

## Objectives

- Tools for Finding Concerns
  - Dora

May 11, 2009

Sprenkle - CS297

## Review

- What is a problem for developers to perform maintenance tasks?
- What are some tools to address this problem?
  - What approach do the tools take?
- What are the tools' strengths and limitations?

May 11, 2009

Sprenkle - CS297

"Exploring the neighborhood with Dora to expedite software maintenance"

Gibson Hill, Pollock, Vijay-Shankar  
ASE 2007

## DORA

May 11, 2009

Sprenkle - CS297

## Reviewing Paper

- Problem?
  - Goals
- Approach?
  - Benefits

May 11, 2009

Sprenkle - CS297

## Program Exploration for Maintenance



May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Running Example Scenario

eBay auction sniping (bidding) program has bug in **add auction** event trigger

- **Exploration Task:** Locate code related to **'add auction'** trigger
- **Starting point:** **DoAction()** method, from prior knowledge
  - Handles all user-triggered events

May 11, 2009

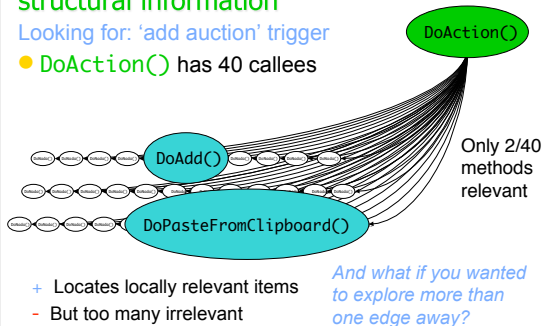
Sprenkle - CS297

Source: Hill ASE 2007

## Exploring with only structural information

Looking for: 'add auction' trigger

- DoAction() has 40 callees



May 11, 2009

Sprengle - CS297

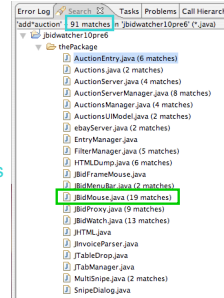
Source: Hill ASE 2007

## Alternative: Exploring with only lexical information

Looking for: 'add auction' trigger  
in 1902 methods (159 files, 23KLOC)

- Use lexical information from comments & identifiers
- Search with query 'add\*auction'
- 91 query matches in 50 methods
- Only 2/50 methods are relevant

- + Locates globally relevant items
- But too many irrelevant



May 11, 2009

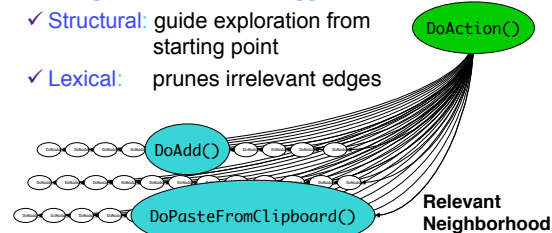
Sprengle - CS297

Source: Hill ASE 2007

## Dora gets it right...

Looking for: 'add auction' trigger

- ✓ **Structural:** guide exploration from starting point
- ✓ **Lexical:** prunes irrelevant edges



May 11, 2009

Sprengle - CS297

Source: Hill ASE 2007

## Software Maintenance: Dora to the rescue

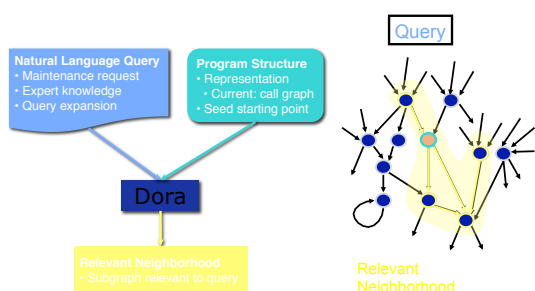
- Developers spend more time finding and understanding code than actually fixing bugs [Kersten & Murphy 2005, Ko et al. 2005]
  - Critical need for automated tools to help developers explore and understand today's large & complex software
- ➔ **Key Contribution:** Automated tools can use **program structure** and **identifier names** to save the developer time and effort

May 11, 2009

Sprengle - CS297

Source: Hill ASE 2007

## Dora the Program Explorer\*



\* Dora comes from *exploradora*, the Spanish word for a female explorer.

