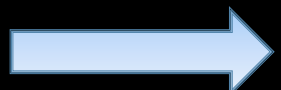
Data Science in Software Engineering

Baishakhi Ray
University of Virginia

http://rayb.info/
rayb@virginia.edu

Most slides are taken from **Tao Xie and Miryung Kim**

New Era…Software itself is changing…

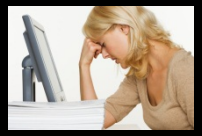


Software

Services

How people use software is changing…

Individual



Isolated

Not much content generation

Social

Collaborative

Huge amount of artifacts generated anywhere anytime

How software is built & operated is changing…

Code centric	Data pervasive
In-lab testing	Debugging in the large
Experience & gut-feeling	Informed decision making
Centralized development	Distributed development
Long product cycle	Continuous release
…	…

The Secret for Software Decision Making

- Which software or its property to use?
- How to improve your software?
- How to write better code?
- How to efficiently debug your code?
- Which code we should test?
- Which project to join?
- Whom to recruit?
- …

‘Big’ Software Data
Use Data Science to find the answers

Data Science in Software Engineering

Manager

Project Architect

Developer

Tester

User

Record all project related activities and archive it

Software Archive

Code | Bug | E-Mail/Chat | User Reviews | Others

Data Science in Software Engineering

Manager

Project Architect

Developer

Tester

User

Record all project related activities and archive it

stackoverflow

GitHub

Bugzilla

Bitbucket

App Store

Google play

Apache SOFTWARE FOUNDATION

Data Science in Software Engineering

Analyze software data

Make informed data-driven decisions

Supporting decision making using facts instead of fortune tellers!

Software Archive

Data sources

Runtime traces
Program logs
System events
Performance counters
…

Usage log
User surveys
Online forum posts
Blog & Twitter
…

Source code
Bug history
Check-in history
Test cases
…

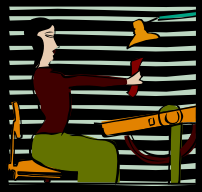
Target audience – software practitioners

Program Manager

Developer

Management personnel

Designer



Tester

Support engineer

Operation engineer

Usability engineer

Output – insightful information

- Conveys *meaningful* and *useful* understanding or knowledge towards completing the target task
- Not easily attainable via directly investigating raw data without aid of *analytics technologies*
- Example
  - It is easy to count the number of re-opened bugs, but how to find out the primary reasons for these re-opened bugs?

Output – actionable information

- Enables software practitioners to come up with *concrete solutions* towards completing the target task
- Examples
  - Why bugs were re-opened?
    - A list of bug groups each with the same reason of re-opening
  - Which part of my code should be refactored?
    - A list of cloned code snippets easily explored from different perspectives

Few Examples!!

Leveraging Big Software Data to Improve Software Quality

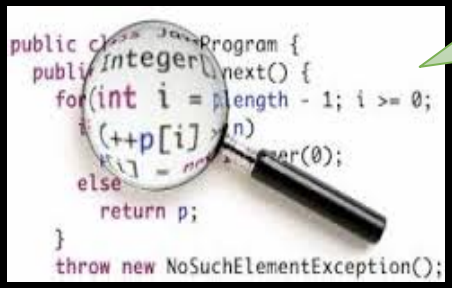
PL/SE research effort to reduce bugs

Languages

Type System,
Memory Management

Assertions (invariant checking), Code reuse

Best Coding Practices



Automatic Bug Finding Tools

Program Analysis,
Testing

Code Reviews
Development
Processes

Team Process

Do we know the answers ?

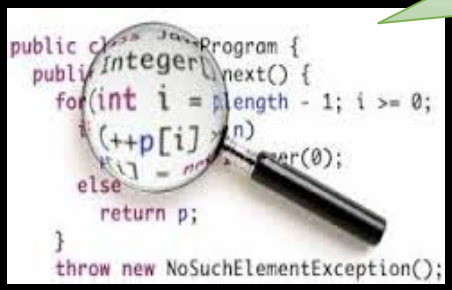
Languages

Does a choice of language affect code quality?

What kinds of bugs are caused by copy-paste?

Best Coding Practices

Do automatically generated unit tests find real faults?



Automatic Bug Finding Tools

How does API evolution affect code quality?

