CSEP 504
Advanced topics in software systems

Reid Holmes • Winter 2010 • CSEP504 Lecture 6
CSEP 504: Advanced topics in software systems

• Tonight: last lecture on software tools and environments – a look at recent research and future directions
  – Emphasis on:
    • Capturing latent knowledge
    • Task specificity and awareness
    • Supporting collaborative development
Tonight

• Capturing latent knowledge
  – Hipikat / Bridge / Deep Intellisense / Hatari

• Task specificity
  – Mylyn
  – TeamTracks
  – Bubbles

• Supporting collaborative development
  – Jazz
  – FastDash
  – Customized Awareness Streams
  – Codebook

• Visualization stuff
  – Only if interested
Latent Knowledge

• Embrace practice rather than force change
• Use available data more effectively
• Source code is king [Singer ‘98]
  – Bugs
  – Version control
  – E-Mail / mailing lists
  – Forums
• Improve mapping from data to task
Utility of Latent Knowledge

• High-level information is key
  • Observing developers to identify their information needs [Ko & DeLine ‘07]

• We rely heavily on implicit knowledge
  • Surveying developers to infer their habits and mental models [LaToza, DeLine, & Venolia ’06]

• We search change and bug history daily
  • Surveying Windows developers about how they search through their source code
Hipikat

- Cubranic et al. [TSE ‘05]
- Implicit group memory (project memory)
- Acts as informal mentor
  - Relevant for non-collocated teams
- Volume hampers browsing
- Silos impede searching
Hipikat Artifacts
People as First-Class Members

Diagram:
- Person
  - writes
  - writes
  - works on
    - Hipikat
      - works on
        - posts
  - writes
Hipikat Evaluation

• 20 task case study
  – Identify most relevant files for 15T
  – 1\textsuperscript{st} or 2\textsuperscript{nd} file rec. relevant for 11 / 16T
  – 1\textsuperscript{st} or 2\textsuperscript{nd} construct rec. relevant for 10 / 13T

• Eclipse study (8 devs)
  – Used to study problem but not perform task
    • (e.g., orientation within test system)
Bridge

• Venolia [MSR ‘06]
• Augments Hipikat with additional relations
  – Simple code relationships
  – Enhanced textual allusions
• Most pressing concern:
  – “Understanding the rationale behind a piece of code”
Bridge Artifacts
Bridge Evaluation

• Textual allusions == 19% of index
  – Source: 6 months of Windows development

<table>
<thead>
<tr>
<th>Mentioned Item Type</th>
<th>SCCS check-in</th>
<th>Email</th>
<th>Bug revision</th>
<th>Bug</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier</td>
<td>221,510</td>
<td>197,301</td>
<td>298,173</td>
<td>224,436</td>
<td>941,420</td>
</tr>
<tr>
<td>Number</td>
<td>309,403</td>
<td>209,630</td>
<td>89,139</td>
<td>66,431</td>
<td>674,603</td>
</tr>
<tr>
<td>Bug</td>
<td>81,680</td>
<td>20,440</td>
<td>3,563</td>
<td>1,826</td>
<td>107,509</td>
</tr>
<tr>
<td>HTTP URL</td>
<td>939</td>
<td>22,542</td>
<td>28,748</td>
<td>25,480</td>
<td>77,709</td>
</tr>
<tr>
<td>Local file path</td>
<td>495</td>
<td>35,729</td>
<td>10,164</td>
<td>6,458</td>
<td>52,846</td>
</tr>
<tr>
<td>Server file path</td>
<td>280</td>
<td>4,272</td>
<td>2,359</td>
<td>1,892</td>
<td>8,803</td>
</tr>
<tr>
<td>Email address</td>
<td>21</td>
<td>773</td>
<td>241</td>
<td>207</td>
<td>1,242</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>614,328</td>
<td>490,687</td>
<td>432,387</td>
<td>326,730</td>
<td>1,864,132</td>
</tr>
</tbody>
</table>
Deep Intellisense

- Holmes & Begel [MSR ‘08]
- Embed Hipikat-like functionality in VS
- Automatic updating; no queries required
Usage Scenario

```java
sb = new StringBuilder();
else if(trimmed.Length > 0) {
    sb.AppendLine();
    sb.AppendLine();
    sb.AppendLine(Environment.NewLine);
}
```
Deep Intellisense
Deep Intellisense

Deep Intellisense: Current Item

CKS.EBE  BlogContext

**BlogEnabled(...)**

**Checkin Overview:**
2 checkins between 11/29/2007 and 8/12/2008

**Work Item Overview:**
0 active work items (last activity 4/24/2008)
3 resolved since 6/9/2007
1 proposed work items

**People Overview:**
1 committer
3 bug reporters/ commenters
5 message authors
Deep Intellisense

Deep Intellisense: Related People

8 people shown.

