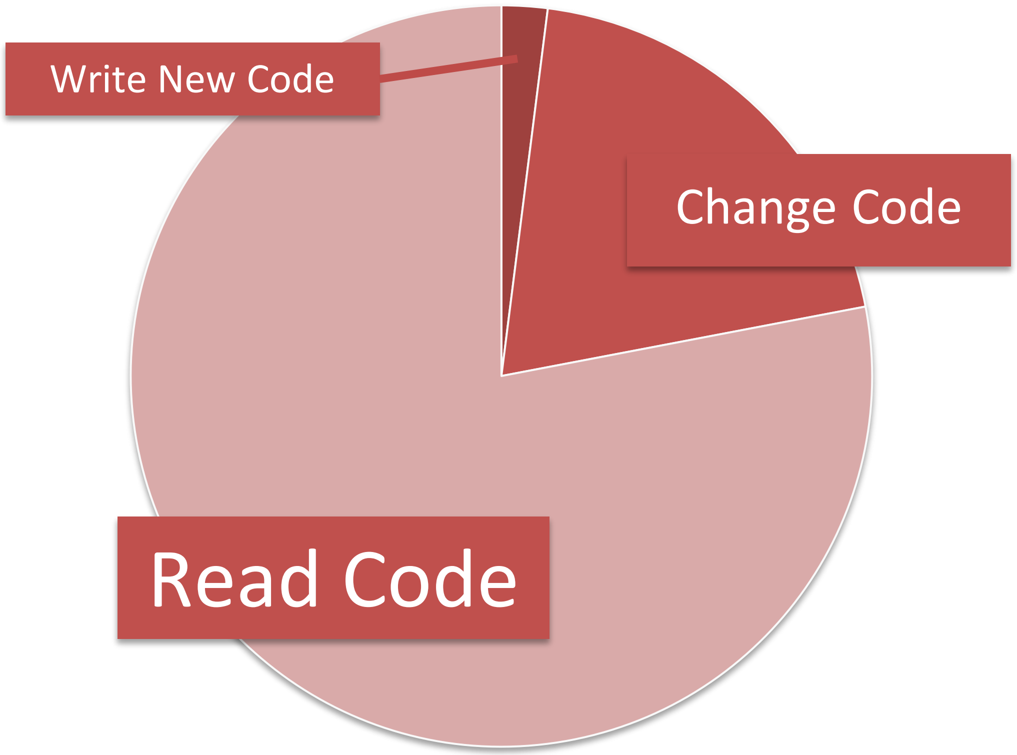
raybuse

**PhD Defense**
4.17.2012

# Automatically Describing Program Structure and Behavior

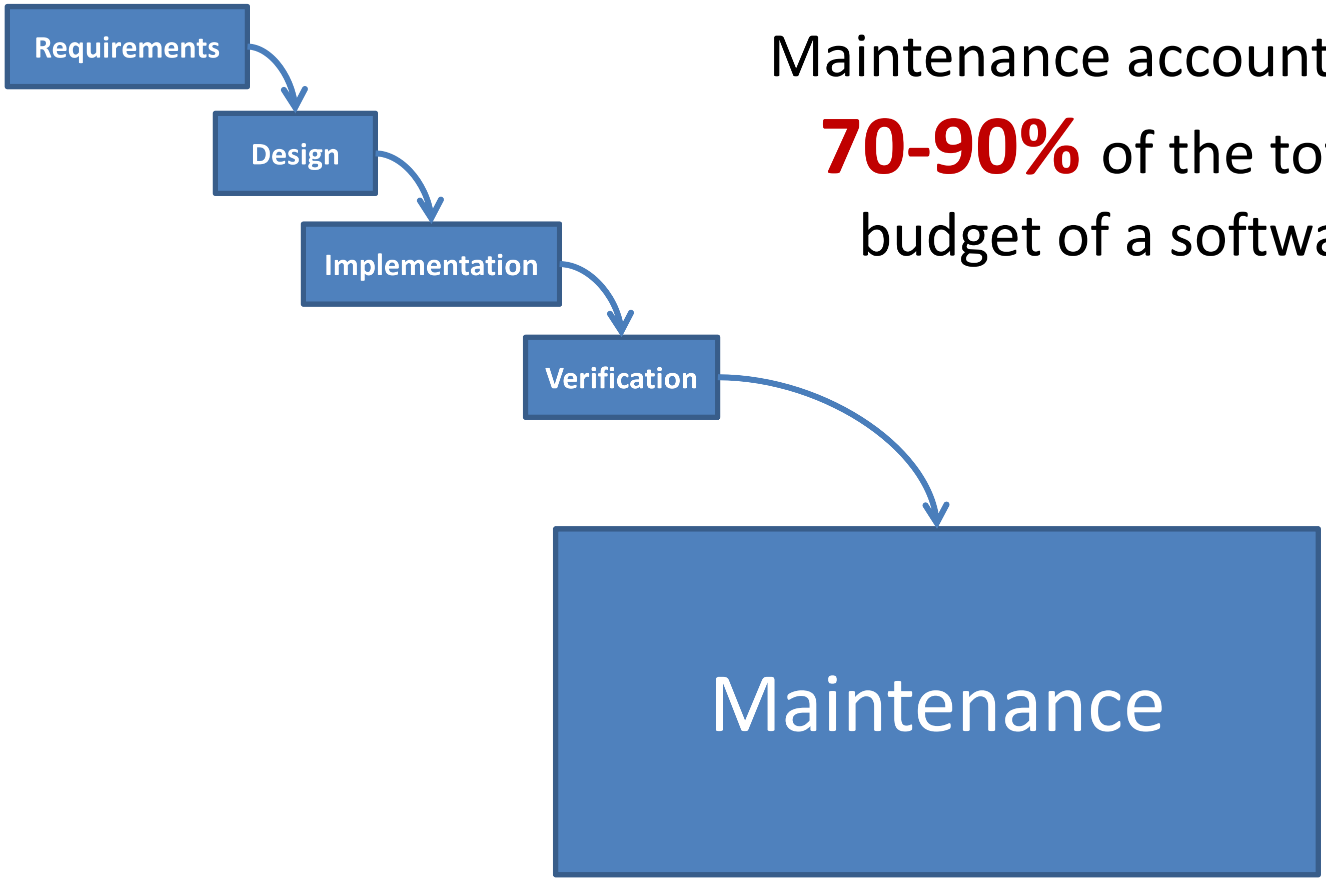
**Readability** **Runtime Behavior** **Documentation**

Code is Difficult to Understand.



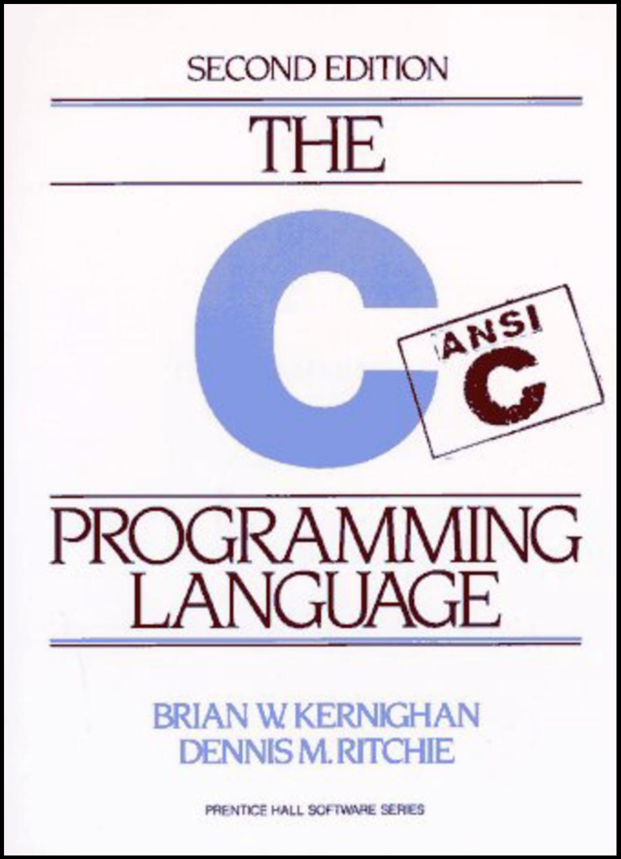
"**Understanding code** is **by far** the activity at which professional developers spend most of their time."

Peter Hallam. *What Do Programmers Really Do Anyway?* Microsoft Developer Network (MSDN) – C# Compiler. Jan 2006



Maintenance accounts for about **70-90%** of the total lifecycle budget of a software project.

T. M. Pigoski. *Practical Software Maintenance: Best Practices for Managing Your Software Investment*.
R. C. Seacord, D. Plakosh, and G. A. Lewis. *Modernizing Legacy Systems: Software Technologies,*



SECOND EDITION

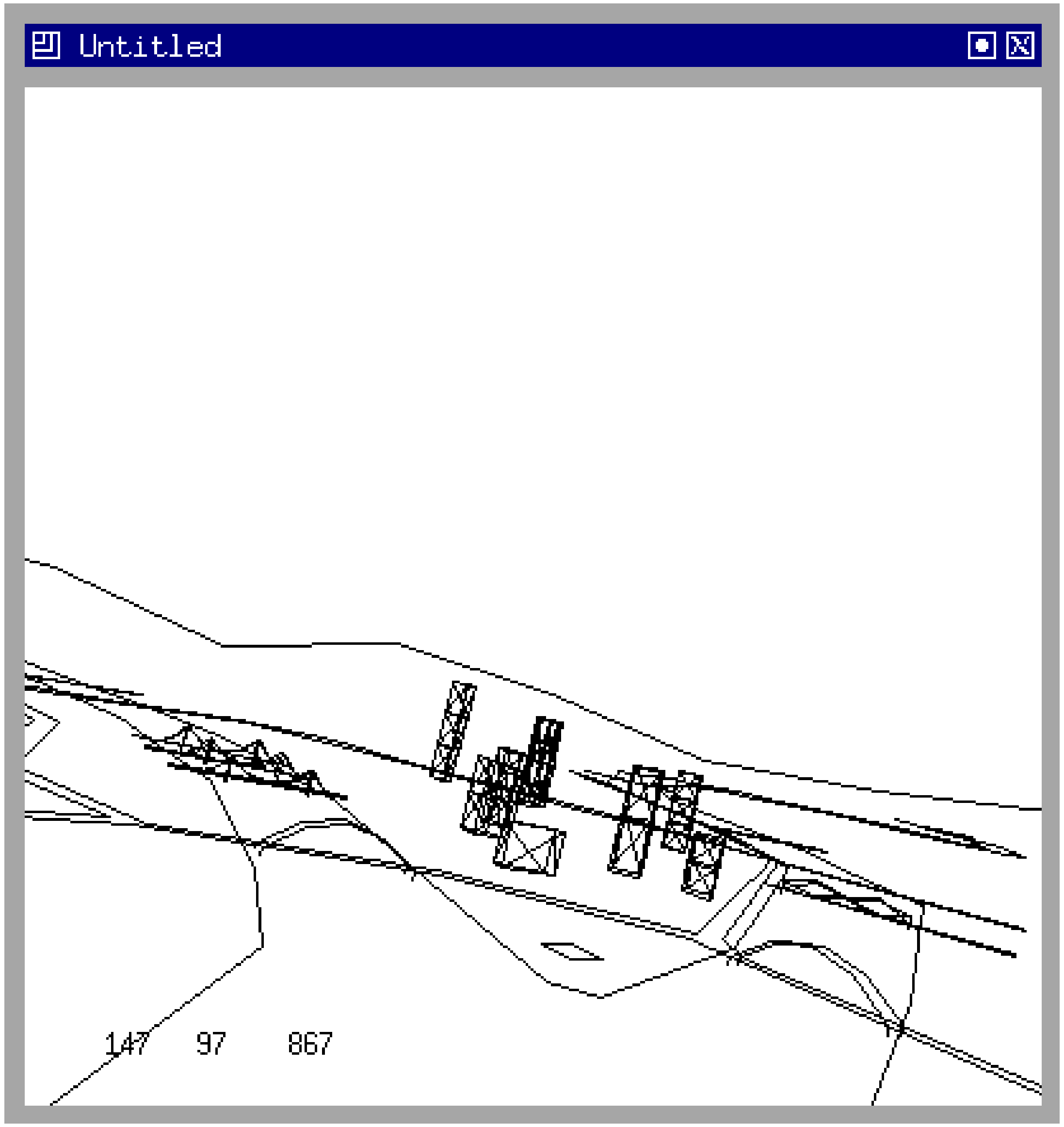
THE C PROGRAMMING LANGUAGE

ANSI C

BRIAN W. KERNIGHAN
DENNIS M. RITCHIE

PRENTICE HALL SOFTWARE SERIES

```
#include                                     <math.h>
#include                                   <sys/time.h>
#include                                   <X11/Xlib.h>
#include                                  <X11/keysym.h>
                                          double L ,o ,P
                                         ,_=dt,T,Z,D=1,d,
                                         s[999],E,h= 8,I,
                                         J,K,w[999],M,m,O
                                        ,n[999],j=33e-3,i=
                                        1E3,r,t, u,v ,W,S=
                                        74.5,l=221,X=7.26,
                                        a,B,A=32.2,c, F,H;
                                        int N,q, C, y,p,U;
                                       Window z; char f[52]
                                    ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                   XEvent z; XNextEvent(e ,&z);
                                       ++*((N=XLookupKeysym
                                         (&z.xkey,0))-IT?
                                         N-LT? UP-N?& E:&
                                         J:& u: &h); --*(
                                         DN -N? N-DT ?N==
                                         RT?&u: & W:&h:&J
                                          ); } m=15*F/l;
                                          c+=(I=M/ l,l*H
                                          +I*M+a*X)*_; H
                                          =A*r+v*X-F*l+(
                                          E=.1+X*4.9/l,t
                                          =T*m/32-I*T/24
                                           )/S; K=F*M+(
                                           h* 1e4/l-(T+
                                           E*5*T*E)/3e2
                                           )/S-X*d-B*A;
                                           a=2.63 /l*d;
                                           X+=( d*l-T/S
                                            *(.19*E +a
                                            *.64+J/1e3
                                            )-M* v +A*
                                            Z)*_; l +=
                                            K *_; W=d;
                                            sprintf(f,
                                            "%5d  %3d"
                                            "%7d",p =l
                                           /1.7,(C=9E3+
                              O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                             X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                             *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                             179*v)/2312; select(p=0,0,0,0,&G); v-=(
                              W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                               )/107e2)*_; D=cos(o); E=sin(o); } }
```



147 97 867

# Java

## What does this print?

```
class Change {
  public static void main(String[] args) {
      System.out.println(2.00 - 1.10);
    }
}
```

*Adapted from Josh Bloch, Jeremy Manson

## What does this print?

```
class Change {
  public static void main(String[] args) {
      System.out.println(2.00 - 1.10);
    }
}
```

Output: 0.8999999999999999

# What does this print?

```java
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

## What does this print?

```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Output: 0.899999999999999911182158092 99874766109466552734375

## BigDecimal

public **BigDecimal**(double val)

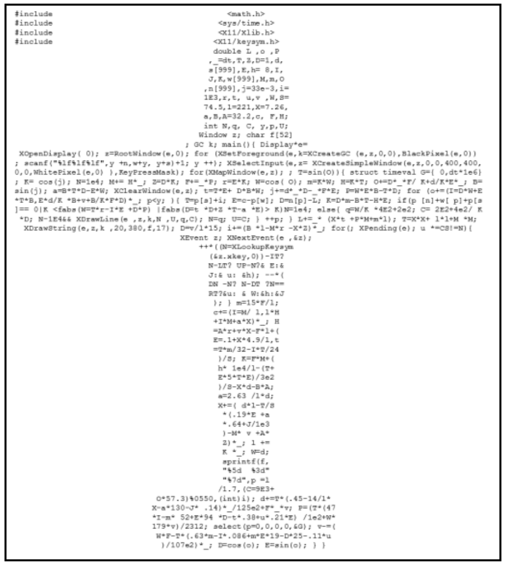
Translates a double into a BigDecimal which is the **exact decimal representation of the double's binary floating-point value**. The scale of the returned BigDecimal is the smallest value such that ($10^{scale} \times$ val) is an integer.

http://docs.oracle.com/javase/6/docs/api/java/math/BigDecimal.html

## What we should have done

```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal("2.00");
    BigDecimal cost = new BigDecimal("1.10");
    System.out.println(payment.subtract(cost));
  }
}
```

Output: 0.90



```
#include                                       <math.h>
#include                                     <sys/time.h>
#include                                     <X11/Xlib.h>
#include                                    <X11/keysym.h>
                                            double L ,o ,P
                                           ,_=dt,T,Z,D=1,d,
                                           s[999],E,h= 8,I,
                                           J,K,w[999],M,m,O
                                          ,n[999],j=33e-3,i=
                                          1E3,r,t, u,v ,W,S=
                                          74.5,l=221,X=7.26,
                                          a,B,A=32.2,c, F,H;
                                          int N,q, C, y,p,U;
                                         Window z; char f[52]
                                       ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                   XEvent z; XNextEvent(e ,&z);
                                       ++*((N=XLookupKeysym
                                         (&z.xkey,0))-IT?
                                         N-LT? UP-N?& E:&
                                         J:& u: &h); --*(
                                         DN -N? N-DT ?N==
                                         RT?&u: & W:&h:&J
                                          ); } m=15*F/l;
                                          c+=(I=M/ l,l*H
                                          +I*M+a*X)*_; H
                                          =A*r+v*X-F*l+(
                                          E=.1+X*4.9/l,t
                                          =T*m/32-I*T/24
                                           )/S; K=F*M+(
                                           h* 1e4/l-(T+
                                           E*5*T*E)/3e2
                                           )/S-X*d-B*A;
                                           a=2.63 /l*d;
                                           X+=( d*l-T/S
                                            *(.19*E +a
                                            *.64+J/1e3
                                            )-M* v +A*
                                            Z)*_; l +=
                                            K *_; W=d;
                                            sprintf(f,
                                            "%5d  %3d"
                                            "%7d",p =l
                                           /1.7,(C=9E3+
                              O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                             X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                             *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                             179*v)/2312; select(p=0,0,0,0,&G); v-=(
                              W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                               )/107e2)*_; D=cos(o); E=sin(o); } }
```