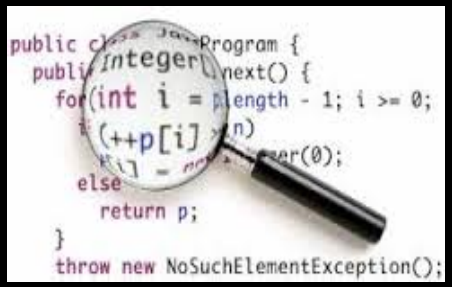
Team Process

`Big' Software Data

Languages

Use data science to find the answers

Best Coding Practices



Automatic Bug Finding Tools

Get Insights for future directions

Team Process

Empirical Findings

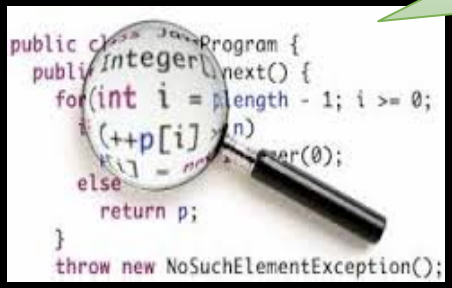
Languages

A choice of language matters more for specific error categories than it does for overall defects [FSE'14]

Incorrect adaptation of copied code introduces bugs [ASE'13]

Best Coding Practices

Automatically generated tests effectively find real faults [ASE'15]



Automatic Bug Finding Tools

Aggressive API update leads to bugs and delayed adoption in client code [ICSM'2013]

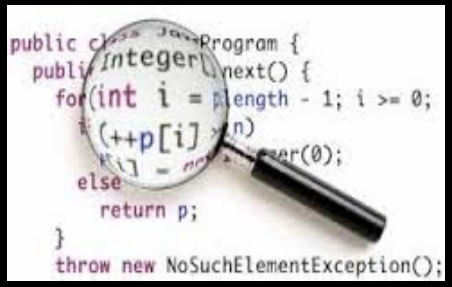
Team Process

Develop new techniques based on Empirical Findings

Languages

Design new algorithms and build tools (e.g. Static analysis tools, bug prediction tools, testing strategies) that can address the empirically found problems.