- pointme  Member since 7/7/200
  0 work items  0 commits  1 message

- rdcpro  Member since 8/10/200
  0 work items  0 commits  1 message

- TheKid  Member since 1/29/200
  2 work items  2 commits  1 message
### Deep Intellisense

#### Event History

<table>
<thead>
<tr>
<th>Discussion #33905</th>
<th>rdcpro</th>
<th>Last year – Thu 8/21/2008 8:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EBE:</strong> Email address is collected when posting comments - where is email from?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change Set #16109</th>
<th>TheKid</th>
<th>Last year – Tue 8/12/2008 1</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Discussion #33119</th>
<th>pointme</th>
<th>Last year – Thu 8/7/2008 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EBE Error when deleting sites</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discussion #31112</th>
<th>danielmcc</th>
<th>Last year – Tue 7/15/2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failed to render control: An error occurred during a call to extension 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Work Item #6067</th>
<th>TheKid</th>
<th>Pri 1 Proposed Last year – Thu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Since activating the CKS:EBE feature on our Web</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Deep Intellisense Evaluation

• Focus was on the information developers wanted, not the resulting tool
• Rolled into CodeBook prototypes
Hatari

- Sliwerski et al. [FSE ‘05]
- Using past defects to predict future defects
- Identify fix-inducing changes
/* (non-Javadoc)
 * @see org.eclipse.jdt.launching.IRuntimeClasspathProvider#resolveClasspath
 */
public IRuntimeClasspathEntry[] resolveClasspath(IRuntimeClasspathEntry[] entries, ILaunchConfiguration configuration)
throws CoreException {
List all = new ArrayList();
for (int i = 0; i < entries.length; i++) {
  switch (entries[i].getType()) {
    case IRuntimeClasspathEntry.NATIVE:
      // a project resolves to itself for source lookup (rather than for classpath lookup)
      all.add(entries[i]);
      break;
    case IRuntimeClasspathEntry.TARGET:
      if (entries[i] instanceof ProjectClasspathEntry) {
        // add the resolved project
        IRuntimeClasspathEntry entry = ((ProjectClasspathEntry) entries[i]).getRuntimeClasspathEntry();
        all.add(entry);
      }
      break;
    case IRuntimeClasspathEntry.RUNTIME:
      for (int j = 0; j < entries[i].getChildren().length; j++) {
        all.add(entries[i].getChildren()[j]);
      }
      break;
    default:
      IRuntimeClasspathEntry[] va = resolveRuntimeClasspathEntry(entries[i], configuration);
      if (va != null)
        for (int j = 0; j < va.length; j++)
          all.add(va[j]);
      break;
  }
}
return (IRuntimeClasspathEntry[]) all.toArray(new IRuntimeClasspathEntry[all.size()]);
How to predict future risk

Correlation vs. Number of upcoming changes

McCabe Cyclomatic Complexity
How to predict future risk

![Graph showing the correlation between Nested Block Depth and McCabe Cyclomatic Complexity against the number of upcoming changes](image)
How to predict future risk

Correlation

Number of upcoming changes

Past Risk
Nested Block Depth
McCabe Cyclomatic Complexity
### Hatari: Risky Locations

<table>
<thead>
<tr>
<th>Location</th>
<th>Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolveClasspath</td>
<td>0.8888</td>
</tr>
<tr>
<td>run</td>
<td>0.8182</td>
</tr>
<tr>
<td>initializeProviders</td>
<td>0.7777</td>
</tr>
<tr>
<td>abort</td>
<td>0.7272</td>
</tr>
<tr>
<td>dispose</td>
<td>0.7000</td>
</tr>
<tr>
<td>init</td>
<td>0.6000</td>
</tr>
<tr>
<td>translate</td>
<td>0.5555</td>
</tr>
<tr>
<td>cleanup</td>
<td>0.5000</td>
</tr>
<tr>
<td>launch</td>
<td>0.4545</td>
</tr>
</tbody>
</table>
Hatari: Annotations

“Safe” Location (green)

Risky Location (dark red)
Task Specificity

• Many development activities are task based
  – Fix this bug
  – Add this feature

• Single tasks can involve many different data sources (files, documents, past changes etc.)