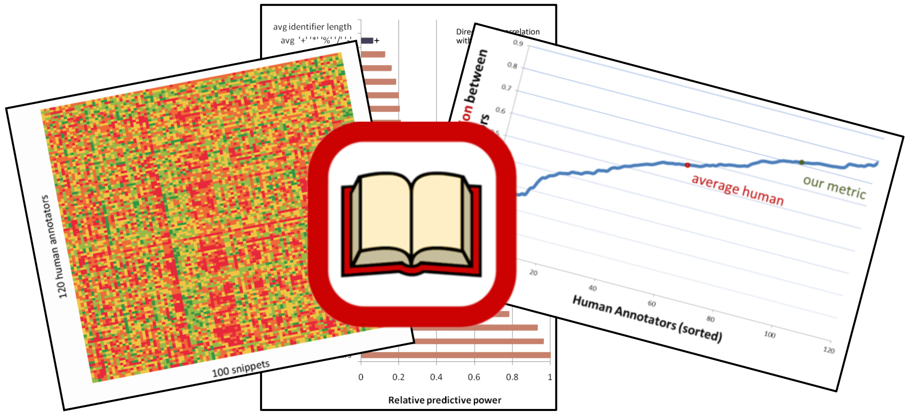
Hard to Read

```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

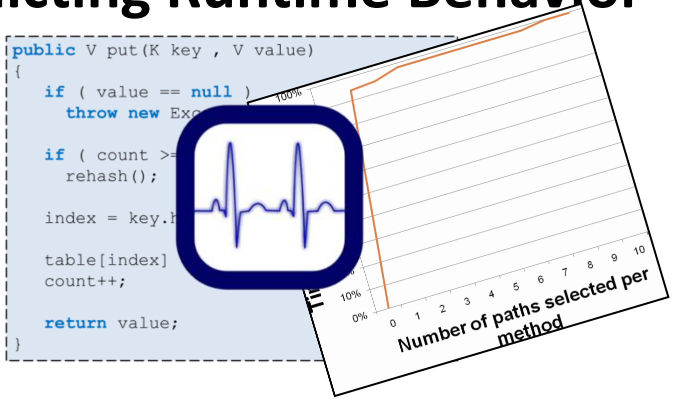
Confusing

# The Rest of this Talk

## Modeling Code Readability

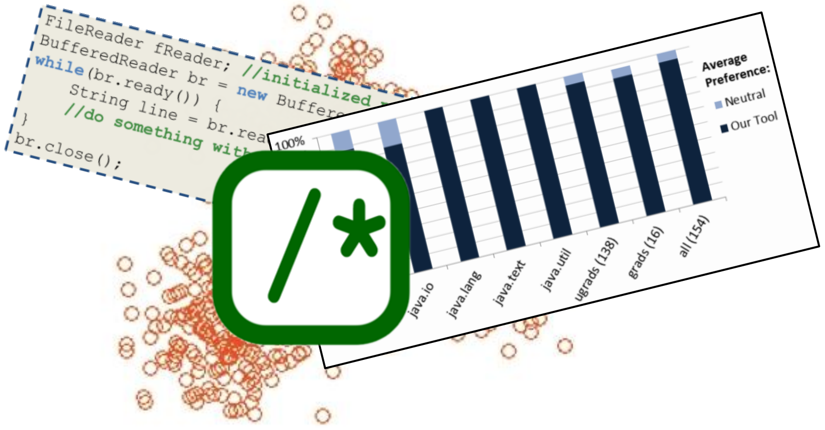


## Predicting Runtime Behavior

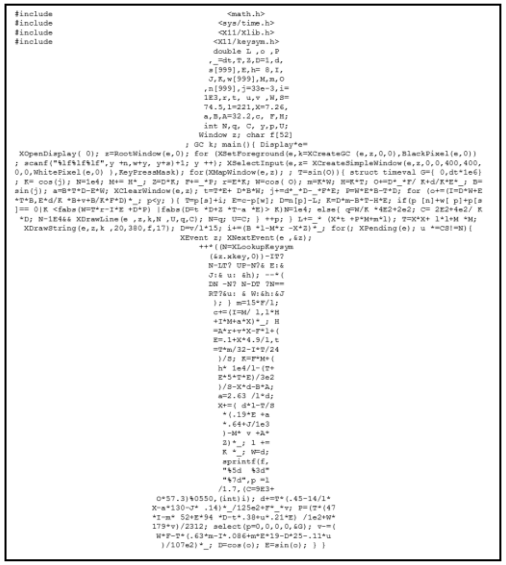


```
public V put(K key , V value)
{
  if ( value == null )
    throw new Exc

  if ( count >=
    rehash();

  index = key.h

  table[index]
  count++;

  return value;
}
```

## Synthesizing Documentation



```
FileReader fReader; //initialized
BufferedReader br = new Buffere
while(br.ready()) {
  String line = br.rea
  //do something wit
}
br.close();
```



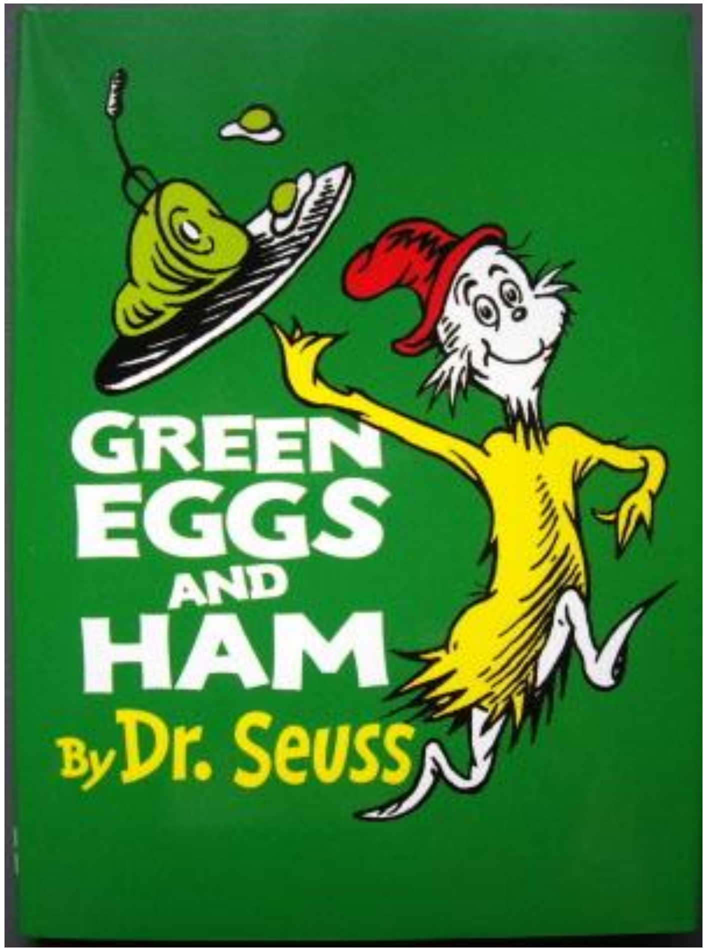
```c
#include                                  <math.h>
#include                                <sys/time.h>
#include                                <X11/Xlib.h>
#include                               <X11/keysym.h>
                                       double L ,o ,P
                                      ,_=dt,T,Z,D=1,d,
                                      s[999],E,h= 8,I,
                                      J,K,w[999],M,m,O
                                     ,n[999],j=33e-3,i=
                                     1E3,r,t, u,v ,W,S=
                                     74.5,l=221,X=7.26,
                                     a,B,A=32.2,c, F,H;
                                     int N,q, C, y,p,U;
                                    Window z; char f[52]
                                  ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                 XEvent z; XNextEvent(e ,&z);
                                    ++*((N=XLookupKeysym
                                      (&z.xkey,0))-IT?
                                      N-LT? UP-N?& E:&
                                      J:& u: &h); --*(
                                      DN -N? N-DT ?N==
                                      RT?&u: & W:&h:&J
                                       ); } m=15*F/l;
                                       c+=(I=M/ l,l*H
                                       +I*M+a*X)*_; H
                                       =A*r+v*X-F*l+(
                                       E=.1+X*4.9/l,t
                                       =T*m/32-I*T/24
                                        )/S; K=F*M+(
                                        h* 1e4/l-(T+
                                        E*5*T*E)/3e2
                                        )/S-X*d-B*A;
                                        a=2.63 /l*d;
                                        X+=( d*l-T/S
                                         *(.19*E +a
                                         *.64+J/1e3
                                         )-M* v +A*
                                         Z)*_; l +=
                                         K *_; W=d;
                                         sprintf(f,
                                         "%5d  %3d"
                                         "%7d",p =l
                                        /1.7,(C=9E3+
                            O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                           X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                           *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                           179*v)/2312; select(p=0,0,0,0,&G); v-=(
                            W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                             )/107e2)*_; D=cos(o); E=sin(o); } }
```

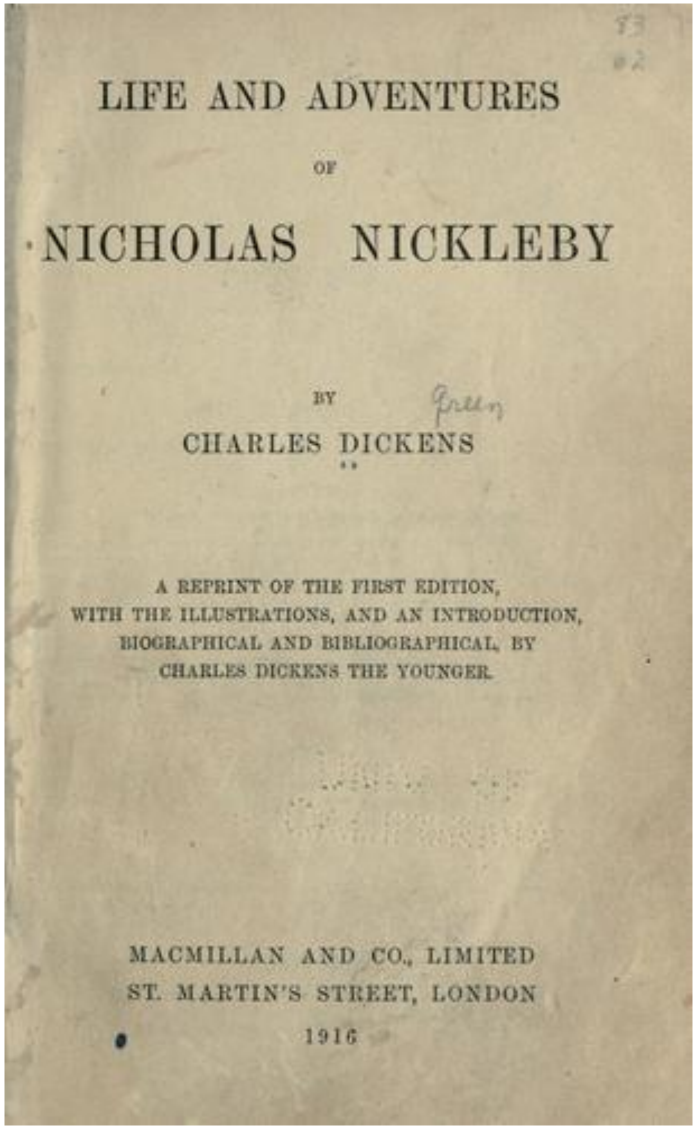
Hard to Read

```java
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Confusing

## How can we tell if code is readable?




53
#2

LIFE AND ADVENTURES

OF

NICHOLAS NICKLEBY

BY

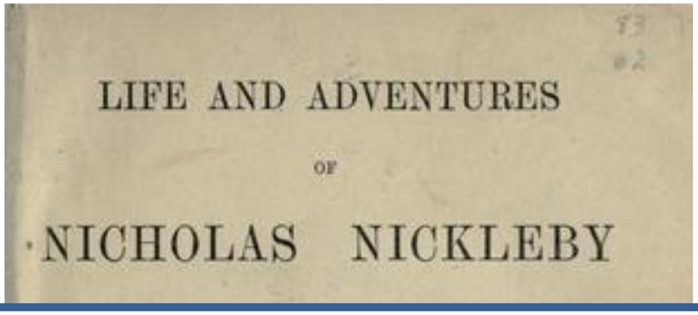
Green

CHARLES DICKENS

A REPRINT OF THE FIRST EDITION,
WITH THE ILLUSTRATIONS, AND AN INTRODUCTION,
BIOGRAPHICAL AND BIBLIOGRAPHICAL, BY
CHARLES DICKENS THE YOUNGER.

MACMILLAN AND CO., LIMITED
ST. MARTIN'S STREET, LONDON
1916

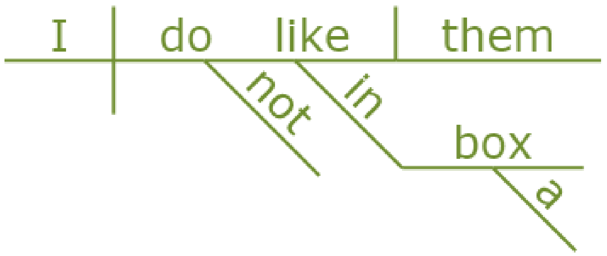
“…
I do not like them in a box.
I do not like with a fox.
I do not like them in a house.
I do not like them with a mouse.
…”