May 11, 2009

Sprengle - CS297

Source: Hill ASE 2007

## The Dora Approach

Prune irrelevant structural edges from seed

- Obtain set of methods one call edge away from seed
- Determine each method's relevance to query  
Calculate lexical-based relevance score
- Prune low-scored methods from neighborhood, using threshold
- Recursively explore

May 11, 2009

Sprengle - CS297

Source: Hill ASE 2007

## Calculating Relevance Score: Term Frequency

Query: 'add auction'

- Score based on number of occurrences of query terms in the method
- Intuition:** The more query terms in a method, the more likely it is relevant

```
private void DeAdd(Component src) {
    String endResult;
    String prompt = "Enter the auction number";
    endResult = promptString(src, prompt, "Adding");
    // They closed the window or cancelled.
    if (endResult == null) return;
    endResult = endResult.trim();
    MqFactory.getConcrete("user").enqueue(ADD_AUCTION + endResult);
}

private void DeleteComment(AuctionEntry ae) {
    if (ae == null) {
        ErrorMessage.logMessage("Auction selected to delete");
        return;
    }
    ae.deleteComment("");
    FileManager.getInstance().redrawEntry(ae);
}
```

6 query term occurrences

Only 2 occurrences

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Calculating Relevance Score: Inverse Document Frequency

Query: 'add auction'

- What about terms that appear all over the program?
- Use **inverse document frequency (idf)**
  - Intuition:** Highly weight terms that appear in few documents/methods
    - Terms appearing all over program not good discriminators
    - Don't separate relevant from irrelevant methods

= Number of methods / number of methods containing the term

1902 Methods

```
private void DeAdd(Component src) {
    String endResult;
    String prompt = "Enter the auction number to add";
    endResult = promptString(src, prompt, "Adding");
    // They closed the window or cancelled.
    if (endResult == null) return;
    endResult = endResult.trim();
    MqFactory.getConcrete("user").enqueue(ADD_AUCTION + endResult);
}
```

public idf = 1902/1311 = 1.45  
 auction idf = 1902/415 = 4.58  
 add idf = 1902/258 = 7.37  
 password idf = 1902/29 = 65.59

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Calculating Relevance Score: TF-IDF

- Score based on method query term frequency (**tf**)
- Multiplied by natural log of inverse document frequency (**idf**)

Query: 'add auction'

```
private void DeAdd(Component src) {
    String endResult;
    String prompt = "Enter the auction number";
    endResult = promptString(src, prompt, "Adding");
    // They closed the window or cancelled.
    if (endResult == null) return;
    endResult = endResult.trim();
    MqFactory.getConcrete("user").enqueue(ADD_AUCTION + endResult);
}

private void DeleteComment(AuctionEntry ae) {
    if (ae == null) {
        ErrorMessage.logMessage("Auction selected to delete");
        return;
    }
    ae.deleteComment("");
    FileManager.getInstance().redrawEntry(ae);
}
```

6 query term occurrences

tf-idf = 4 \* ln(7.37) + 2 \* ln(4.58) = 11.03

Only 2 occurrences

tf-idf = 2 \* ln(4.58) = 3.04

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Calculating Relevance Score: What about location?

- Weigh term frequency (**tf-idf**) based on location:
  - Method name more important than body
  - Method body statements normalized by length

Query: 'add auction'

```
private void DeAdd(Component src) {
    String endResult;
    String prompt = "Enter the auction number to add";
    endResult = promptString(src, prompt, "Adding");
    // They closed the window
    if (endResult == null) return;
    endResult = endResult.trim();
    MqFactory.getConcrete("user").enqueue(ADD_AUCTION + endResult);
}

private void DoPasteFromClipboard() {
    String auctionId = getClipboardString();
    if (auctionId != null) {
        MqFactory.getConcrete("user").enqueue(ADD_AUCTION + auctionId);
    }
}
```

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Dora's Relevance Score

- Factors
  - $\sum$  tf-idf for each query term in the *method name*
  - $\sum$  tf-idf for each query term in the *method body*  
the number of statements in the method
  - Binary methods: a library?
- How to determine weights?
  - Applied logistic regression
  - Trained on methods from 9 concerns in previous concern location tool evaluation [Shepherd et al. 2007]

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Logistic Regression

- Train on methods with known relevance to a concept

Method	Name	Statement	Binary	IsRelevant?
1	11.5	2	Y	No
2	6.5	10.1	N	Yes
3	2.5	7.5	N	No
...				