• Tasks have collaborative and temporal aspects
Mylyn

- Kersten and Murphy [FSE ‘06]
- Degree-of-interest model
  - Captures task context
  - Generated by observing navigation
- Connectors encourage adoption
  - Bug repositories
  - Version control
  - Tasks
Mylyn
Mylyn
Mylyn
Mylyn
Mylyn: Switching Contexts
Mylyn: Switching Contexts
Mylyn Evaluation

• Early DOI study (6 devs)
  – Tasks are *key*

• Monitor study (99 devs monitored)

• Mylyn study (16 devs from monitor study)
  – Significant increase in edit ratio

• Ultimately, we vote with our feet
  – Mylyn is very popular in the Eclipse ecosystem
TeamTracks

- DeLine et al. [VLHCC ‘05]
- “Pick the brain” of the original developer
- Two measures
  - Element is important if it is often visited
  - Two elements are related if visited in succession
Team Tracks
Team Tracks

Most related to GATetrisControl.TetrisGrid.InitFigure

- InitNextFigure()
- nextFig
- TetrisGrid()
- Figure.CanDraw()
- InitNewGame()
- Figure.DrawFigure()
TeamTracks Evaluation

• Increased chances of success (set tasks)
• Large increase in comprehension
  – 2x more likely to give the right answer
• Privacy concerns must be considered
• Scope navigation data by time?
  – Hints at task specificity
Code Bubbles

• Bragdon et al. [ICSE ‘10]
• Editable fragments rather than files
• Encourage task-based grouping
• Easily persist and share past tasks
  – Support context switching
Code Bubbles Evaluation

• Scrolling:
  – Decreased by ~49%

• Search / navigation:
  – Decreased by ~55%

• User study (20+ devs)
  – Generally positive, esp. about task-based features
  – Worried about scalability
Collaborative Development

• Software is developed in teams
• IDEs are typically designed for individual devs
• Collaboration external to IDE
  – Valuable data can be lost
  – Processes unnecessarily ad hoc
Jazz

• Li-Te Cheng [OOPSLA ’03]

• Team communication
  – Explicitly link artifacts rather than mine them
  – Build chat logs into historical data
  – Enable snapshots to be sent by IM

• Lifecycle integration
  – Integrated handling of builds & tests
  – Promote enhanced reporting etc.
Jazz

What if your tools know...

... *about* your teams

... *about* your artifacts

... who *is responsible for* what

... *about your* process
   Code delivery rules, code quality, traceability, test runs, intellectual property

... *how to* bootstrap a project

... *how to* help new team members *get* started

... your favorite work item types and *their* state transitions

... *when the* build runs and what to do if it breaks
Jazz Information Sources
Sample Jazz Workflow

1. A build breaks owing to a test failure
2. A developer creates a bug report
3. Jazz links the bug report to both the build and the failed test
4. Jazz assigns the bug to an appropriate dev.
5. The dev commits their change set
6. Jazz links the change set to the bug and to the build
Jazz Dashboard
Jazz Dashboard
FastDash

- Biehl et al. [CHI ‘07]
- Assumption: Developers want to know what their co-workers are doing
  - Maintain awareness about:
    - Files being edited
    - Task assignments
    - Bug assignments
- Targeted at large displays in common space
- Decreased need for explicit communication
FastDash
FastDash
FastDash Evaluation

• Shared resource contention decreased
Codebook

• Begel & DeLine [MSR ‘08]
• Extend social networks into software systems
  – E.g., developers can be ‘friends’ with their code
  – Social call graphs
    • Enable informal feedback channel for your APIs
• Provide alternative means for discovering time contention
• Evaluation not yet complete
**EventLogger.Connect() in EventLogger.cs**

In class `EventLogger` in `Microsoft.Research`
Compiled into `Logging.DLL`
16 checkins between 1/24/2005 and 1/31/2006
5 pri0 bugs, 10 pri1 bugs, 1 pri2 bug
3 sibling methods:
- void `OnConnection()`
- bool `Close()`
- void `OnFailure()`
2 sibling fields:
- int `numberConnections`
- bool `currentlyConnected`

Uses MAPI, OWA, and Passport external APIs.
Spec can be found in [http://team/sites/devui/docs/Logger.doc](http://team/sites/devui/docs/Logger.doc)

**Callers**
Called by 41 methods: See all
- `EventLogger.OnConnection()`: 3 calls
- `EventLogger.OnFailure()`: 2 calls
- `Recommender.Startup()`: 1 call
Code owned by 24 people calls `Connect()`:
- Mike Diaz, Jerry Ryan, Sumeet Gupta, Aaron Martin, Jenna Goldberg ... (see all)

**Related People**
2 committers, 3 bug reporters/commenters (see all)
- pialic RSDE MSR-Research 99/4219
- rdeline SENIOR RESEARCHER MSR-Research 99/2132
- sumeetg DEV LEAD 2 Windows 26/3012

**Newsfeed**

— March 2009 —
Pialic checked in #1181 (tfs) and marked bug #9902 (ps) as closed.
- changed methods `openLogFile()` and `Connect()` in class `Connect`
- Moved to `EventLogger` class from `OldEventLogger` class by pialic
- Modified by checkin #1181 (“BUG 9902…”) by pialic
- Mentioned in bug #9902 (“fails to connect…”) is pri 1 by abegel