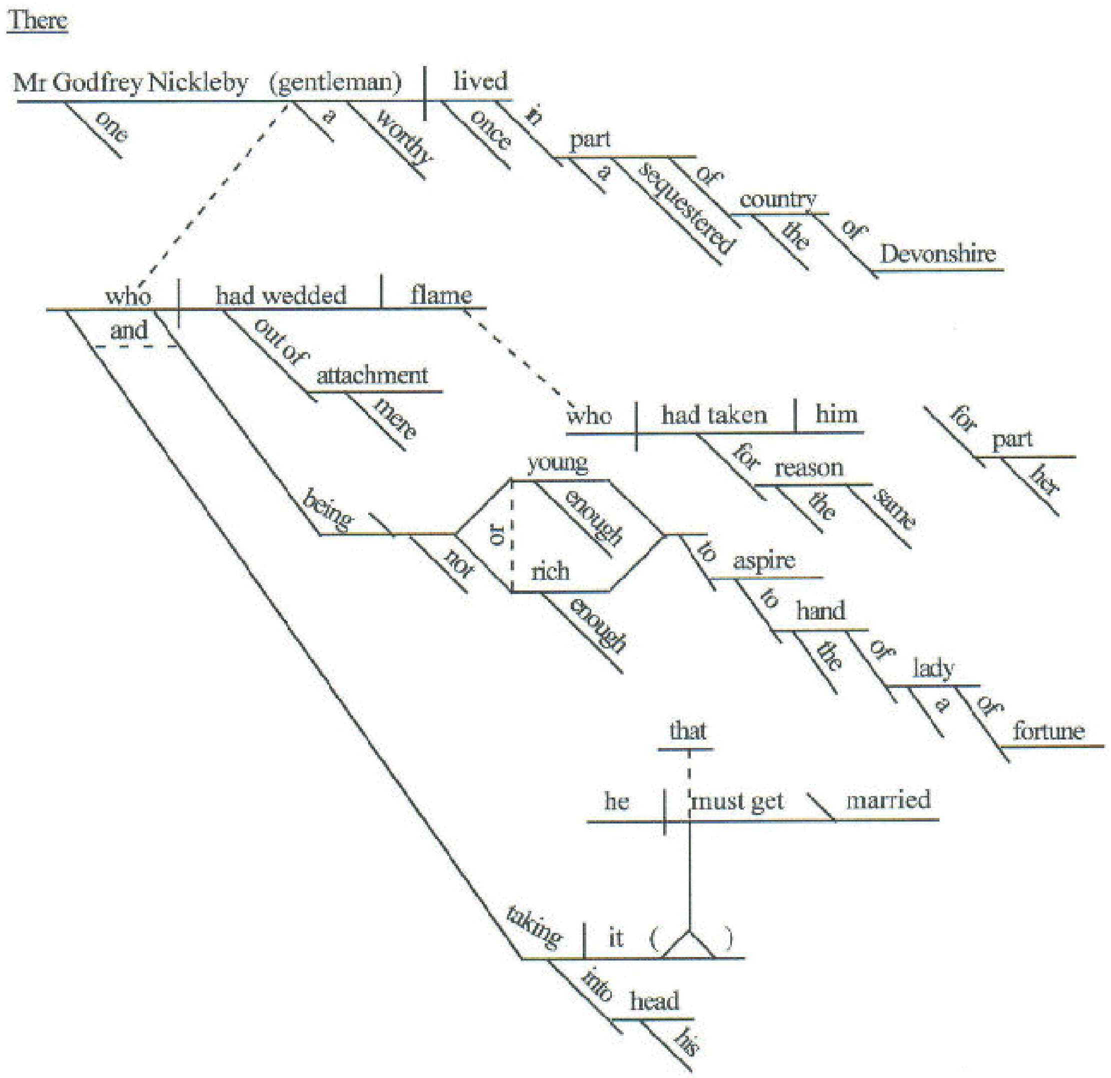

LIFE AND ADVENTURES

OF

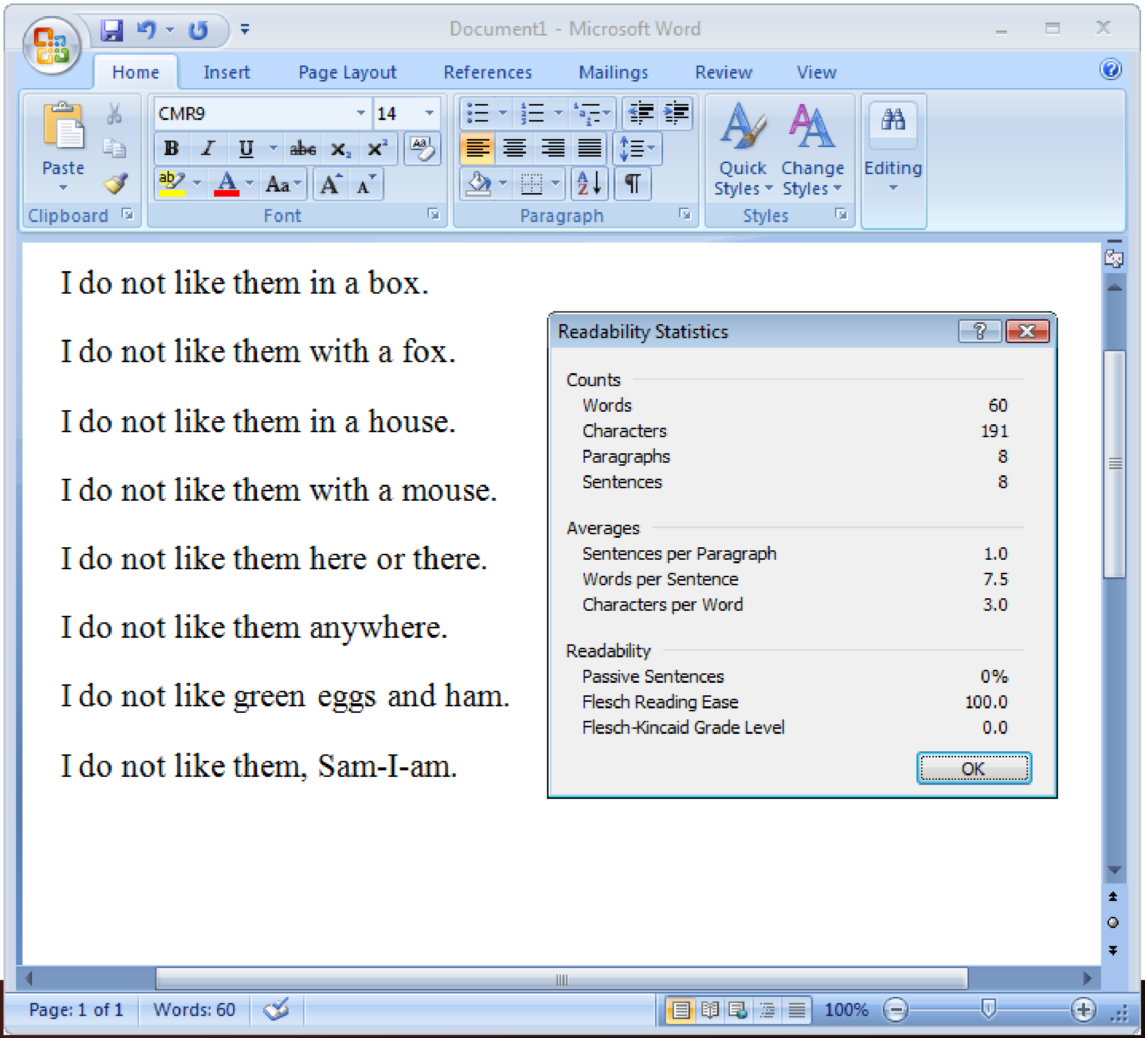
NICHOLAS NICKLEBY

"There once lived, in a sequestered part of the country of Devonshire, one Mr. Godfrey Nickleby: a worthy gentleman, who, taking it into his head rather late in life that he must get married, and not being young enough or rich enough to aspire to the hand of a lady of fortune, had wedded an old flame out of mere attachment, who in her turn had taken him for the same reason."


I
do
like
them
not
in
box
a


There
Mr Godfrey Nickleby
(gentleman)
lived
one
a
worthy
once
in
part
a
sequestered
of
country
the
of
Devonshire
who
had wedded
flame
and
out of
attachment
mere
who
had taken
him
for
reason
the
same
for
part
her
being
not
young
enough
or
rich
enough
to
aspire
to
hand
the
of
lady
a
of
fortune
that
he
must get
married
taking
it
into
head
his

# Flesch-Kincaid Readability



I do not like them in a box.

I do not like them with a fox.

I do not like them in a house.

I do not like them with a mouse.

I do not like them here or there.

I do not like them anywhere.

I do not like green eggs and ham.

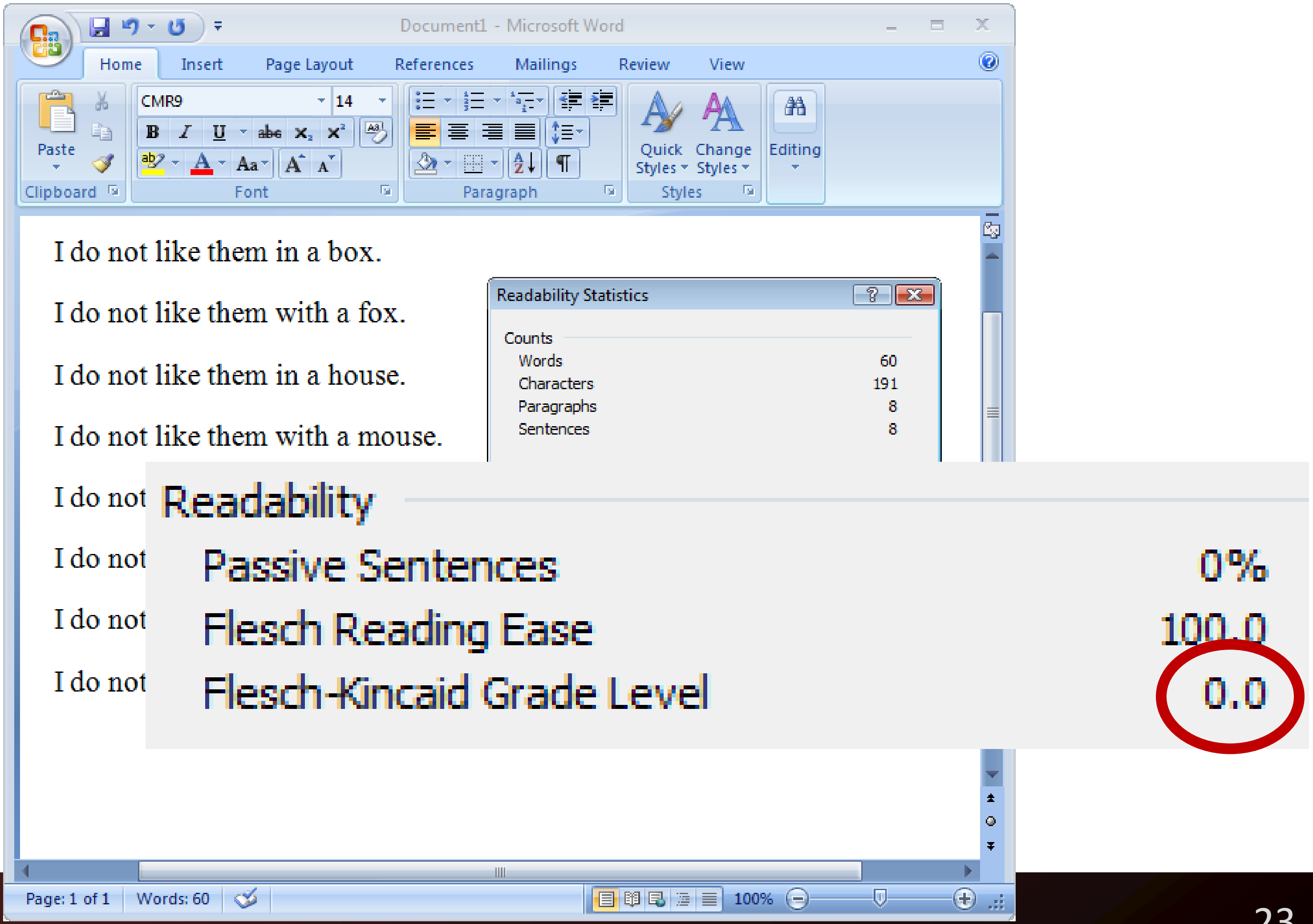
I do not like them, Sam-I-am.

Readability Statistics

| | |
|---|---|
| **Counts** | |
| Words | 60 |
| Characters | 191 |
| Paragraphs | 8 |
| Sentences | 8 |
| **Averages** | |
| Sentences per Paragraph | 1.0 |
| Words per Sentence | 7.5 |
| Characters per Word | 3.0 |
| **Readability** | |
| Passive Sentences | 0% |
| Flesch Reading Ease | 100.0 |
| Flesch-Kincaid Grade Level | 0.0 |

OK

# Flesch-Kincaid Readability



I do not like them in a box.

I do not like them with a fox.

I do not like them in a house.

I do not like them with a mouse.

I do not

I do not

I do not

I do not

Readability Statistics

Counts

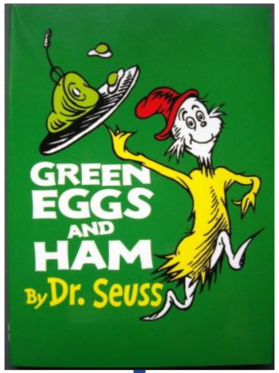
| | |
|---|---|
| Words | 60 |
| Characters | 191 |
| Paragraphs | 8 |
| Sentences | 8 |

Readability

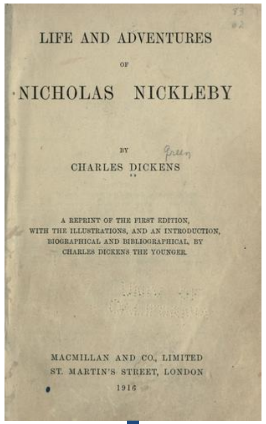
| | |
|---|---|
| Passive Sentences | 0% |
| Flesch Reading Ease | 100.0 |
| Flesch-Kincaid Grade Level | 0.0 |

# Flesch-Kincaid Readability

0.0



15.3


LIFE AND ADVENTURES

OF

NICHOLAS NICKLEBY

BY

Green

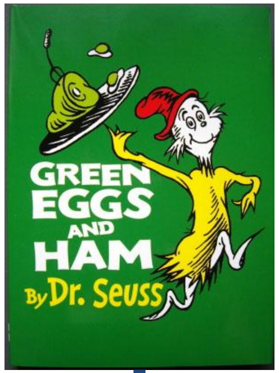
CHARLES DICKENS

A REPRINT OF THE FIRST EDITION,
WITH THE ILLUSTRATIONS, AND AN INTRODUCTION,
BIOGRAPHICAL AND BIBLIOGRAPHICAL, BY
CHARLES DICKENS THE YOUNGER.

MACMILLAN AND CO., LIMITED
ST. MARTIN'S STREET, LONDON
1916

# Flesch-Kincaid Readability

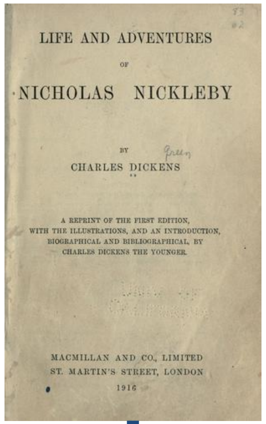
0.0



10.0

DOD MIL-M-38784B

15.3



LIFE AND ADVENTURES

OF

NICHOLAS NICKLEBY

BY

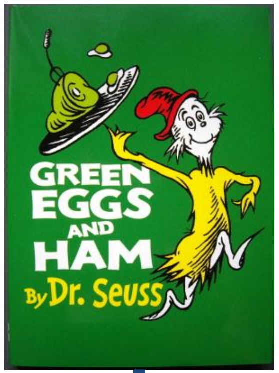
CHARLES DICKENS

A REPRINT OF THE FIRST EDITION, WITH THE ILLUSTRATIONS, AND AN INTRODUCTION, BIOGRAPHICAL AND BIBLIOGRAPHICAL, BY CHARLES DICKENS THE YOUNGER.

MACMILLAN AND CO., LIMITED
ST. MARTIN'S STREET, LONDON
1916

# Flesch-Kincaid Readability

0.0



10.0

DOD MIL-M-38784B

12.1

**Automatically Describing Program Structure and Behavior**

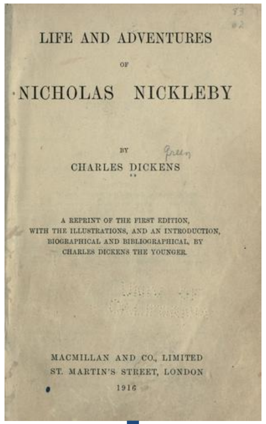
A Dissertation
Presented to
the Faculty of the School of Engineering and Applied Science
University of Virginia

In Partial Fulfillment
of the requirements for the Degree
Doctor of Philosophy (Computer Science)

by

Raymond P. L. Buse
May 2012

15.3



LIFE AND ADVENTURES
OF
NICHOLAS NICKLEBY
BY
CHARLES DICKENS

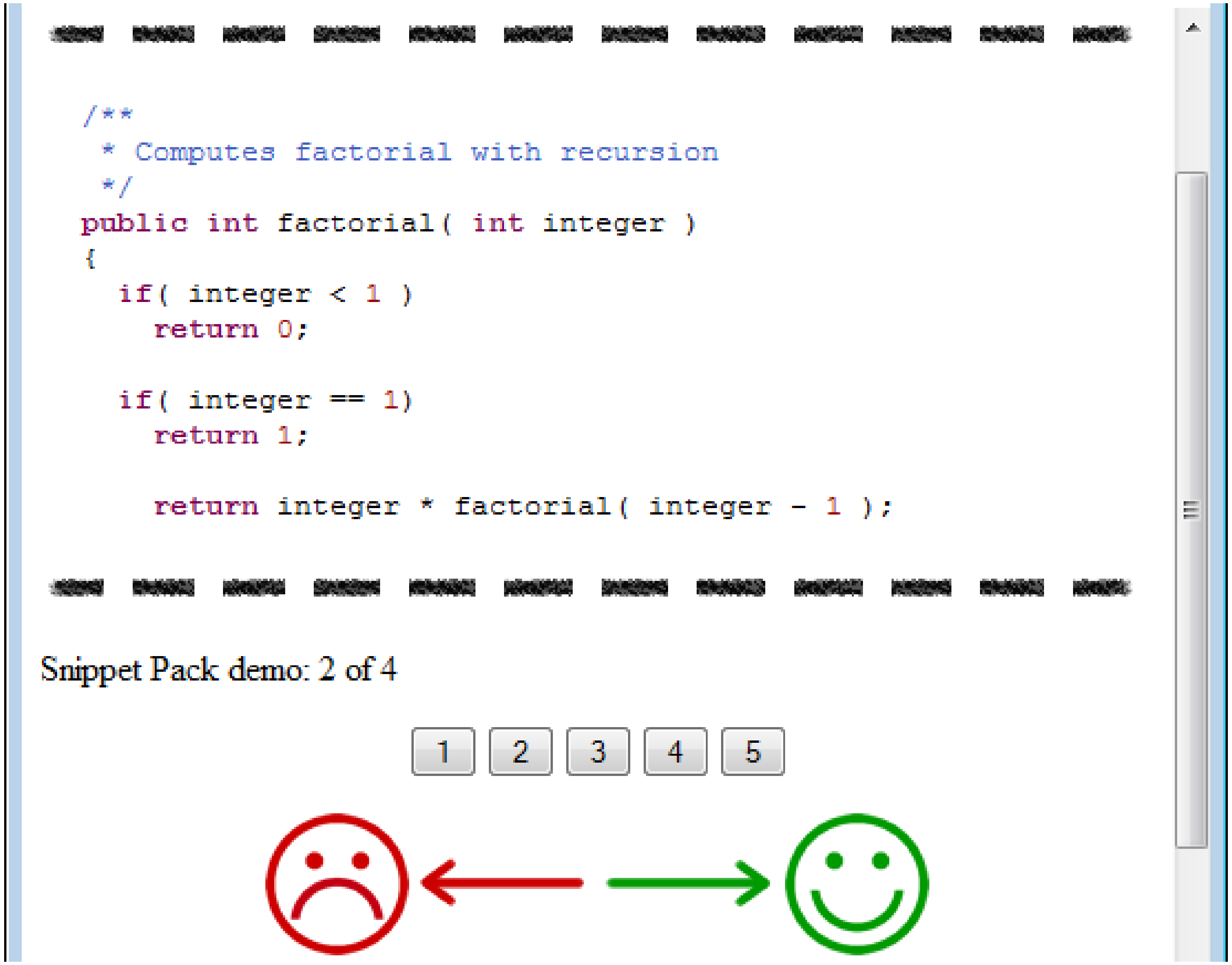
A REPRINT OF THE FIRST EDITION,
WITH THE ILLUSTRATIONS, AND AN INTRODUCTION,
BIOGRAPHICAL AND BIBLIOGRAPHICAL, BY
CHARLES DICKENS THE YOUNGER.

MACMILLAN AND CO., LIMITED
ST. MARTIN'S STREET, LONDON
1916

## Can this work for code?

Research questions:

- To what extent do humans agree on what code is readable?
- Can we derive a accurate descriptive model for readability?
- Does the model correlate significantly with software quality?