Best Coding Practices

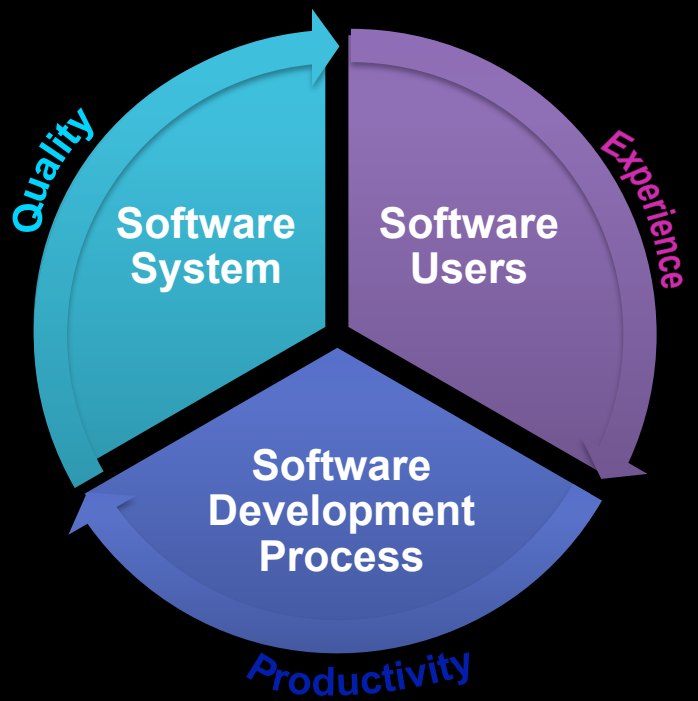


Automatic Bug Finding Tools

Team Process

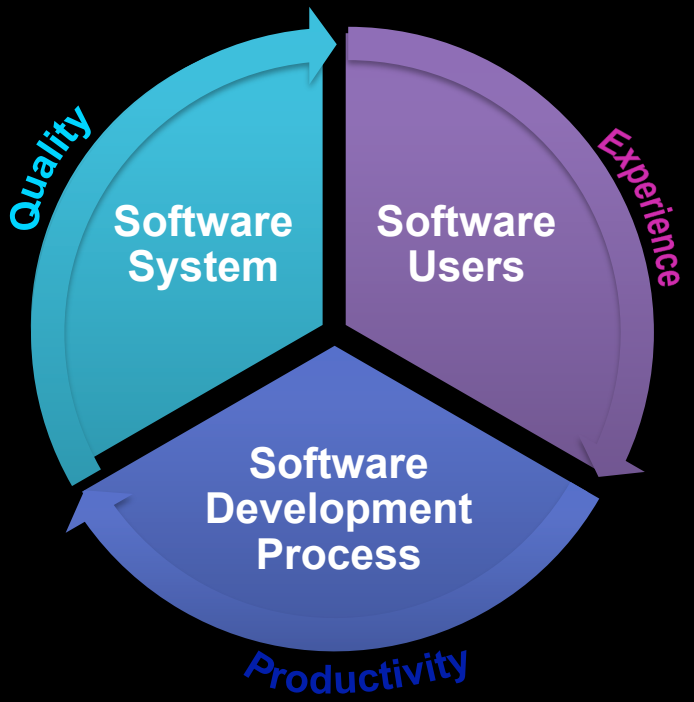
Research Topics

Research topics



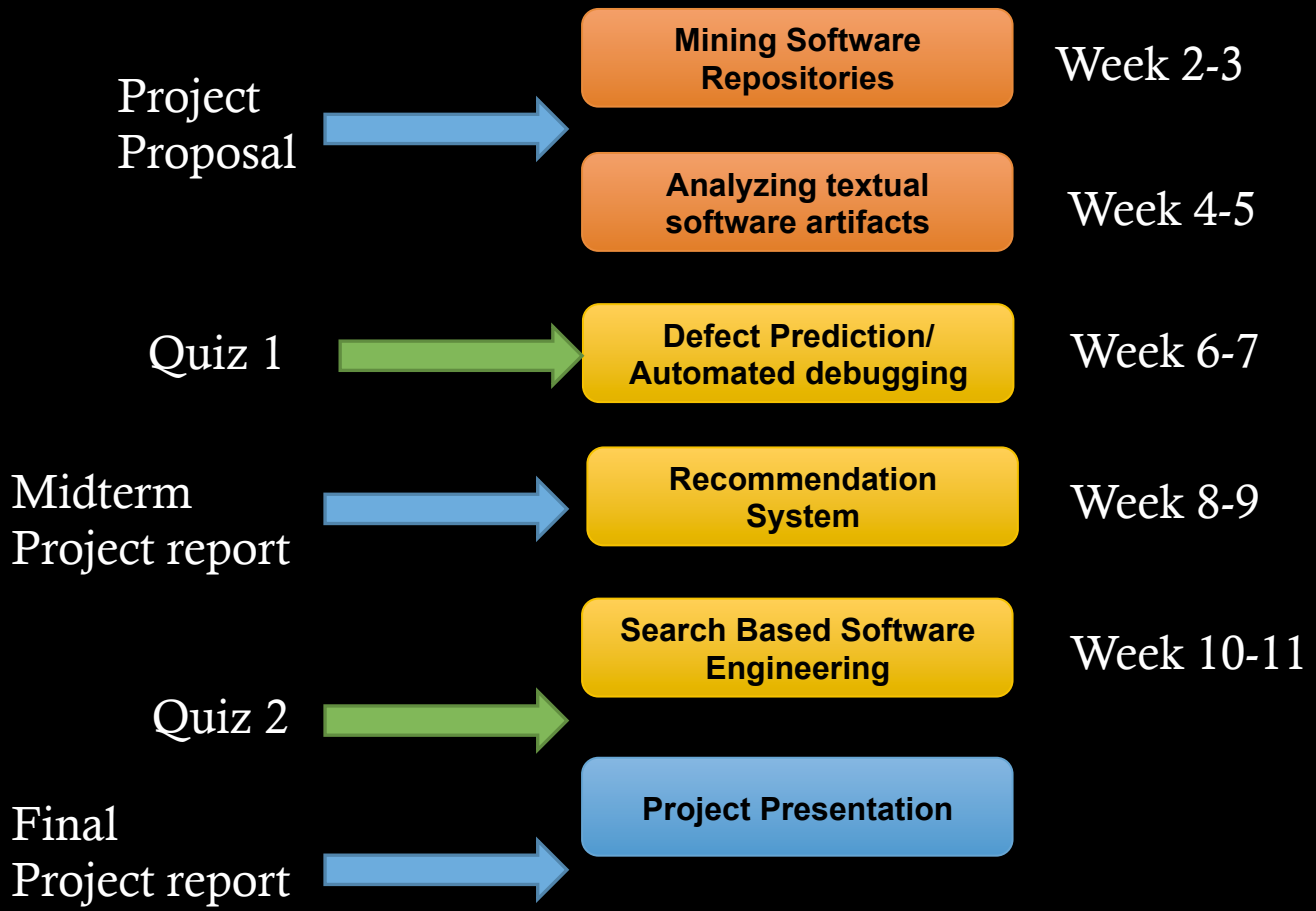
- Covering different areas of software domain
- Throughout entire development cycle
- Enabling practitioners to obtain insights

Research topics



- Technique
  - Mining Software Repositories
  - Analyzing textual software artifacts
- Application
  - Defect Prediction/ Automated debugging
  - Recommendation System
  - Search Based Software Engineering