- Figure out weights to put on factors so that we can predict if the method is relevant

Depends on query?

Weights change if query terms change?

May 11, 2009

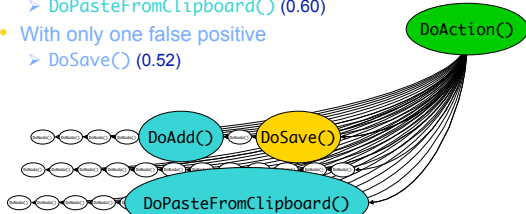
Sprenkle - CS297

**Example:**

Dora explores 'add auction' trigger

Scores from DoAction() seed:

- Identified as relevant with 0.5 threshold
  - DoAdd() (0.93)
  - DoPasteFromClipboard() (0.60)
- With only one false positive
  - DoSave() (0.52)



May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

**"Source Code Exploration with Google"**

- Problem?
  - Limitations of state of the art
  - Goals
- Approach?
  - Benefit?
- Evaluation?
  - Results?

May 8, 2009

Sprenkle - CS297

**Experimental Evaluation:  
Research Questions**

- Does an integrated lexical- and structural-based approach outperform a purely structural approach?
- Is a sophisticated lexical scoring technique required, or are naïve lexical scoring techniques sufficient to identify the relevant neighborhood?

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

**Experimental Evaluation:  
Design**

- Gold Set:** 8 concerns from 4 Java programs, manually mapped by 3 independent developers [Robillard et al. 2007]
- Compare** 4 exploration techniques: 1 structural, 3 lexical + structural
  - Structural:** Suade [Robillard 2005]
    - Automatically generates exploration suggestions from seed set
    - Elements that have few connections outside the seed set are more relevant
    - Uses caller/callee & field def-use information to make recommendations
  - Lexical + Structural:** Dora (sophisticated)
  - Lexical + Structural:** boolean AND (naïve)
  - Lexical + Structural:** boolean OR (naïve)

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

**Experimental Evaluation:  
Design**

- Gold Set:** 8 concerns from 4 Java programs, manually mapped by 3 independent developers [Robillard et al. 2007]
- Compare** 4 exploration techniques: 1 structural, 3 lexical + structural
- Measures:** Precision (P), Recall (R), & F Measure (F)
  - $P = \frac{TP}{TP+FP}$  (Are the results returned actually relevant?)
  - $R = \frac{TP}{TP+FN}$  (How close are the returned results to the gold set?)
  - $F = \frac{2PR}{P+R}$  (High when P & R are similarly high)

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

**Experimental Evaluation:  
Design**

- Gold Set:** 8 concerns from 4 Java programs, manually mapped by 3 independent developers [Robillard et al. 2007]
- Compare** 4 exploration techniques: 1 structural, 3 lexical + structural
- Measures:** Precision (P), Recall (R), & F Measure (F)
- Methodology**
  - For each exploration technique  $t$ 
    - For each method  $m$  in the gold set
      - Score each caller & callee of  $m$  with  $t$
      - Calculate P, R, & F for  $m$  with  $t$
  - 160 seed methods, 1885 call edges (with overlap)

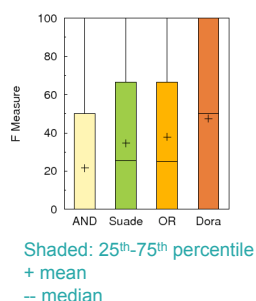
May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Results: All Concerns