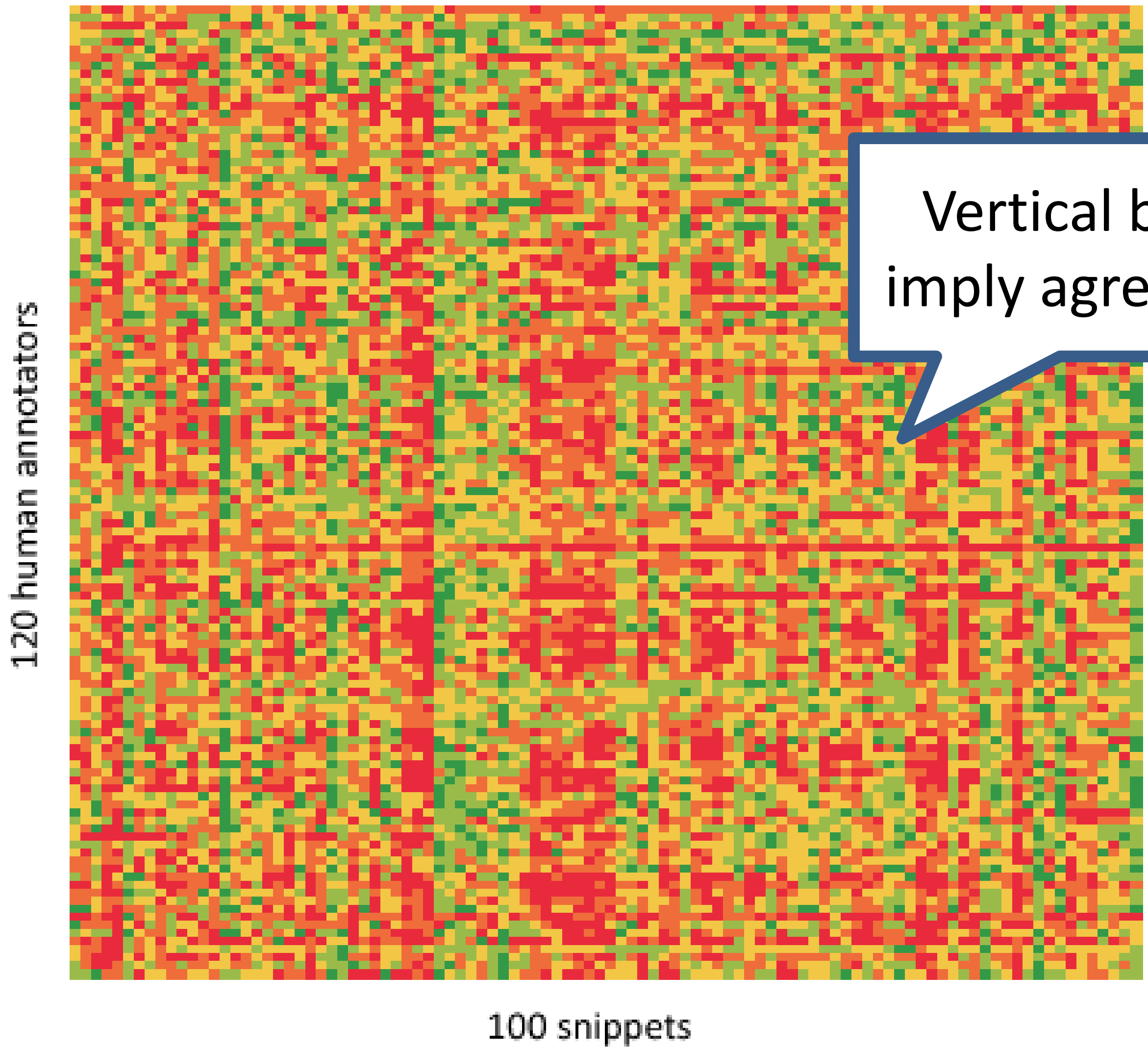
```
/**
 * Computes factorial with recursion
 */
public int factorial( int integer )
{
  if( integer < 1 )
    return 0;

  if( integer == 1)
    return 1;

    return integer * factorial( integer - 1 );
```

Snippet Pack demo: 2 of 4

More Readable

🟩
🟩
🟨
🟧
🟥

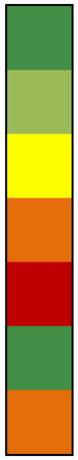
Less Readable

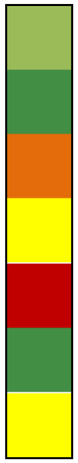


Vertical bands imply agreement

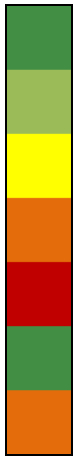
NOT SURE IF HIGH AGREEMENT
OR JUST RED-GREEN COLOR BLIND

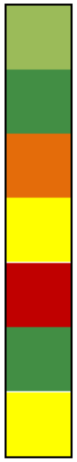
# Quantifying Agreement





# Quantifying Agreement

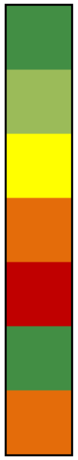


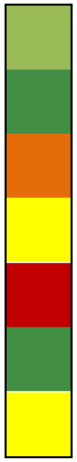


## Correlation Statistics

- Pearson's *r* – linear dependence
- Spearman's *ρ* – monotonic dependence
- Kendall's *τ* – counts bubble sort operations
- Cohen's *κ* – nominal agreement

## Quantifying Agreement



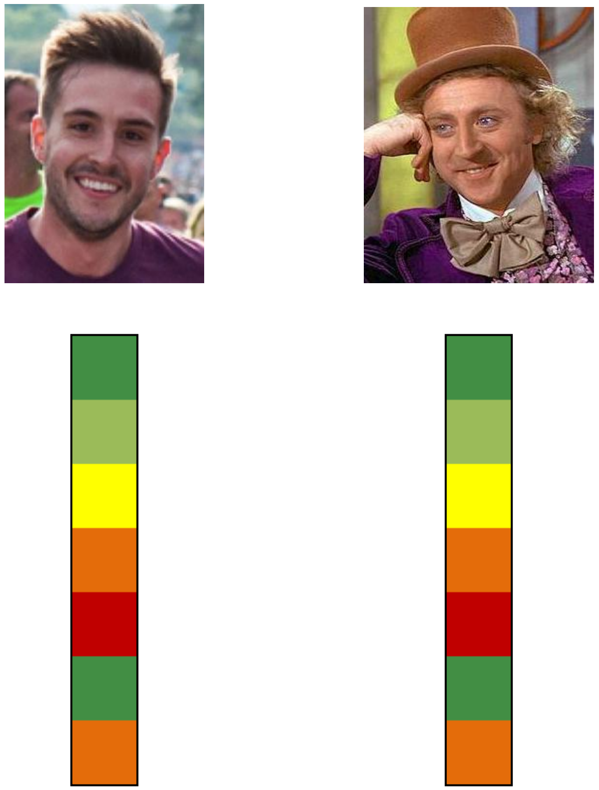


### Correlation Statistics

- Pearson's $r$ – linear dependence
- Spearman's $\rho$ – monotonic dependence
- Kendall's $\tau$ – co bubble sort operations
- Cohen's $\kappa$ –

Absolute agreement less important than relative agreement

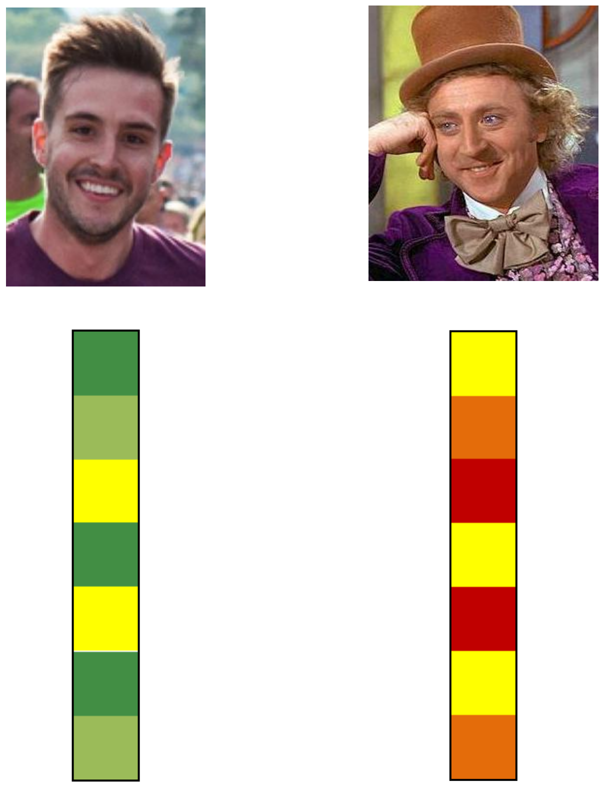
# Quantifying Agreement



Perfect **Absolute** Agreement

$$\rho = 1$$

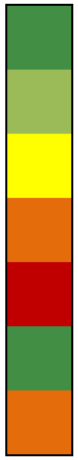
# Quantifying Agreement

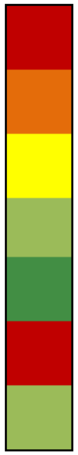


Perfect **Relative** Agreement

$$\rho = 1$$

# Quantifying Agreement

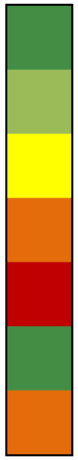


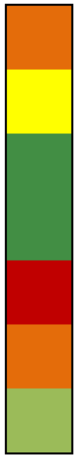


Absolute Disagreement

$\rho = -1$

# Quantifying Agreement

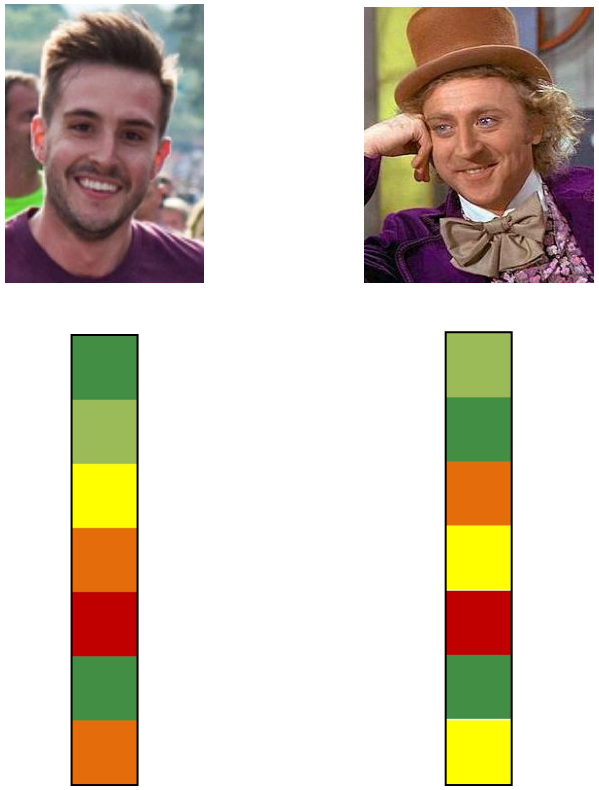




Random Agreement

$$\rho = 0$$

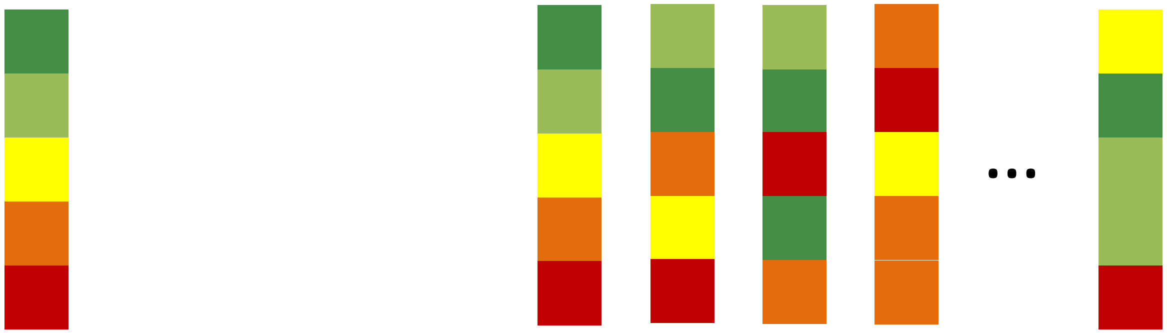
# Quantifying Agreement



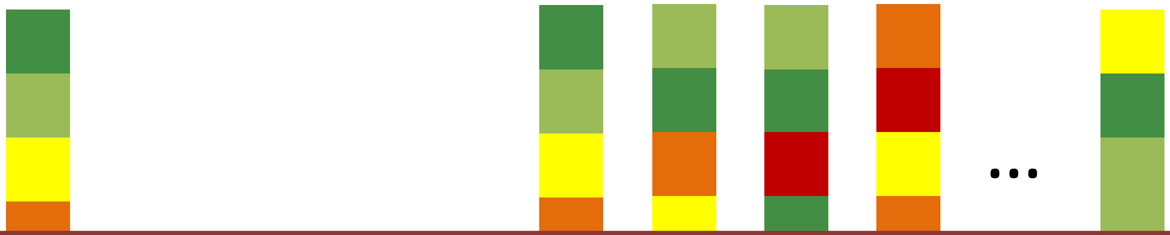
"Strong" Agreement

$\rho > 0.5$

# Quantifying Agreement With a Group

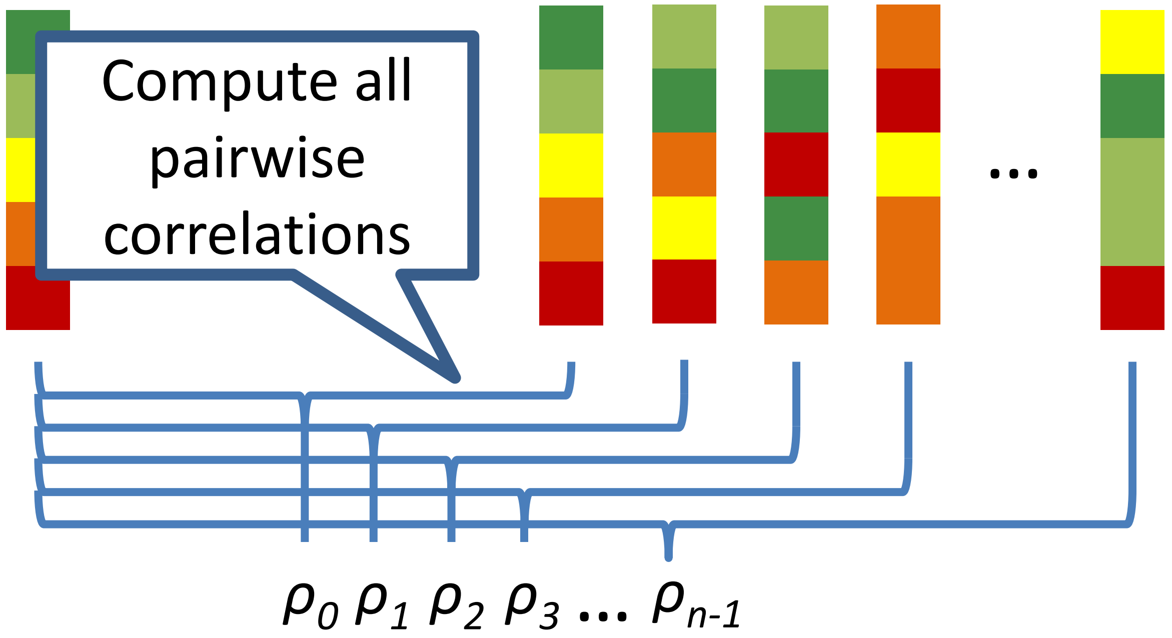


# Quantifying Agreement With a Group

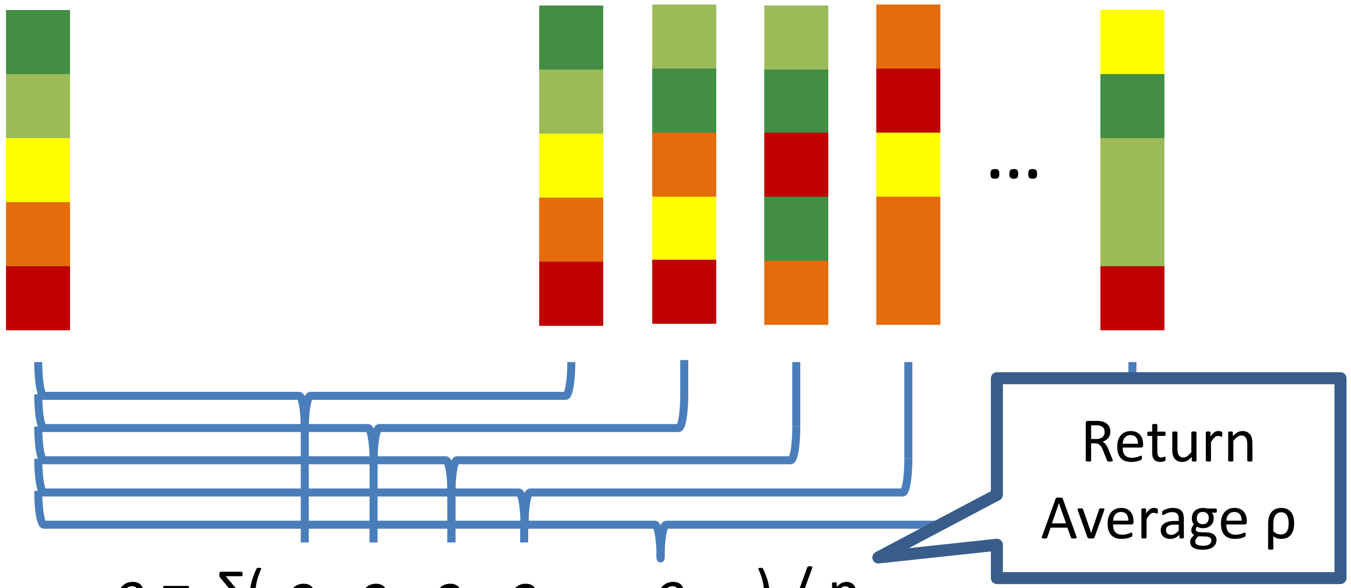


**Apples and Oranges: An Empirical Comparison of Commonly Used Indices of Interrater Agreement**
Allan P. Jones, Lee A. Johnson, Mark C. Butler and Deborah S. Main
*The Academy of Management Journal* , Vol. 26, No. 3 (Sep., 1983), pp. 507-519