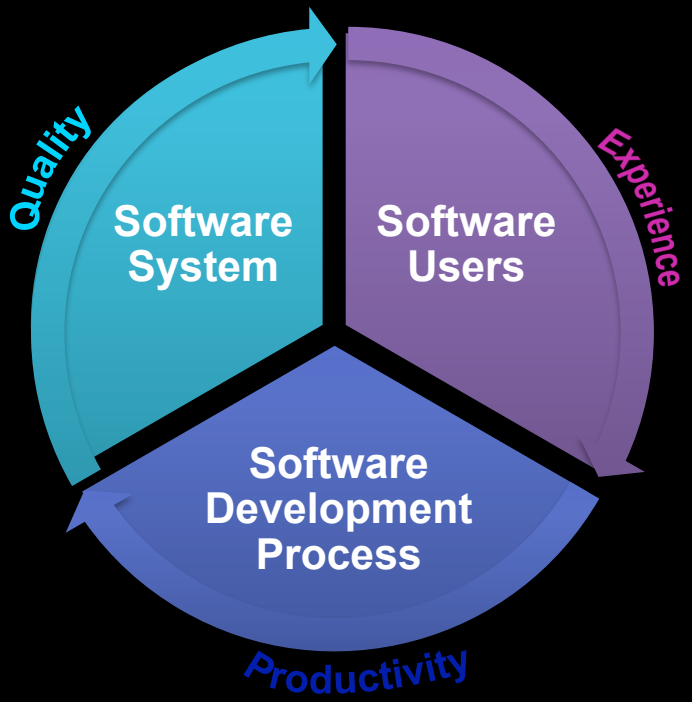
Tentative Course Layout



Grading Policy

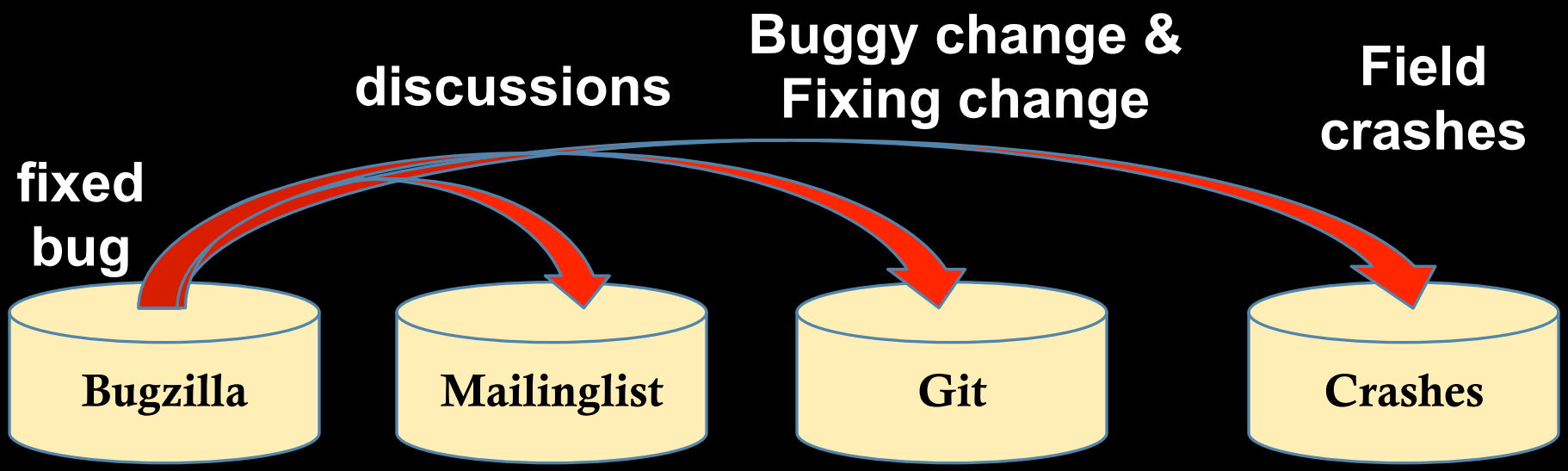
- Group Project (2-3 students) – 50%
  - End of the semester presentations/demo – 15%
  - Project Report : (proposal – 5%, mid-term report – 5%, end report – 25%)
- Quiz – 25%
- Class Participation – 25%
  - Present a paper (you may present informally) – 5%
  - Will randomly ask questions – 15%
  - After class send a note: New idea, how the paper can be extended – 5%

Research topics



- Technique
  - Mining Software Repositories
  - Analyzing textual software artifacts
- Application
  - Defect Prediction/ Automated debugging
  - Recommendation System
  - Search Based Software Engineering