- Dora outperforms Suade with statistical significance ( $\alpha = 0.05$ )
- Dora, OR, and Suade perform significantly better than AND
- Dora and Suade not significantly different from OR ( $\alpha = 0.05$ )
  - OR > Suade,  $p = 0.43$
  - Dora > OR,  $p = 0.033$
  - Dora > Suade,  $p = 0.0037$
- Dora achieves 100% P & R (exact gold set) for 25% of the data—more than any other technique

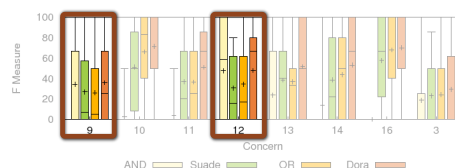


May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Results: By Concern



- Overall trend also seen for most concerns
- Exceptions: 9 & 12
  - AND had much higher precision
  - Relevant methods contained both query terms

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Experimental Evaluation: Result Summary

- Does an integrated lexical- and structural-based approach (*Dora*) outperform a purely structural approach (*Suade*)?
    - Dora outperforms Suade with statistical significance ( $\alpha = 0.05$ )
  - Is a sophisticated lexical scoring technique required, or are naïve lexical scoring techniques sufficient to identify the relevant neighborhood?
    - Although not statistically significant, Dora outperforms OR
    - Dora, Suade, & OR outperform AND ( $\alpha = 0.05$ )
- Integrated lexical- and structural-based approaches can outperform purely structural, but not all lexical scoring mechanisms are sufficient to do so

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## "Source Code Exploration with Google"

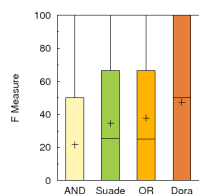
- Problem?
  - Limitations of state of the art
  - Goals
- Approach?
  - Benefit?
- Evaluation?
  - Results?
- Limitations?
- Conclusions

May 8, 2009

Sprenkle - CS297

## Conclusion

→ Integrated lexical- and structural-based approaches outperform purely structural ones



[www.cis.udel.edu/~hill/dora](http://www.cis.udel.edu/~hill/dora)

This work was supported by an NSF Graduate Research Fellowship and Award CCF-0702401.

May 11, 2009

Sprenkle - CS297

Source: Hill ASE 2007

## Limitations

- Relies on method, variable names
- Alternate relevance scores?
- How easy is it to find the seed/starting point?

May 11, 2009

Sprenkle - CS297

## Future Work

- Automatically find starting **seeds**
- Use more **sophisticated lexical information**
  - Synonyms, topic words (currency, price related to bidding)
  - Abbreviation expansion
- Evaluate on **slicing**

May 11, 2009

Sprenkle - CS297

## Dora Installation

1. From the Eclipse menu bar, select Help | Software Updates | Available Software
2. Select "Add Site..."
3. Enter in the URL, "http://www.cis.udel.edu/~gibson/dora/download/" and hit OK
4. Select the Dora plug-in and hit Install..., then click Next
5. Accept the terms of the license agreement and click Finish
6. Restart your workspace
7. Go to Window | Preferences | Dora and make sure the 'dot:' preference points to your installation
  - set to /usr/bin/dot by default.
  - which dot

May 11, 2009

Sprenkle - CS297

## Open Discussion

- What do you think of this tool compared to the others we discussed on Friday?
- Would you use the tools differently? Under different circumstances?

May 11, 2009

Sprenkle - CS297

## Why So Much Time on Search?

- Seemed like it was solved
  - We have "Find", "Grep"
  - Problems caused by larger code bases
- Different perspectives on a problem/solution
  - Be aware of what authors are telling you and what they aren't
- Current research
  - NLP tools do not work for software-specific synonyms

May 11, 2009

Sprenkle - CS297

## Merging Software

- Who has experience with CVS? With Subversion?

May 11, 2009

Sprenkle - CS297

## Reminder: Midterm Wednesday

- 15% of grade
- Focus on UNIX commands, Bash scripting
  - UNIX philosophy
  - Reading and writing Unix commands
  - Understand purpose of various tools
- Software tools
  - What can they do?
- Tool types we've covered so far
  - Build tools
  - Search/navigation tools

May 11, 2009

Sprenkle - CS297