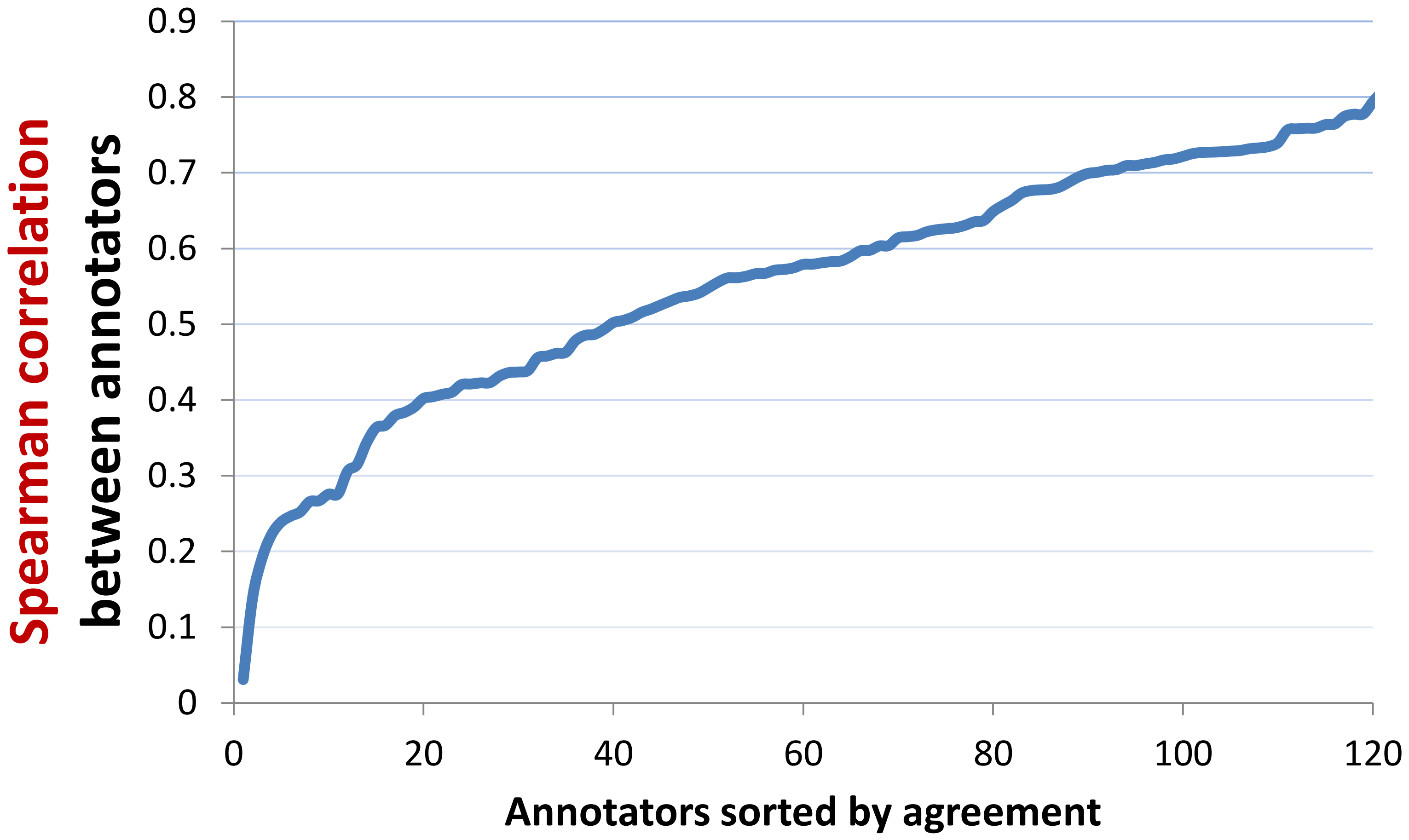
## Quantifying Agreement With a Group


Compute all pairwise correlations
…
$\rho_0$ $\rho_1$ $\rho_2$ $\rho_3$ … $\rho_{n-1}$

## Quantifying Agreement With a Group



$\rho = \Sigma(\ \rho_0\ \rho_1\ \rho_2\ \rho_3 \ldots \rho_{n-1}\ ) / n$


Spearman correlation
between annotators
Annotators sorted by agreement
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
20
40
60
80
100
120

## Building a Model

## Flesch-Kincaid Readability

$$0.39\left(\frac{wordCount}{sentenceCount}\right)+11.8\left(\frac{syllableCount}{wordCount}\right)-15.59$$

## Flesch-Kincaid Readability

$$0.39\left(\frac{wordCount}{sentenceCount}\right)+11.8\left(\frac{syllableCount}{wordCount}\right)-15.59$$

$$\mathrm{f}(\vec{x})=\beta_0+\beta_1(x_1)\ +\beta_2(x_2)+\cdots+\beta_n(x_n)$$

# Flesch-Kincaid Readability

$$0.39\left(\boxed{\frac{wordCount}{sentenceCount}}\right) + 11.8\left(\boxed{\frac{syllableCount}{wordCount}}\right) - 15.59$$

$$\mathrm{f}(\vec{x}) = \beta_0 + \beta_1(\boxed{x_1}) + \beta_2(\boxed{x_2}) + \cdots + \beta_n(\boxed{x_n})$$

**Features**

# Flesch-Kincaid Readability

$$\boxed{0.39}\left(\boxed{\frac{wordCount}{sentenceCount}}\right) + \boxed{11.8}\left(\boxed{\frac{syllableCount}{wordCount}}\right) - \boxed{15.59}$$

$$f(\vec{x}) = \boxed{\beta_0} + \boxed{\beta_1}(\boxed{x_1}) + \boxed{\beta_2}(\boxed{x_2}) + \cdots + \boxed{\beta_n}(\boxed{x_n})$$

**Features** **Weights**

**Features**

Creativity / Intuition

**Weights**

Supervised Learning

- Regression
- Bayesian
- Neural Net
- SVM
- …

Use training data from human study

## Potential Code Readability Features

```
class Change {

  //Computes 2.00 – 1.10
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(“2.00”);
    BigDecimal cost = new BigDecimal(“1.10”);
    System.out.println(payment.subtract(cost));
  }
}
```

## Potential Code Readability Features

```
class Change {

  //Computes 2.00 – 1.10
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(“2.00”);
    BigDecimal cost = new BigDecimal(“1.10”);
    System.out.println(payment.subtract(cost));
  }
}
```

Line Length

## Potential Code Readability Features

Comments

```
class Change {

  //Computes 2.00 – 1.10
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal("2.00");
    BigDecimal cost = new BigDecimal("1.10");
    System.out.println(payment.subtract(cost));
  }
}
```

## Potential Code Readability Features

Identifier Length

```
class Change {

  //Computes 2.00 – 1.10
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(“2.00”);
    BigDecimal cost = new BigDecimal(“1.10”);
    System.out.println(payment.subtract(cost));
  }
}
```

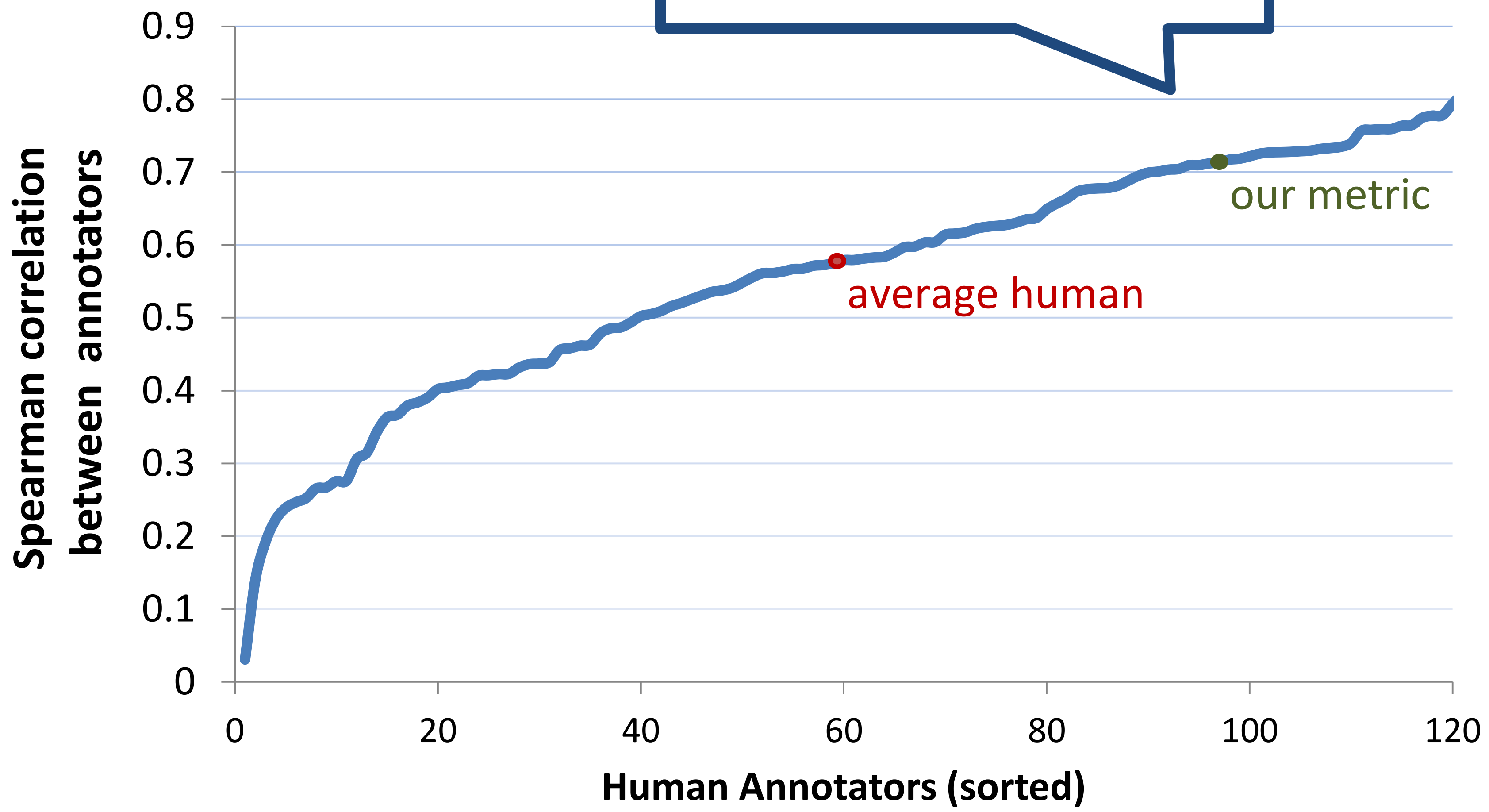
## Potential Code Readability Features

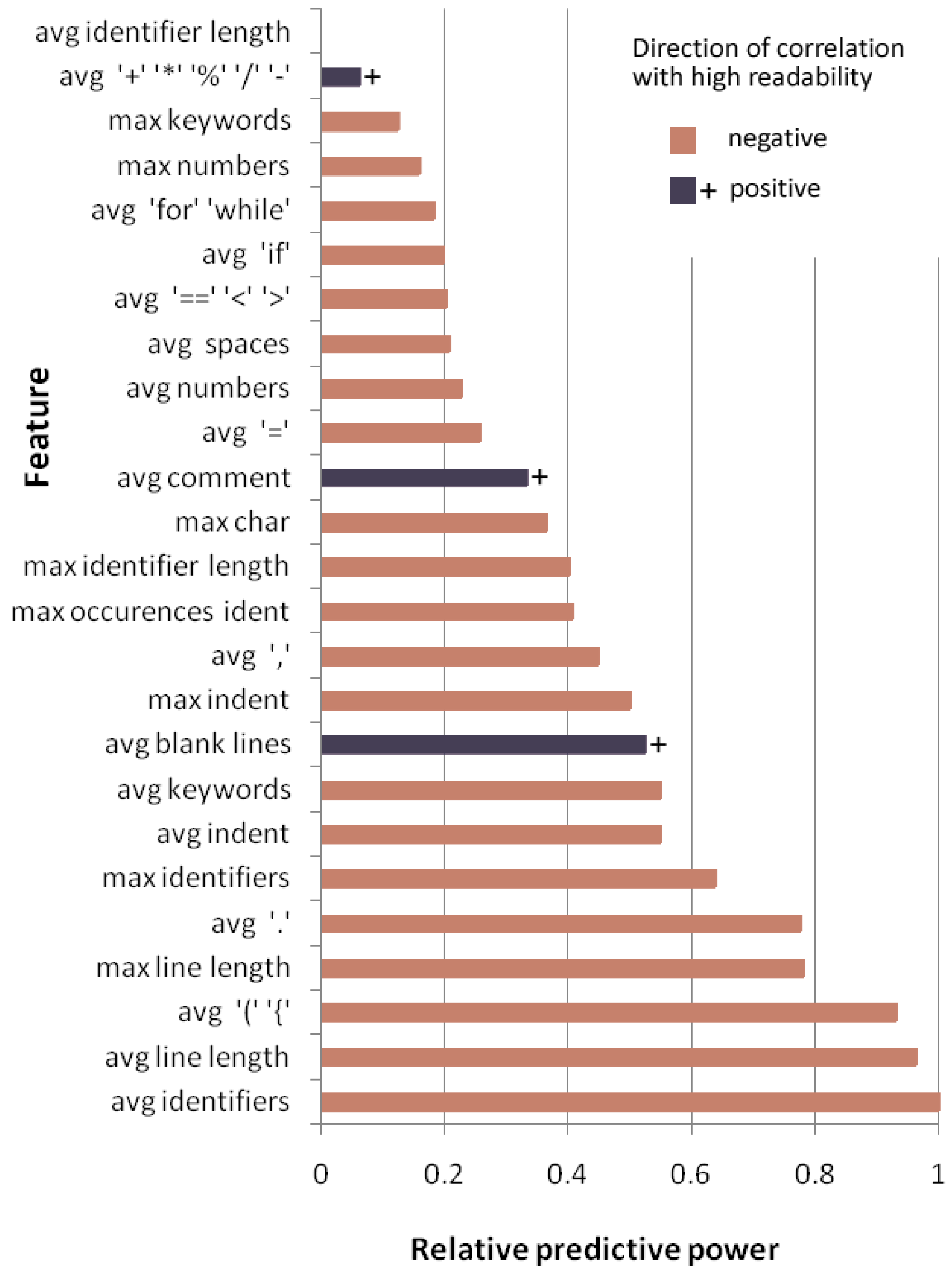
Blank Lines

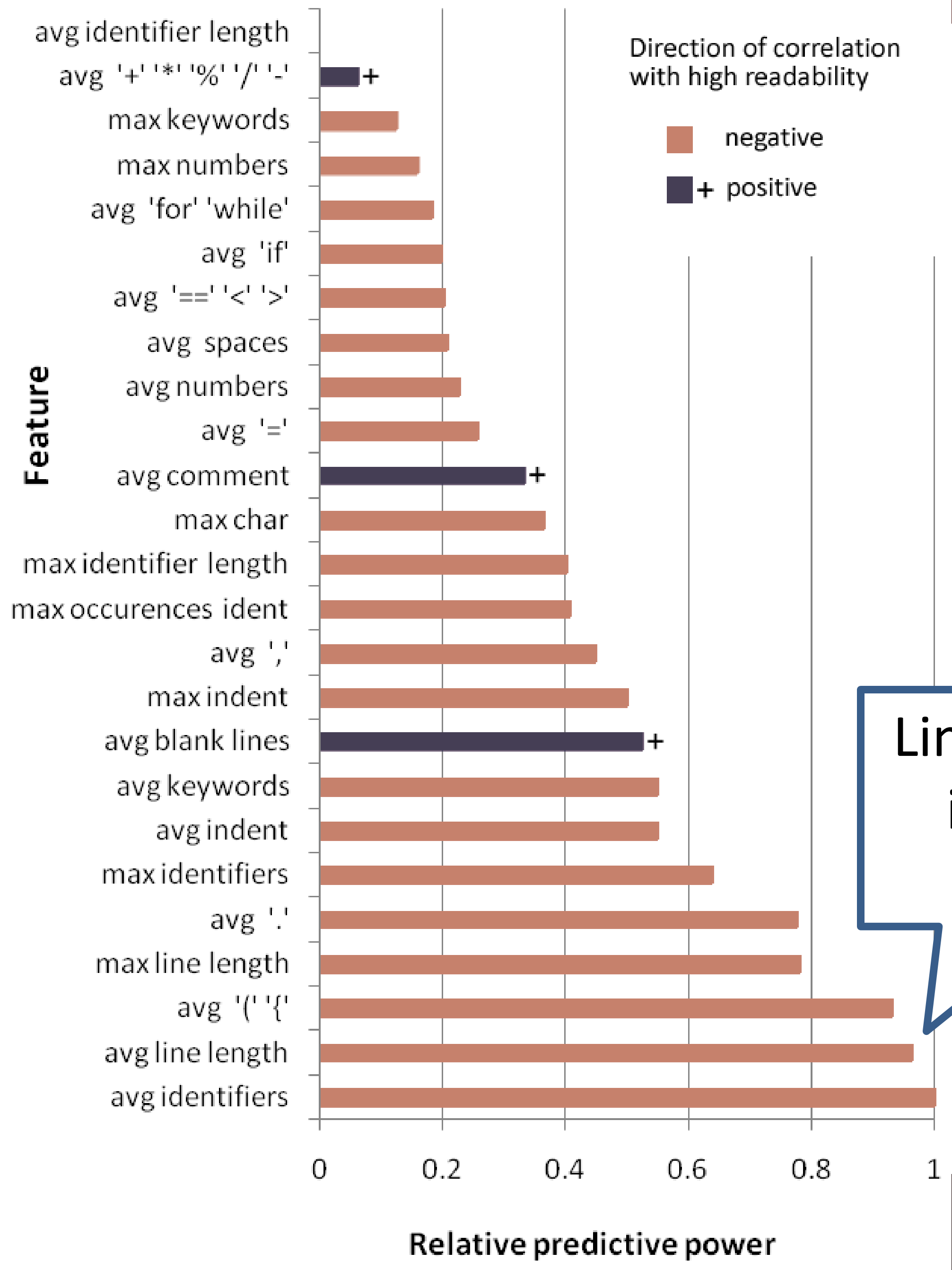
```
class Change {

  //Computes 2.00 – 1.10
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(“2.00”);
    BigDecimal cost = new BigDecimal(“1.10”);
    System.out.println(payment.subtract(cost));
  }
}
```