— February 2009 —
- Mentioned in checkin #381 (“BUG 3384…”) by sumeetg
- Mentioned in email (“Failed to connect…”) from rdeline
- Mentioned in bug #3384 (“hang when…”) is pri 1 by ginav
- Mentioned in bug #1022 (“connects too slow…”) is pri 2 by pialic

— December 2008 —
- Added by checkin #211 (“ongoing…”) by pialic

**Gadgets**
Churn metrics
Get definition
Customized Awareness

- Holmes and Walker [ICSE ‘10]
- Projects are not created in isolation
  - Sometimes control is ceded to external teams
- Similar information spread across many projects
Heterogeneous Environments
Heterogeneous Environments
Large Teams
Large Teams

Makes Change
Large Teams

- Master SCM
- Branch SCM

2 week delay

Makes Change
Large Teams

2 week delay

Master SCM

Branch SCM

2 week delay

Branch SCM

Makes Change

Code now failing
Commit Logs

Subject: [r9g-cvs] [hg] refactoring.java: #144961: fixing NPE
From: Jan P. <jp... (at) netbeans.org>
Message-id: hg.abbaf0f1ff63.1219918286.1797694213 (at)
hg.netbeans.org
Date: 2008-08-28 12:11:26
changeset abbaf0f1ff63 in main
details: http://hg.netbeans.org/main?cmd=changes baf0f1ff63
description:
#144961: fixing NPE
diffs (12 lines):
diff -r eb7c38562250 -r abbaf0f1ff63
a/refactoring.java/src/org/netbeans/modules/refactoring/java/ui/UIUtilities.java
b/refactoring.java/src/org/netbeans/modules/refactoring/java/ui/UIUtilities.java
Aug 28 12:03:58 2008 +0200
@@ -92,7 +92,7 @@ public final class UIUtilities {
    headerRenderer = table.getTableHeader().
    getDefaultRenderer();
}

    Component comp = headerRenderer.
    getTableCellRendererComponent(
-   null, column.getHeaderValue(), false, false, 0, 0);
+   table, column.getHeaderValue(), false, false, 0, 0);
    int width = comp.getPreferredSize().width;
Commit Log Volume

<table>
<thead>
<tr>
<th>System</th>
<th>msg / work day</th>
</tr>
</thead>
<tbody>
<tr>
<td>KDE</td>
<td>515</td>
</tr>
<tr>
<td>Open Office</td>
<td>505</td>
</tr>
<tr>
<td>NetBeans</td>
<td>353</td>
</tr>
<tr>
<td>Linux Kernel</td>
<td>157</td>
</tr>
</tbody>
</table>
Concrete Example

- Eclipse Metrics plug-in
  - Depends on 9xEclipse, 7xApache, 1xSF
  - Only uses a minority of each
Concrete Example

- Eclipse Metrics plug-in
  - Depends on 9xEclipse, 7xApache, 1xSF
  - Only uses a minority of each

88% Classes
94% Methods
98% Fields
Approach

• Infer interest set
  – Code ownership + static analysis

• Analyze changes
  – Identify changed elements

• Determine change relevance
  – Structural relevance
  – Practical relevance
Evaluation

- RQ1: How well is the stream compressed?
- RQ2: Are impactful events really impactful?
- RQ3: Are any impactful events misclassified?

<table>
<thead>
<tr>
<th></th>
<th>Impactful</th>
<th>Uncertain</th>
<th>Information</th>
<th>Irrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics (sauerf)</td>
<td>9</td>
<td>674</td>
<td>273</td>
<td>618,219</td>
</tr>
<tr>
<td>Checkstyle</td>
<td>6</td>
<td>604</td>
<td>266</td>
<td>381,788</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>TP</th>
<th>FP</th>
<th>TN</th>
<th>FN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics</td>
<td>10</td>
<td>2</td>
<td>619,163</td>
<td>0</td>
</tr>
<tr>
<td>Checkstyle</td>
<td>5</td>
<td>0</td>
<td>382,658</td>
<td>1</td>
</tr>
</tbody>
</table>
Visualizations

• Seek to provide unique insights about systems
  – Task-specific or general ‘understanding’?
• Mappings to physical analogs difficult
SeeSoft
RIGI
AspectBrowser
AspectBrowser
Tarantula
Software Terrain Maps

- DeLine [VLC 2005]
- Navigate perceptually rather than cognitively
- Ease navigation esp. backtracking
- Region size corresponds to code size
- Region locations capture affinity
Software Terrain Maps
Software Cartography
Tesseract
Tesseract
Code City
Summary

• Tools help developers do _something_
• The path from a research prototype to an industrial tool is convoluted at best
  – Start with an idea that *could* be useful in practice
• Evaluation mismatch:
  – Academic merit vs. industrial merit