples

Graphite:
GRAphical
Palettes
Help
Instantiate
Types in the
Editor

Apatite:
Associative
Perusing of
APIs
That
Identifies
Targets
Easily

Calcite:
Construction
And
Language
Completion
Integrated
Throughout

Jadeite:
Java
API
Documentation with
Extra
Information
Tacked-on for
Emphasis

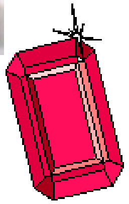
Crystal:
Clarifications
Regarding Your
Software using a
Toolkit,
Architecture and
Language

Mica:
Makes
Interfaces
Clear and
Accessible

Jasper:
Java
Aid with
Sets of
Pertinent
Elements for
Recall

Whyline
Workspace that
Helps
You
Link
Instructions to
Numbers and
Events

GARNET
Generating an
Amalgam of
Real-time,
Novel
Editors and
Toolkits



PEBBLES
PDAs for
Entry of
Both
Bytes and
Locations from External
Sources

C32
CMU's
Clever and
Compelling
Contribution to
Computer Science in
CommonLisp which is
Customizable and
Characterized by a
Complete
Coverage of
Code and
Contains a
Cornucopia of
Creative
Constructs, because it
Can
Create
Complex,
Correct
Constraints that are
Constructed
Clearly and
Concretely, and
Communicated using
Columns of
Cells, that are
Constantly
Calculated so they
Change
Continuously, and
Cancel
Confusion

For more, see: www.cs.cmu.edu/~bam/acronyms.html

Thanks to:

- Funding:
  - NSF under IIS-1116724, IIS-0329090, CCF-0811610, IIS-0757511 (Creative-IT), NSF ITR CCR-0324770 as part of the EUSES Consortium
  - SAP
  - Adobe
  - IBM
  - Microsoft Research RISE
- >30 students:
  - Htet Htet Aung
  - Jack Beaton
  - Ruben Carbonell
  - John R. Chang
  - Kerry S. Chang
  - Polo Chau
  - Luis J. Cota
  - Michael Coblenz
  - Dan Eisenberg
  - Brian Ellis
  - Andrew Faulring
  - Aristiwidya B. (Ika) Hardjanto
  - Erik Harpstead
  - Sae Young (Sophie) Jeong
  - Andy Ko
  - Sebon Koo
  - Thomas LaToza
  - Joonhwan Lee
  - Leah Miller
  - Mathew Mooty
  - Gregory Mueller
  - Yoko Nakano
  - Stephen Oney
  - John Pane
  - Sunyoung Park
  - Chotirat (Ann) Ratanamahatana
  - Christopher Scaffidi
  - Jeff Stylos
  - David A. Weitzman
  - Yingyu (Clare) Xie
  - Zizhuang (Zizzy) Yang
  - YoungSeok Yoon

Thank You!

Improving Program Comprehension by Answering Questions

Brad A. Myers
Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
http://www.cs.cmu.edu/~bam
bam@cs.cmu.edu