## Evaluation

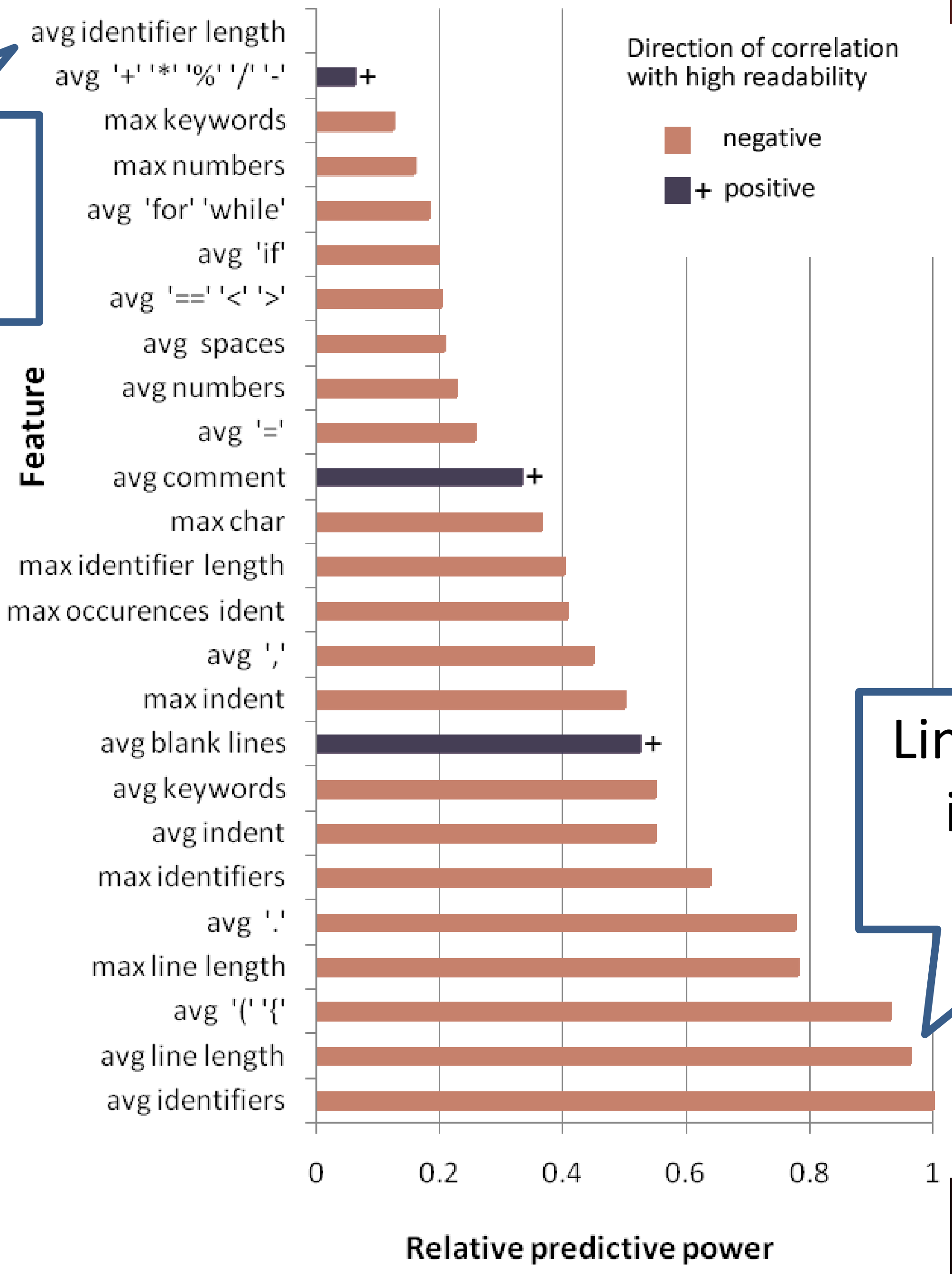
Model agrees with humans as much as they agree with each other




Direction of correlation with high readability
negative
+ positive
Feature
avg identifier length
avg '+' '*' '%' '/' '-'
max keywords
max numbers
avg 'for' 'while'
avg 'if'
avg '==' '<' '>'
avg spaces
avg numbers
avg '='
avg comment
max char
max identifier length
max occurences ident
avg ','
max indent
avg blank lines
avg keywords
avg indent
max identifiers
avg '.'
max line length
avg '(' '{'
avg line length
avg identifiers
+
+
+
0
0.2
0.4
0.6
0.8
1
Relative predictive power



Line Length, # of identifiers is important



Length of identifiers is not

Line Length, # of identifiers is important

## Readability and Software Quality

# Benchmarks

12 Open Source Sourceforge Projects, Over 2M LOC

jE

JasperReports

GANTT project

HIBERNATE

JUnit

soapUI

FreeCol - the Colonization of America

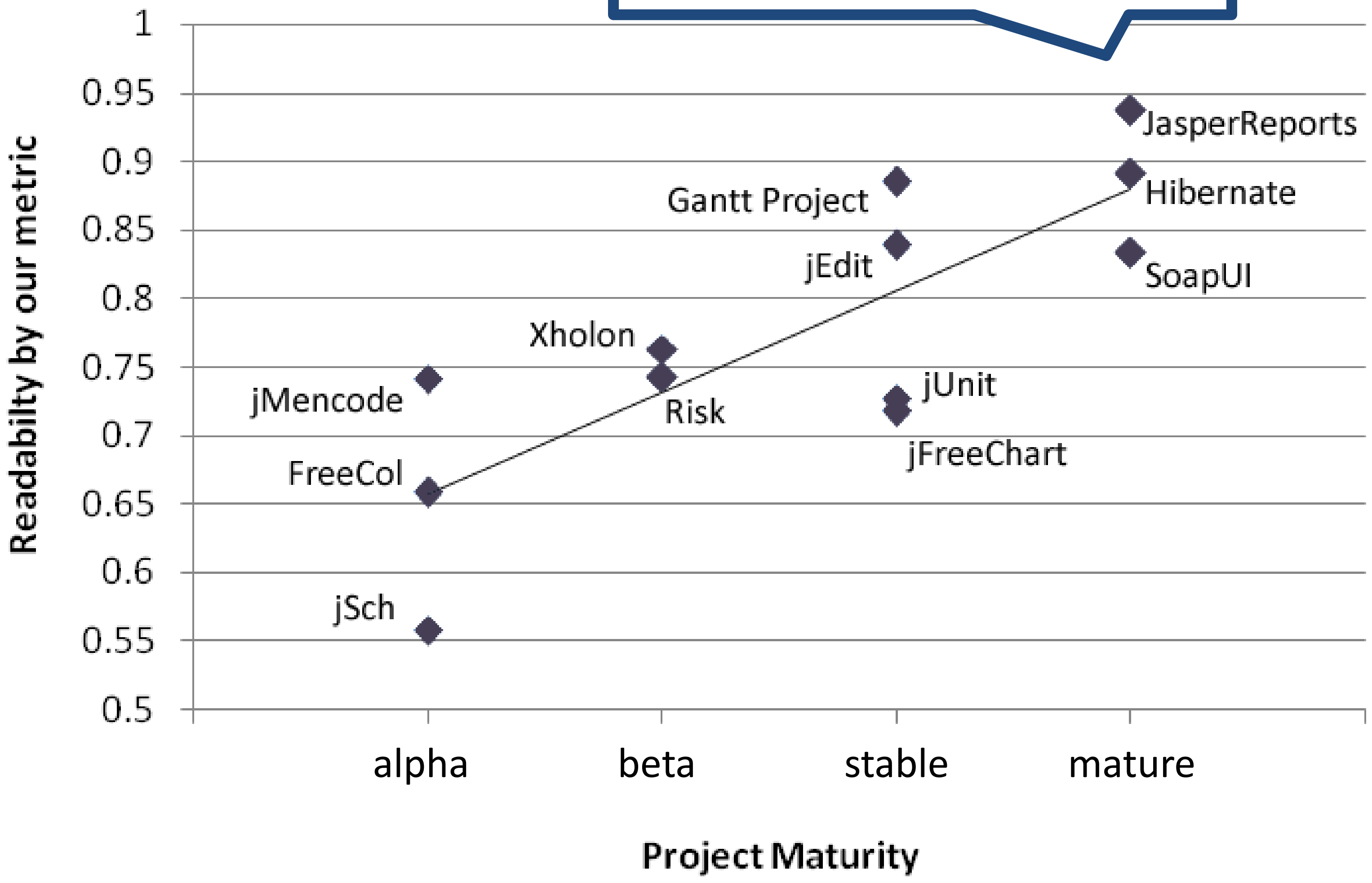
Risk for Java

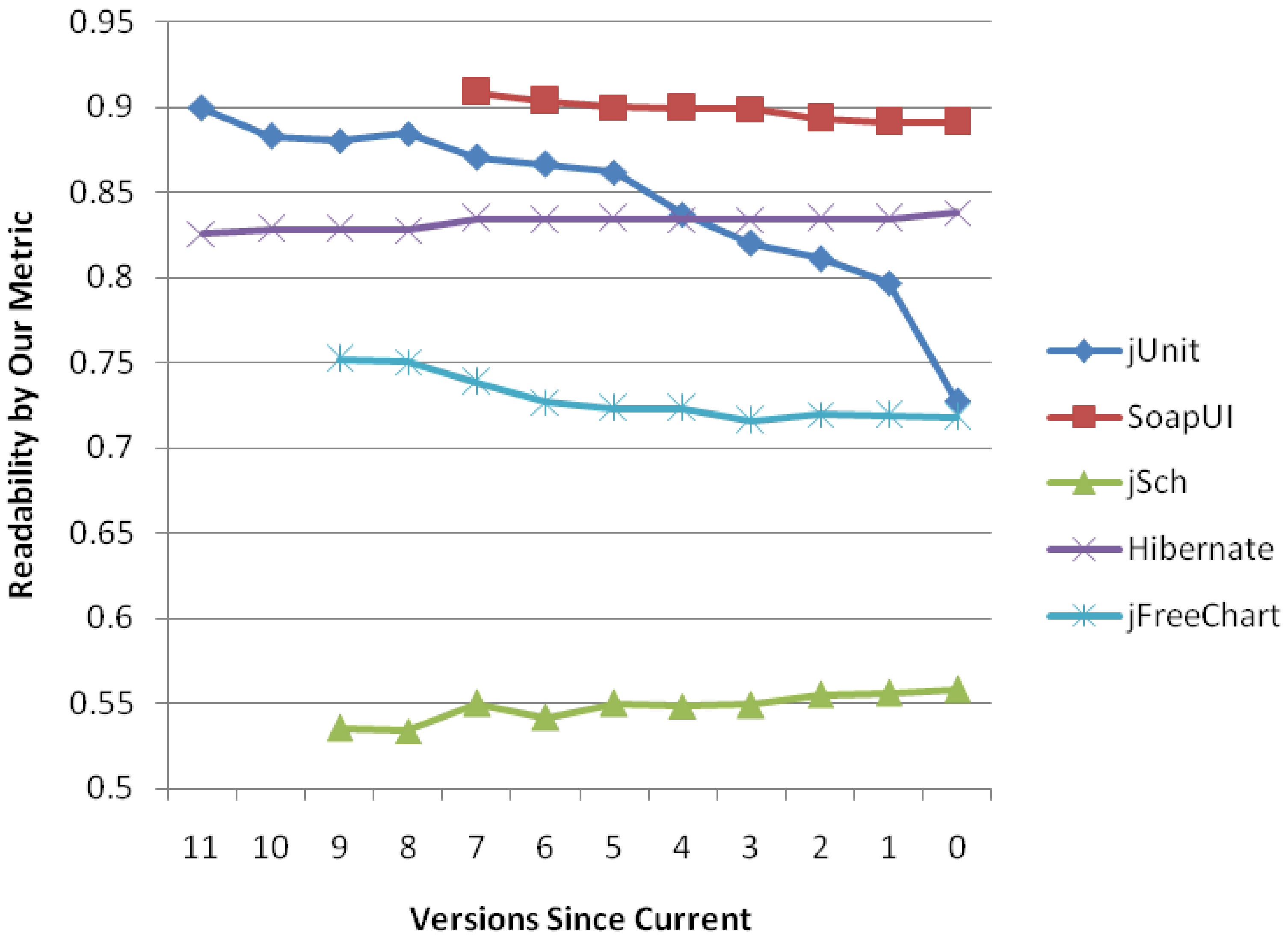
JFreeChart

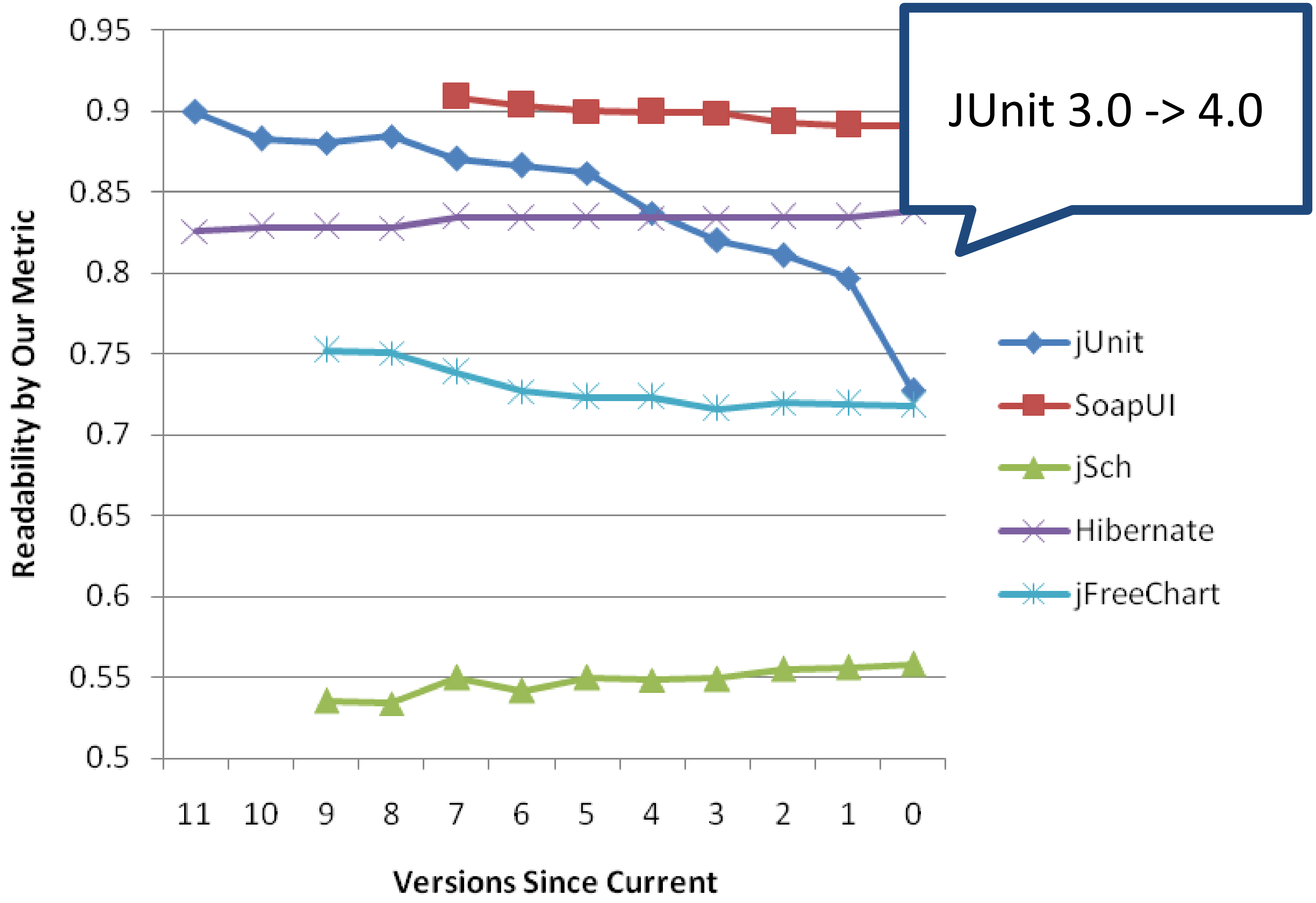
J

jMencode

Mature projects tend to be more readable

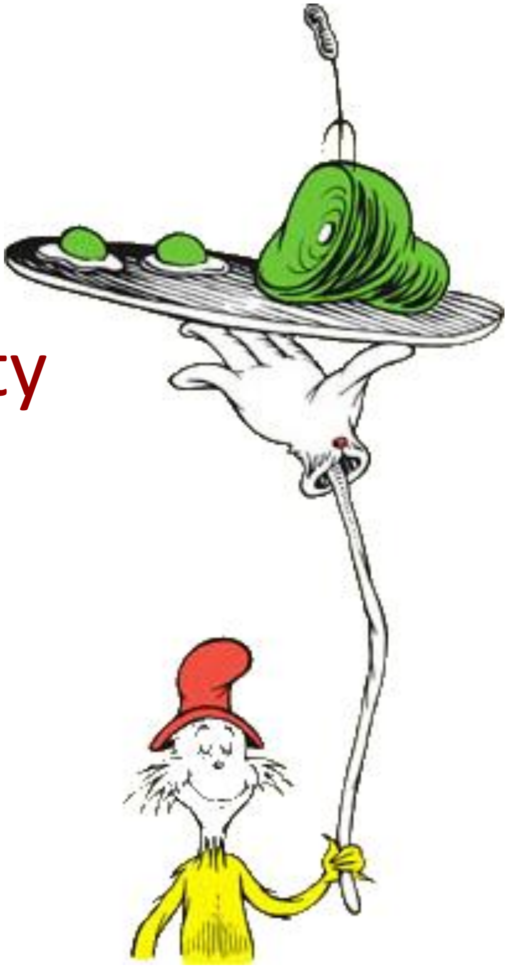


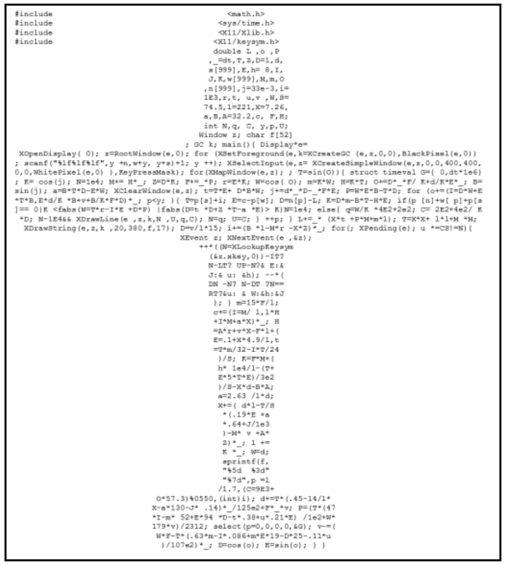

0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
Readability by Our Metric
11
10
9
8
7
6
5
4
3
2
1
0
Versions Since Current
jUnit
SoapUI
jSch
Hibernate
jFreeChart