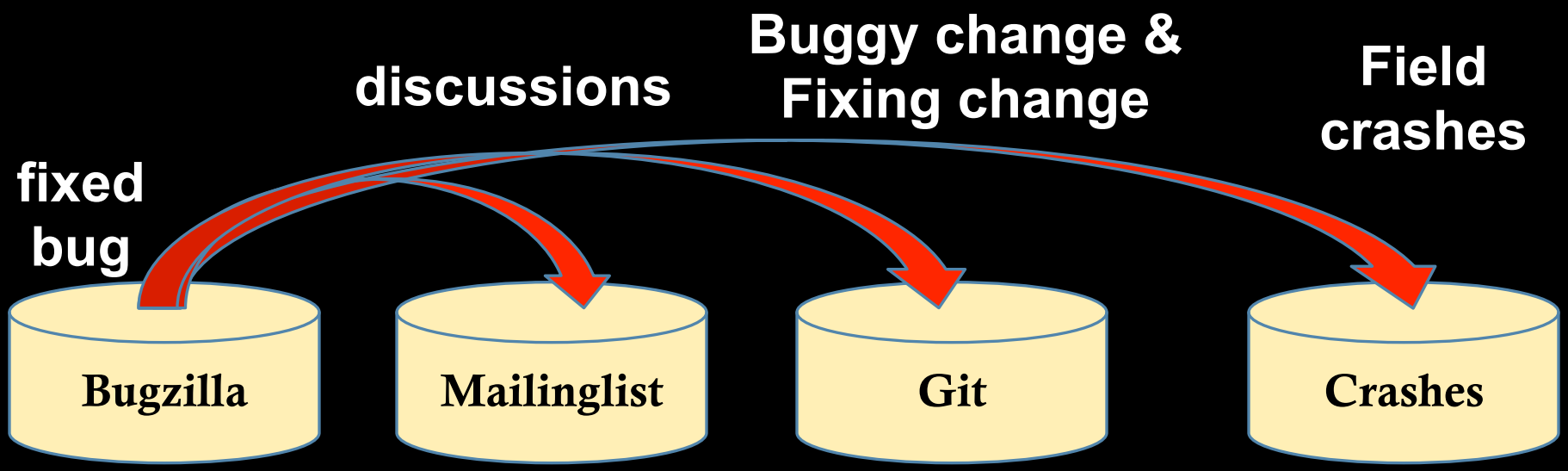
Example: Mining Software Repositories



When a new bug is reported

- Estimate fix effort
- Mark duplicates
- Suggest experts and fix!

Example: Mining Software Repositories



When code is changed

- Suggest APIs
- Warn about risky code or bugs
- Suggest locations to co-change

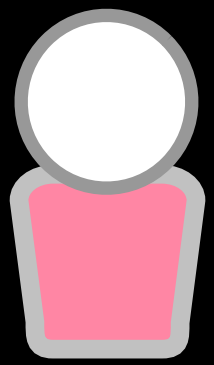
Example: Analyzing Textual Software Artifacts

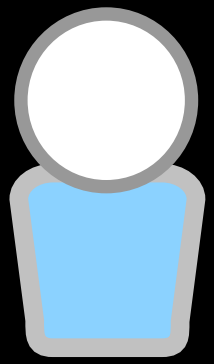
Types of Textual Software Artifacts

- requirement documents
- code comments
- identifier names
- commit logs
- release notes
- bug reports
- …
- emails discussing bugs, designs, etc.
- mailing list discussions
- test plans
- project websites & wikis
- Question answer cites (hybrid)

Text data contains useful information, much of which is not in structured data.

Example: code comment contains *Specification*





linux/drivers/scsi/in2000.c:

```
/* Caller must hold instance lock! */
static int reset_hardware(…)
{…}
```

linux/drivers/scsi/in2000.c:

```
static int in2000_bus_reset(…){    …
    reset_hardware(…);
    …
}
```

No lock acquisition ⇒ A bug!

Tan et al. "/*iComment: Bugs or Bad Comments?*/", SOSP'07

Example: contains *semantics of identifiers*

```
noFirewall = new JRadioButton("No firewall or proxy");
socksFirewall = new JRadioButton("SOCKS 4/5 Firewall");
webProxy = new JRadioButton("HTTP Web Proxy");

allButtons = new ButtonGroup();
allButtons.add(socksFirewall);
allButtons.add(webProxy);
allButtons.add(noFirewall);
socksFirewall.addActionListener(rad);
webProxy.addActionListener(rad);
noFirewall.addActionListener(rad);
```

Create RadioButtons

Add RadioButtons to allButtons

Add radioActionListener to RadioButtons

Sridhara, Pollock, Vijay-Shanker. Automatically Detecting and Describing High Level Actions within Methods. ICSE 2011

Challenges in Analyzing Textual Data

- Unstructured
  - Hard to parse, sometimes wrong grammar
- Ambiguous: often has no defined or precise semantics (as opposed to source code)
  - Hard to understand
- Many ways to represent similar concepts
  - Hard to extract information from

```
/* We need to acquire the write IRQ lock before calling ep_unlink(). */
/* Lock must be acquired on entry to this function. */
/* Caller must hold instance lock! */
```

Why Analyzing Textual Data is Easy(?)

- Redundant data
- Many techniques to borrow from text analytics: NLP, Machine Learning (ML), Information Retrieval (IR), etc.

Stackoverflow Question

stackoverflow

Pex ignores default parameter assignment

0

I am using Pex to analyse function executions. However, I noticed that default parameters are not looked at.

Here's an example of what I mean:

```
public int bla(int x = 2)
{
    return x * 2;
}
```

When I run Pex, it generates the test case for `int result = bla(0);`. (`x = 0`)
Is there a way to tell Pex that it should also try to call `bla(` without parameter (i.e. `int result = bla()`)?

visual-studio pex pex-and-moles

share improve this question

asked Sep 16 at 9:47
S.K.