JUnit 3.0 -> 4.0
Readability by Our Metric
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
11
10
9
8
7
6
5
4
3
2
1
0
Versions Since Current
jUnit
SoapUI
jSch
Hibernate
jFreeChart

## Readability Metric

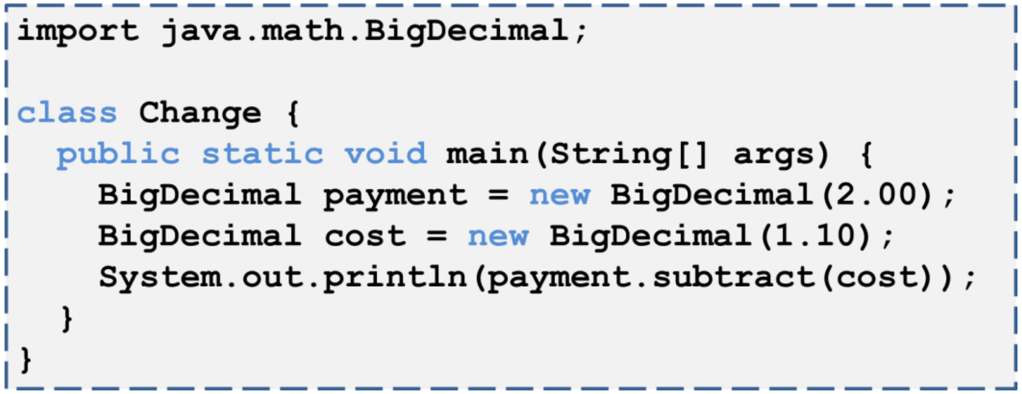
- Metric (and source code) is freely available: arrestedcomputing.com/readability
- Has been used directly in several published papers.





```
#include                                        <math.h>
#include                                      <sys/time.h>
#include                                      <X11/Xlib.h>
#include                                     <X11/keysym.h>
                                             double L ,o ,P
                                            ,_=dt,T,Z,D=1,d,
                                            s[999],E,h= 8,I,
                                            J,K,w[999],M,m,O
                                           ,n[999],j=33e-3,i=
                                           1E3,r,t, u,v ,W,S=
                                           74.5,l=221,X=7.26,
                                           a,B,A=32.2,c, F,H;
                                           int N,q, C, y,p,U;
                                          Window z; char f[52]
                                       ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                   XEvent z; XNextEvent(e ,&z);
                                       ++*((N=XLookupKeysym
                                         (&z.xkey,0))-IT?
                                         N-LT? UP-N?& E:&
                                         J:& u: &h); --*(
                                         DN -N? N-DT ?N==
                                         RT?&u: & W:&h:&J
                                          ); } m=15*F/l;
                                          c+=(I=M/ l,l*H
                                          +I*M+a*X)*_; H
                                          =A*r+v*X-F*l+(
                                          E=.1+X*4.9/l,t
                                          =T*m/32-I*T/24
                                           )/S; K=F*M+(
                                           h* 1e4/l-(T+
                                           E*5*T*E)/3e2
                                           )/S-X*d-B*A;
                                           a=2.63 /l*d;
                                           X+=( d*l-T/S
                                            *(.19*E +a
                                            *.64+J/1e3
                                            )-M* v +A*
                                            Z)*_; l +=
                                            K *_; W=d;
                                            sprintf(f,
                                            "%5d  %3d"
                                            "%7d",p =l
                                           /1.7,(C=9E3+
                              O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                             X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                             *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                             179*v)/2312; select(p=0,0,0,0,&G); v-=(
                              W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                               )/107e2)*_; D=cos(o); E=sin(o); } }
```

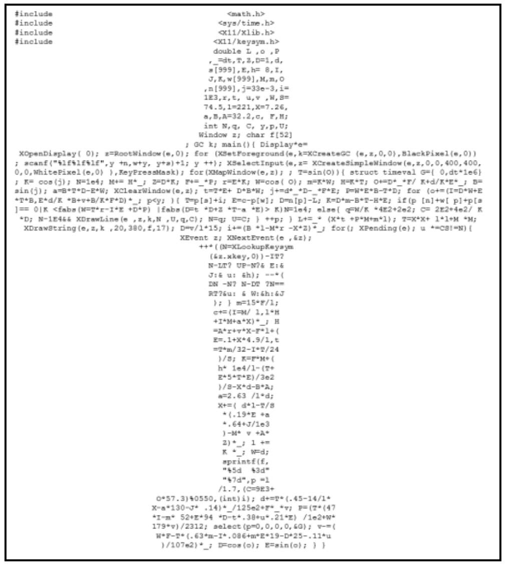
Hard to Read



```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Confusing

0.0014



```
#include                                   <math.h>
#include                                 <sys/time.h>
#include                                 <X11/Xlib.h>
#include                                <X11/keysym.h>
                                        double L ,o ,P
                                       ,_=dt,T,Z,D=1,d,
                                       s[999],E,h= 8,I,
                                       J,K,w[999],M,m,O
                                      ,n[999],j=33e-3,i=
                                      1E3,r,t, u,v ,W,S=
                                      74.5,l=221,X=7.26,
                                      a,B,A=32.2,c, F,H;
                                      int N,q, C, y,p,U;
                                     Window z; char f[52]
                                  ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                   XEvent z; XNextEvent(e ,&z);
                                       ++*((N=XLookupKeysym
                                         (&z.xkey,0))-IT?
                                         N-LT? UP-N?& E:&
                                         J:& u: &h); --*(
                                         DN -N? N-DT ?N==
                                         RT?&u: & W:&h:&J
                                          ); } m=15*F/l;
                                          c+=(I=M/ l,l*H
                                          +I*M+a*X)*_; H
                                          =A*r+v*X-F*l+(
                                          E=.1+X*4.9/l,t
                                          =T*m/32-I*T/24
                                           )/S; K=F*M+(
                                           h* 1e4/l-(T+
                                           E*5*T*E)/3e2
                                           )/S-X*d-B*A;
                                           a=2.63 /l*d;
                                           X+=( d*l-T/S
                                            *(.19*E +a
                                            *.64+J/1e3
                                            )-M* v +A*
                                            Z)*_; l +=
                                            K *_; W=d;
                                            sprintf(f,
                                            "%5d  %3d"
                                            "%7d",p =l
                                           /1.7,(C=9E3+
                              O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                             X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                             *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                             179*v)/2312; select(p=0,0,0,0,&G); v-=(
                              W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                               )/107e2)*_; D=cos(o); E=sin(o); } }
```

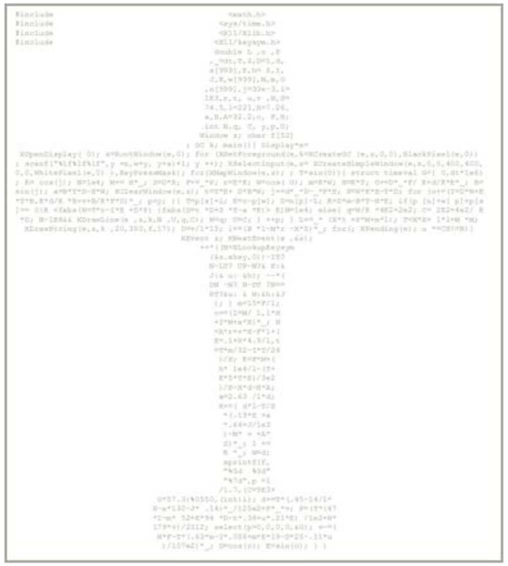
Hard to Read

0.82

```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Confusing

0.0014


[illegible]

Hard to Read

0.82

```
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Confusing

## Predicting Runtime Behavior

# Approaches to Predicting Behavior

| Dynamic Profiles | Static Heuristics |
|---|---|
| • Precise - full program path profiles<br>• Requires indicative workloads | • Cheap<br>• Only need program code<br>• Typically limited in scope |

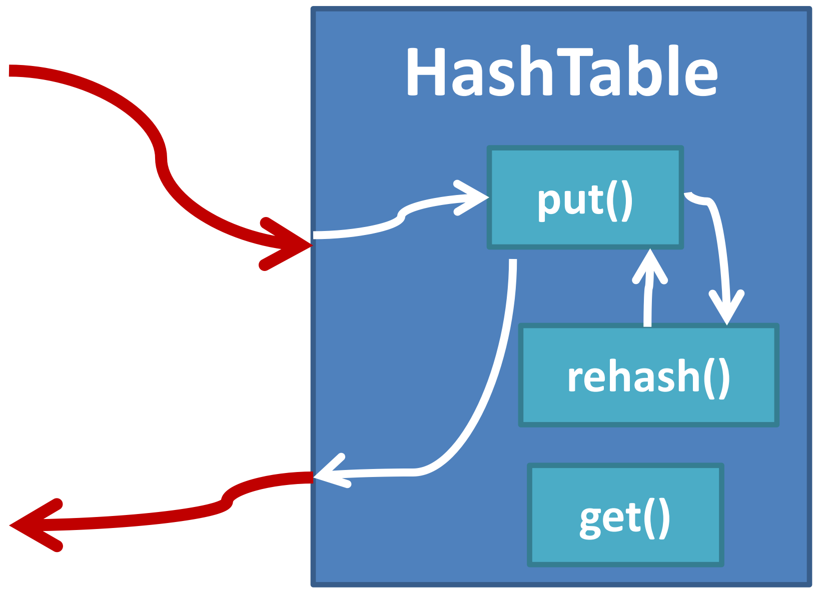
IF YOU REQUIRE WORKLOADS
INSTRUCTOR
YOU'RE GONNA HAVE A BAD TIME

## Static Heuristics

- D. W. Wall. Predicting program behavior using real or estimated profiles. In ACM Conf. on Programming Language Design and Implementation (PLDI'91), pages 59-70, June 1991.
- T. Ball and J. R. Larus. Branch prediction for free. In ACM Conf. on Programming Language Design and Implementation (PLDI'93), pages 300-313, June 1993.
- Boogerd, C. and Moonen, L. Prioritizing Software Inspection Results using Static Profiling, *Source Code Analysis and Manipulation, 2006. SCAM '06.* pp.149-160, Sept. 2006

# Intra-class static path profiles

- Precision similar to a dynamic profiler
- Workloads not required



## Key idea

```
public V put(K key , V value)
{
    if ( value == null )
      throw new Exception();

    if ( count >= threshold )
      rehash();

    index = key.hashCode() % length;

    table[index] = new Entry(key, value);
    count++;

    return value;
}
```

*from java.util.HashTable jdk6.0

```
public V put(K key , V value)
{
    if ( value == null )
      throw new Exception();

    if ( count >= threshold )
      rehash();

    index = key.hashCode() % length;

    table[index] = new Entry(key, value);
    count++;

    return value;
}
```

Exception

Invocation that changes a lot of program state.

Computation

*from java.util.HashTable jdk6.0

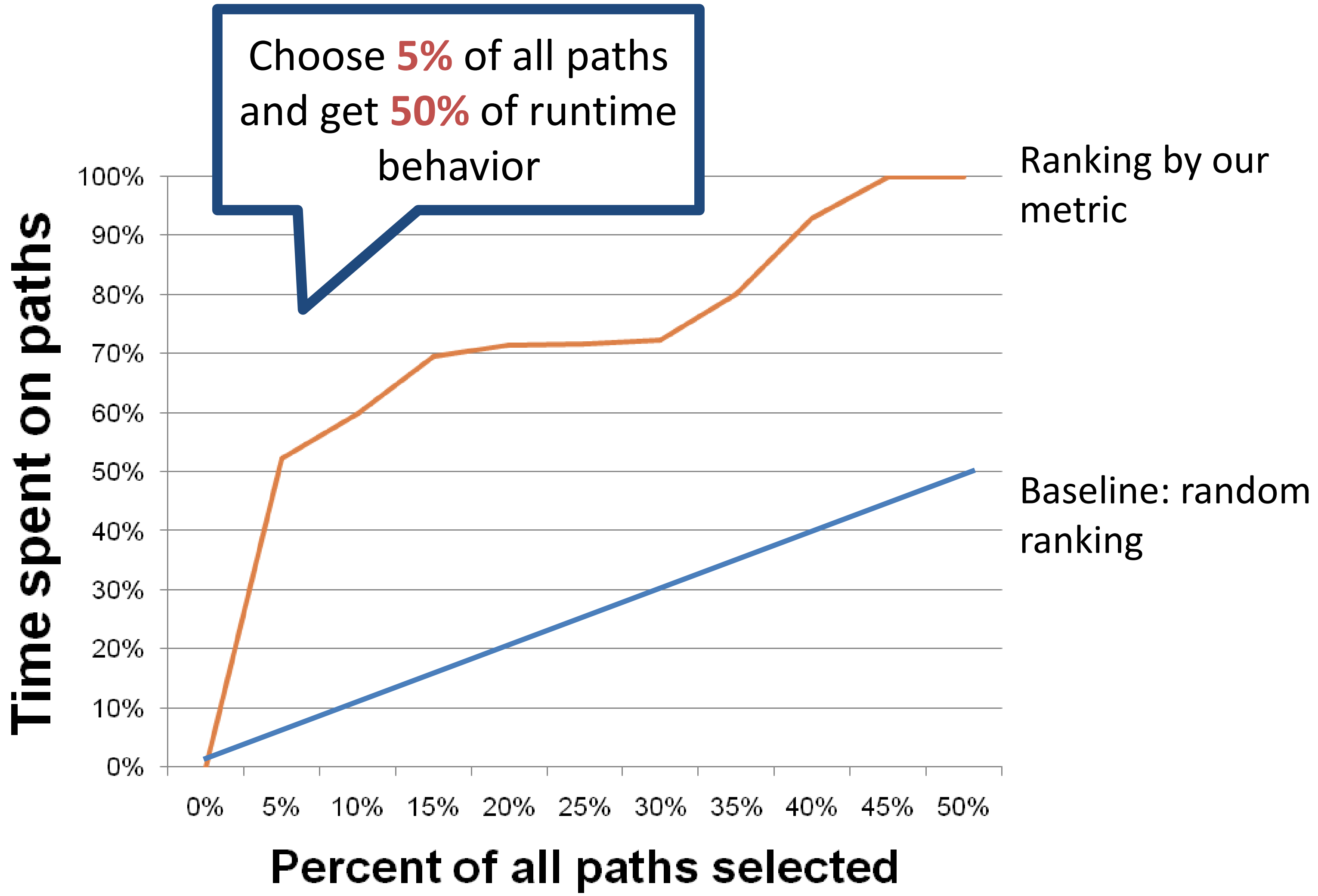
## Static Path Profiling

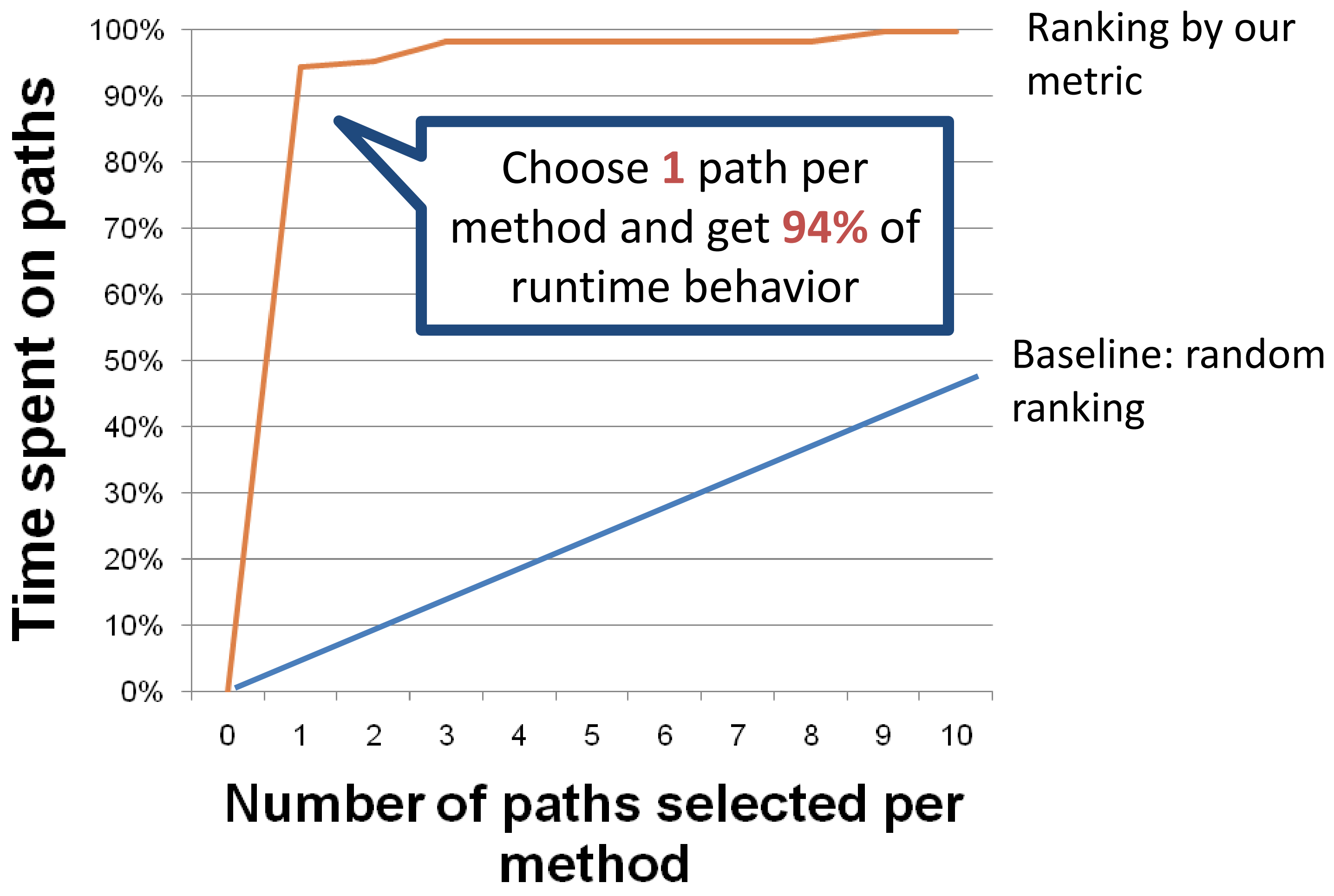
Research questions:

- What static code features are predictive of path execution frequency?
- Can we derive a accurate descriptive model for runtime behavior?

## Approach

- Build a descriptive model of path execution frequency
- Features: length of path, presence of exceptions, number of variables written …
- Train and cross-validate on SPEC Java benchmarks


Choose 5% of all paths and get 50% of runtime behavior
Ranking by our metric
Baseline: random ranking
Time spent on paths
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
0%
5%
10%
15%
20%
25%
30%
35%
40%
45%
50%
Percent of all paths selected


Time spent on paths
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
0
1
2
3
4
5
6
7
8
9
10
Number of paths selected per method
Ranking by our metric
Baseline: random ranking
Choose 1 path per method and get 94% of runtime behavior

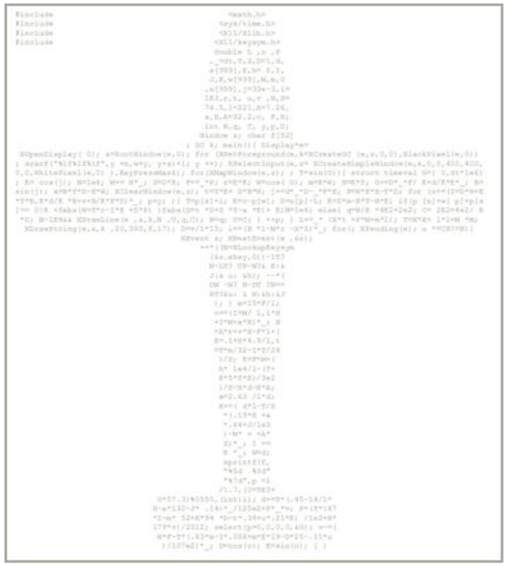
# Applications for Profiles

- Profile guided optimization
- Complexity/Runtime estimation
- Anomaly detection
- Significance of difference between program versions
- Prioritizing output from other analyses
- **Documentation**

## Conclusion

- A formal model that statically predicts relative dynamic path execution frequencies
- The promise of helping other program analyses and transformations

0.0014



Hard to Read

0.82