Challenge: Detect Duplicate Post

stackoverflow

Pex ignores default parameter assignment

0

I am using Pex to analyse function executions. However, I noticed that default parameters are not looked at.

Here's an example of what I mean:

```
public int bla(int x = 2)
{
    return x * 2;
}
```

When I run Pex, it generates the test case for `int result = bla(0);`. (`x = 0`)
Is there a way to tell Pex that it should also try to call `bla(` without parameter (i.e. `int result = bla()`)?

visual-studio pex pex-and-moles

share improve this question

asked Sep 16 at 9:47
S.K.

Challenge: Assign Post to Whom?

stackoverflow

Pex ignores default parameter assignment

0

I am using Pex to analyse function executions. However, I noticed that default parameters are not looked at.

Here's an example of what I mean:

```
public int bla(int x = 2)
{
    return x * 2;
}
```

When I run Pex, it generates the test case for `int result = bla(0);`. (`x = 0`)
Is there a way to tell Pex that it should also try to call `bla(` without parameter (i.e. `int result = bla()`)?

visual-studio pex pex-and-moles

share improve this question

asked Sep 16 at 9:47
S.K.

Challenge: Identify High Severity Post

stackoverflow

Pex ignores default parameter assignment

0

I am using Pex to analyse function executions. However, I noticed that default parameters are not looked at.

Here's an example of what I mean:

```
public int bla(int x = 2)
{
    return x * 2;
}
```

When I run Pex, it generates the test case for `int result = bla(0);`. (`x = 0`)
Is there a way to tell Pex that it should also try to call `bla(` without parameter (i.e. `int result = bla()`)?

visual-studio pex pex-and-moles

share improve this question

asked Sep 16 at 9:47
S.K.

Example Bugzilla Bug Report

Bugzilla Bug 338009 Browser Crashes at cbs.com Last modified: 2006-05-15 09:27:44 PDT

Bug List: (15 of 37) First Last Prev Next Show last search results Search page Enter new bug

Bug#: 338009 **alias:**
Product: Firefox
Component: General
Status: UNCONFIRMED
Resolution:
Assigned To: Nobody's working on this, feel free to take it <nobody@mozilla.org>

Hardware: Macintosh
OS: Mac OS X 10.4
Version: unspecified
Priority: --
Severity: normal
Target Milestone: ---

Reporter: Mark <mozilla@mark-miller.com>
Add CC:
CC:

Assigned To: ?

Description: [reply] **Opened:** 2006-05-15 09:21 PDT

Each time I visit http://www.cbs.com/, Firefox crashes before the page is loaded. I can tell what element of the page is crashing the browser though.

Duplicate?

```
Reproducible: Always

Steps to Reproduce:
1.Open Browser
2.Enter http://www.cbs.com/
3.Press return

Actual Results:
Page starts to load, and then crashes.

Expected Results:
The browser doesn't crash.

No other sites so far have displayed this behavior.
```

Anvik, Hiew, Murphy. Who should fix this bug? ICSE 2006.
Wang, Zhang, Xie, Anvik, Sun. An Approach to Detecting Duplicate Bug Reports using Natural Language and Execution Information. ICSE 2008.

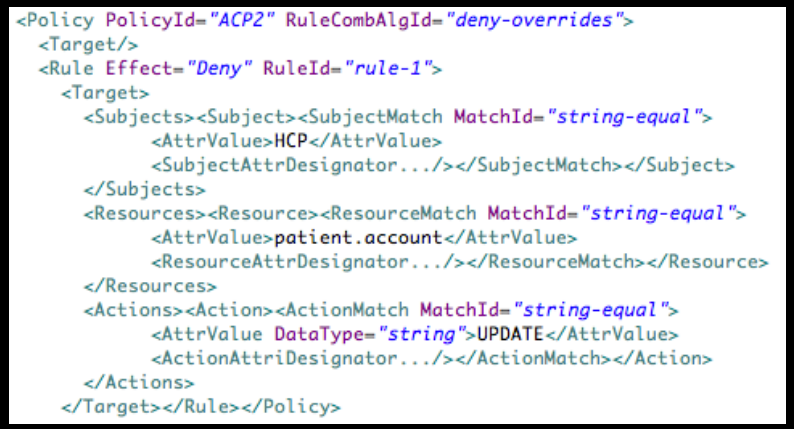
From Requirement Text to Formal Security Policy

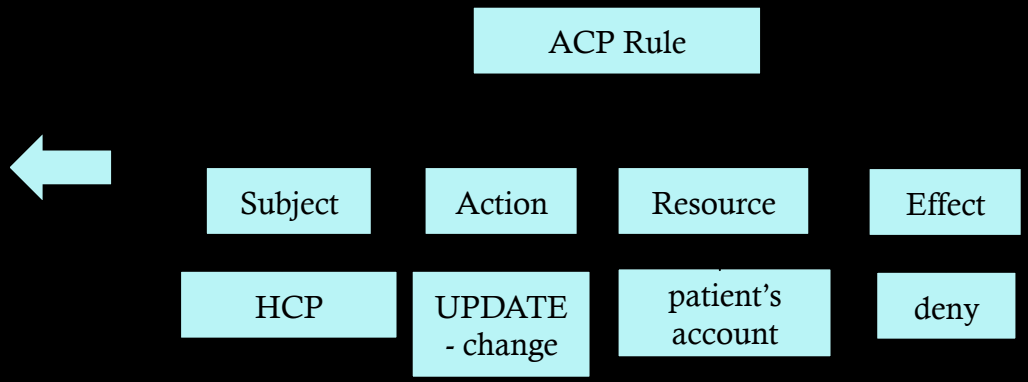
Linguistic Analysis

A HCP should not change patient's account.

An [*subject*: HCP] should not [*action*: change] [*resource*: patient's account].

Model-Instance Construction





Transformation

Xiao, Paradkar, Thummalapenta, Xie. Automated Extraction of Security Policies from Natural-Language Software Documents. FSE 201

You might be surprised!!

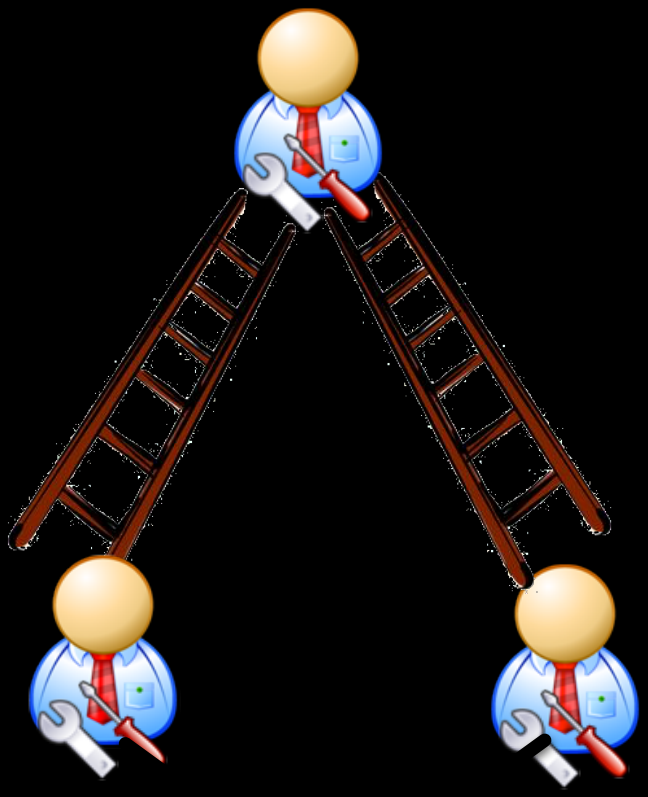
Should I test\review my?

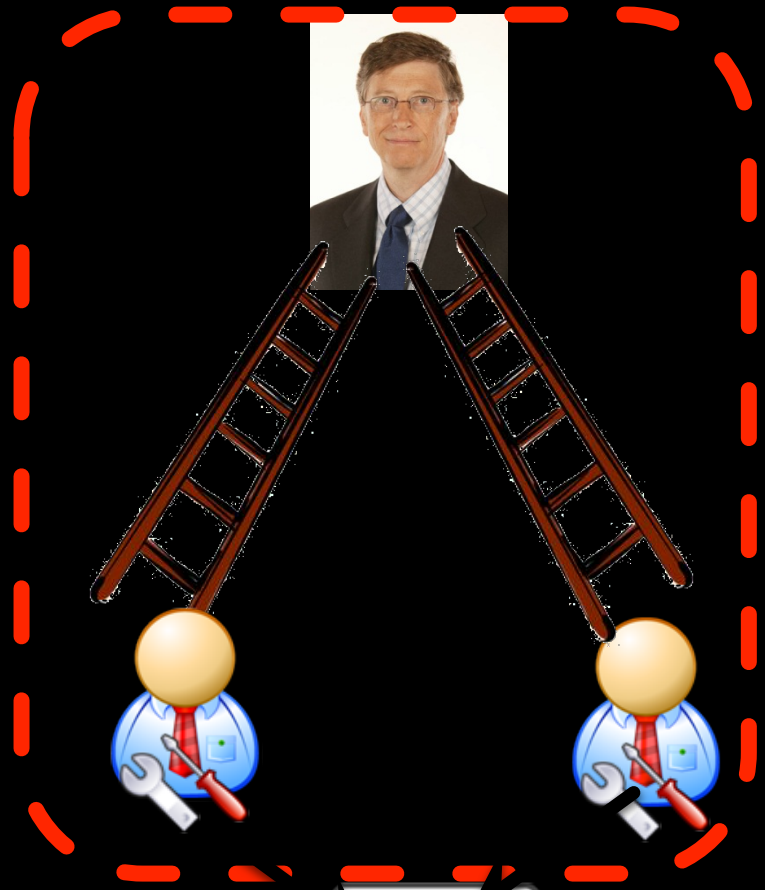
A. Ten *most-complex* functions

B. Ten *largest* functions

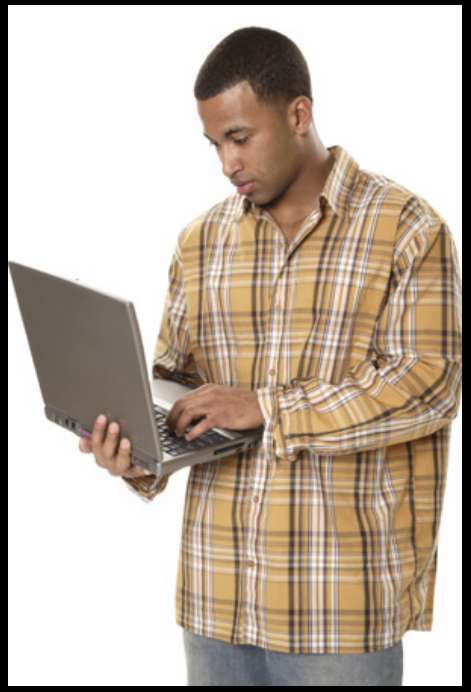
C. Ten *most-fixed* functions

Distance in corporate ladder has a much larger impact





Who produces more buggy code?



A. Junior Developer

B. Senior Developer

Adoption Challenges for Software Mining

Must show value before data quality improves

Correlation vs. Causation

Integration into daily pratice

Domain Knowledge + Close(r) Inspection

- Make sure you manually examine the repositories. Do not fully automate the process!

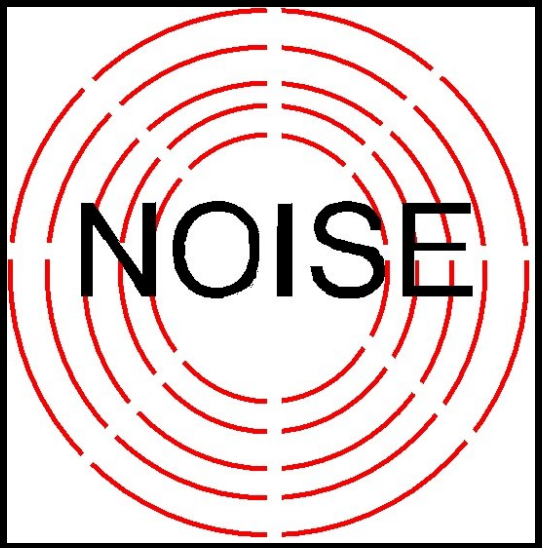
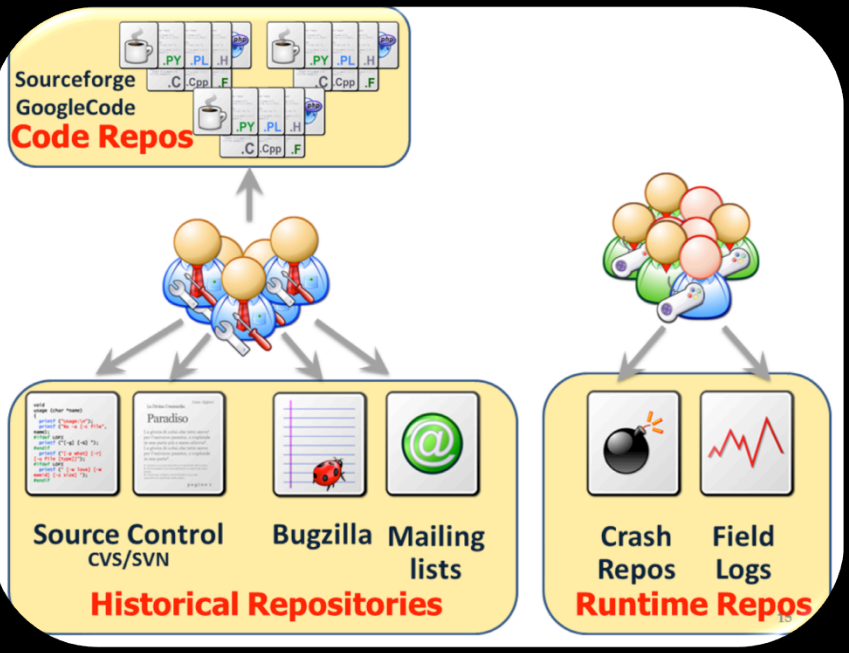


Image from http://www.quincyma.gov/Government/OCS/neighborhoodtips.cfm

The Secret for Software Decision Making



Adoption Challenges for Software Intelligence

Must show value before data quality improves

Correlation vs. Causation

Integration into daily pratice