```java
import java.math.BigDecimal;

class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new BigDecimal(1.10);
    System.out.println(payment.subtract(cost));
  }
}
```

Confusing

## Documentation Synthesis

# Classic Approaches to Understandability

Improving

- Code Reviews
- Training
- Languages
- Documentation

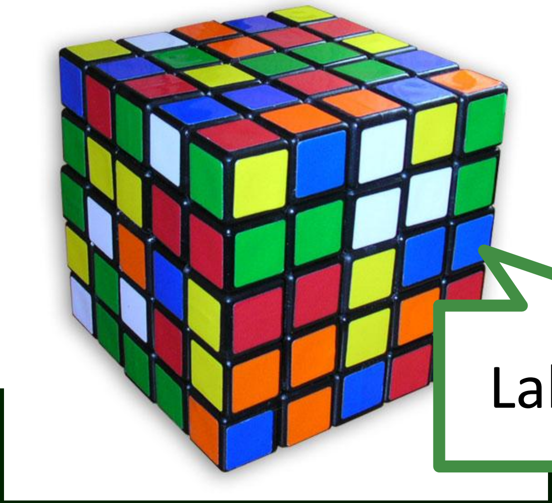
Compensating

- Testing
- Verification
- Other Program Analyses

# Classic Approaches to Understandability

## Improving

- Code Reviews
- Training
- Languages
- Documentation

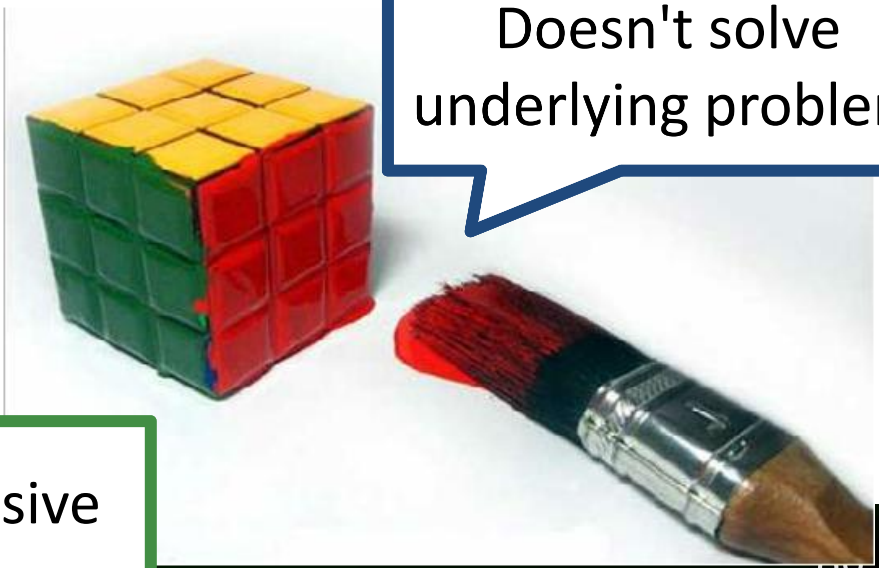


Labor Intensive

## Compensating

- Testing
- Verification
- Other Program Analyses

Doesn't solve underlying problem



Compensating

- Testing
- Verification
- Other Program Analyses

Use these tools…

## Improving

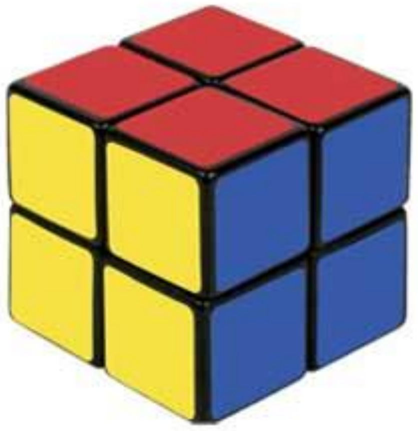
- Code Reviews
- Training
- Languages
- **Documentation**

To do this.

Compensating

- Testing
- Verification
- Other Program Analyses

Use these tools…



The most significant barrier to code reuse is “software is too difficult to understand or is poorly documented.”

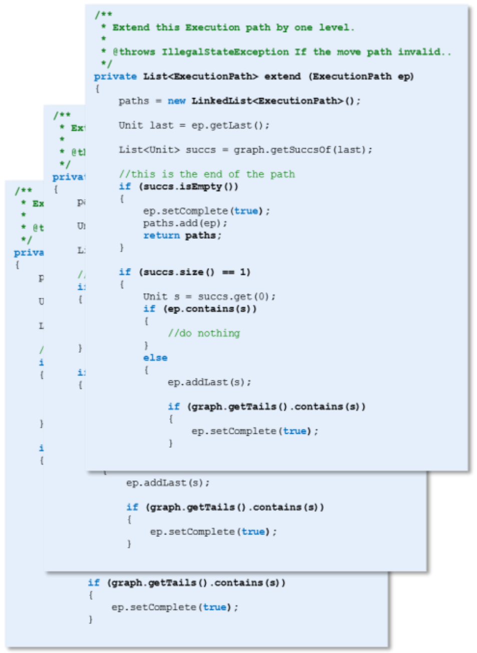
NASA Software Reuse Working Group. Software Reuse Survey. http://www.esdswg.com/softwarereuse/Resources/library/working_group_documents/survey2005

## Source Code Documentation

- Describes some **important** aspect of the code in a way that's **easier to understand.**

- Explanations/Summaries of Behavior
- Pre/Post Conditions, Caveats
- Usage Examples
- …

# Automatic Documentation



```
/**
 * Extend this Execution path by one level.
 *
 * @throws IllegalStateException If the move path invalid..
 */
private List<ExecutionPath> extend (ExecutionPath ep)
{
    paths = new LinkedList<ExecutionPath>();

    Unit last = ep.getLast();

    List<Unit> succs = graph.getSuccsOf(last);

    //this is the end of the path
    if (succs.isEmpty())
    {
        ep.setComplete(true);
        paths.add(ep);
        return paths;
    }

    if (succs.size() == 1)
    {
        Unit s = succs.get(0);
        if (ep.contains(s))
        {
            //do nothing
        }
        else
        {
            ep.addLast(s);

            if (graph.getTails().contains(s))
            {
                ep.setComplete(true);
            }
```

```
            ep.addLast(s);

            if (graph.getTails().contains(s))
            {
                ep.setComplete(true);
            }
```

```
            if (graph.getTails().contains(s))
            {
                ep.setComplete(true);
            }
```

Program Code

Synthesis Tool

When calling LastPage format(String s)
    If s is not null and s.split ("[-]+").length != 2
        return s.split ("[-]+")[0] instead of ""

When calling EntryEditor getExtra()
    If ed.getFieldName().equals("editor")
        call contentSelectors
.add(FieldContentSelector)

Documentation

## Automatic Documentation

- Cheap
- Always up-to-date
- Complete
- Well-defined trust properties
- Structured (Searchable)

## Exceptions

```
IllegalStateException thrown when
        getLocation() is not Europe
```

Raymond P. L. Buse and Westley R.Weimer. Automatic Documentation Inference for Exceptions. In International Symposium on Software Testing and Analysis, Seattle, WA, USA, 2008.

## Program Changes

```
When arg0 == null
  return -1 instead of
arg0.toString()
```

Raymond P.L. Buse and Westley R. Weimer. Automatically documenting program changes. In International Conference on Automated Software Engineering, Antwerp, Belgium, 2010.

## API Usage Examples

```
Iterator iter =
         SOMETHING.iterator();
while( iter.hasNext() )
{
   Object o = iter.next();
   //Do something with o
}
```

Raymond P. L. Buse and Westley Weimer. Synthesizing API Usage Examples. In International Conference on Software Engineering [To Appear], Zurich, Switzerland, 2012.

## Exceptions

```
IllegalStateException thrown when
        getLocation() is not Europe
```

Raymond P. L. Buse and Westley R.Weimer. Automatic Documentation Inference for Exceptions. In International Symposium on Software Testing and Analysis, Seattle, WA, USA, 2008.

## Program Changes

```
When arg0 == null
  return -1 instead of
arg0.toString()
```

Raymond P.L. Buse and Westley R. Weimer. Automatically documenting program changes. In International Conference on Automated Software
0.

## API Usage Examples

```
Iterator iter =
         SOMETHING.iterator();
while( iter.hasNext() )
{
   Object o = iter.next();
   //Do something with o
}
```

This Talk

Raymond P. L. Buse and Westley Weimer. Synthesizing API Usage Examples. In International Conference on Software Engineering [To Appear], Zurich, Switzerland, 2012.

“The greatest obstacle to learning an API … is insufficient or inadequate examples”

Martin Robillard. What Makes APIs Hard to Learn? Answers from Developers. IEEE Software, 26(6):27-34, 2009.

## Synthesizing API Examples

Research questions:

- What makes a good example?
- Can we create good examples automatically?

# Sources of Examples

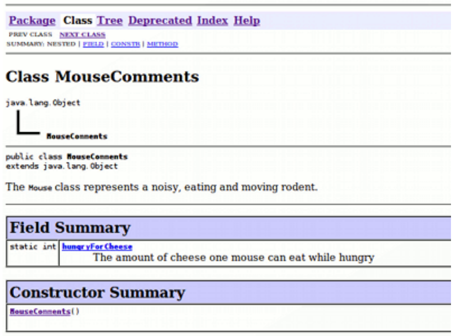
stackoverflow

Google code search labs

JExamples

msdn® Microsoft® Developer Network

JavaDoc



Package Class Tree Deprecated Index Help
PREV CLASS NEXT CLASS
SUMMARY: NESTED | FIELD | CONSTR | METHOD

**Class MouseComments**

```
java.lang.Object
  └ MouseComments

public class MouseComments
extends java.lang.Object
```

The `Mouse` class represents a noisy, eating and moving rodent.

| **Field Summary** | |
|---|---|
| static int | hungryForCheese<br>The amount of cheese one mouse can eat while hungry |

| **Constructor Summary** |
|---|
| MouseComments() |

## Search-based examples

```
BufferedReader reader = new BufferedReader(new
                                      InputStreamReader(page));
try {String line = reader.readLine();
while (line != null) {
if (line.matches(substituteWikiWord(wikiWord,newTopicPattern)))
{
```

Query: BufferedReader

# Hand-Crafted Examples

Example Data

```
FileOutputStream fos = new FileOutputStream("t.tmp");
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeInt(12345);
oos.writeObject("Today");
oos.writeObject(new Date());
oos.close();
```

Query: java.util.ObjectOutputStream

Complete

## Hand-Crafted Examples

Abstract Initialization

```
int glyphIndex = ...;
GlyphMetrics metrics =
GlyphVector.getGlyphMetrics(glyphIndex);
int isStandard = metrics.isStandard();
float glyphAdvance = metrics.getAdvance();
```

Query: java.awt.font.GlyphMetrics

# Hand-Crafted Examples

```
for(char c = iter.first();
     c != CharacterIterator.DONE;
     c = iter.next()) {
   processChar(c);
}
```

Hole

Query: java.text.CharacterIterator

```
try {
file.delete();
} catch (IOException exc) {
  // failed to delete, do error handling here
}
return FileVisitResult.CONTINUE;
```

Hole

Query: java.nio.FileVisitor

## Our API Examples

```
FileReader fReader; //initialized previously
BufferedReader br = new BufferedReader(fReader);
while(br.ready()) {
    String line = br.readLine();
    //do something with line
}
br.close();
```

Query: java.util.BufferedReader

# Our API Examples

Common variable names

Abstract Initialization

```
FileReader fReader; //initialized previously
BufferedReader br = new BufferedReader(fReader);
while(br.ready()) {
    String line = br.readLine();
    //do something with line
}
br.close();
```

"Holes"

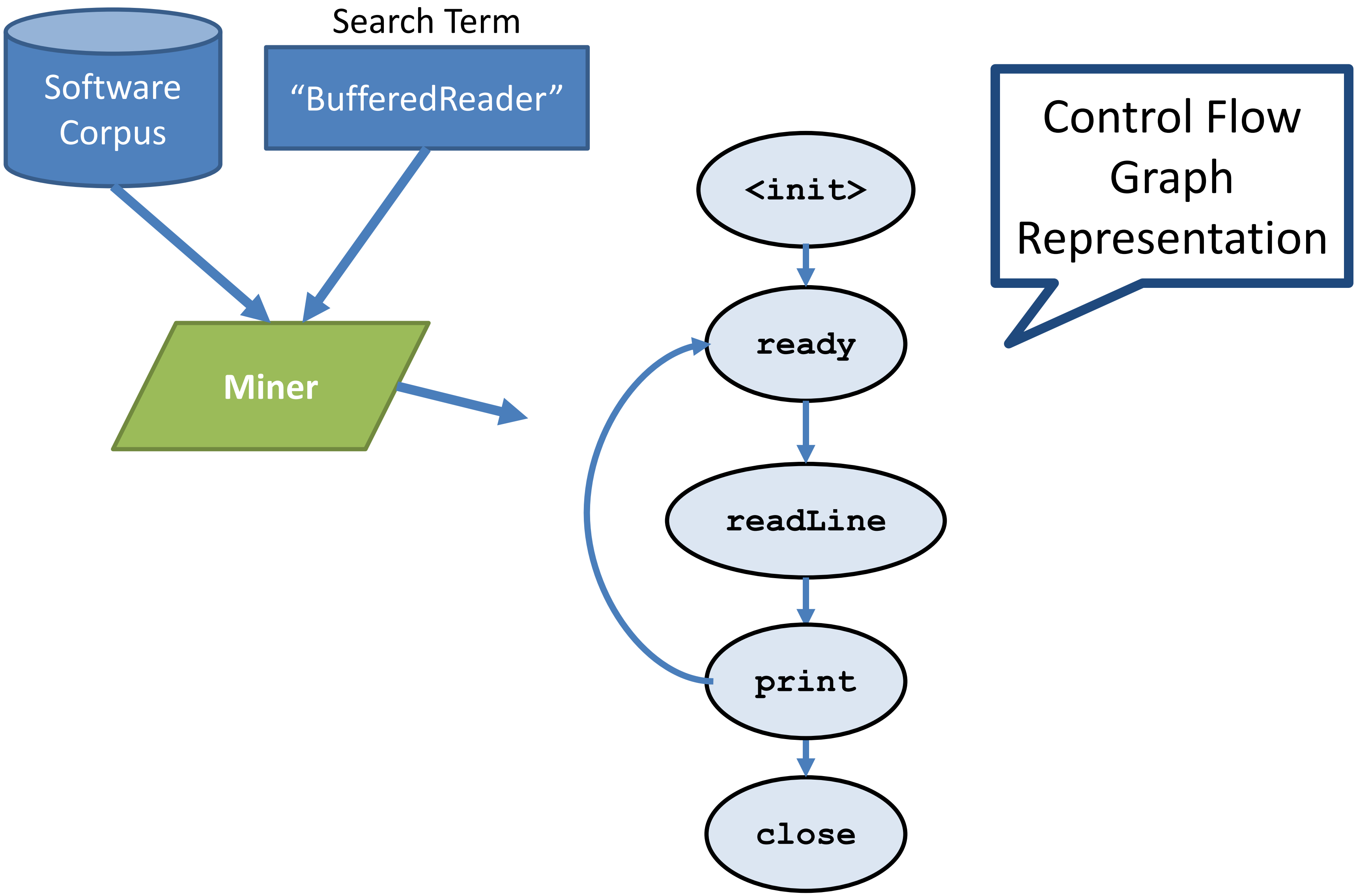
Query: java.util.Bu

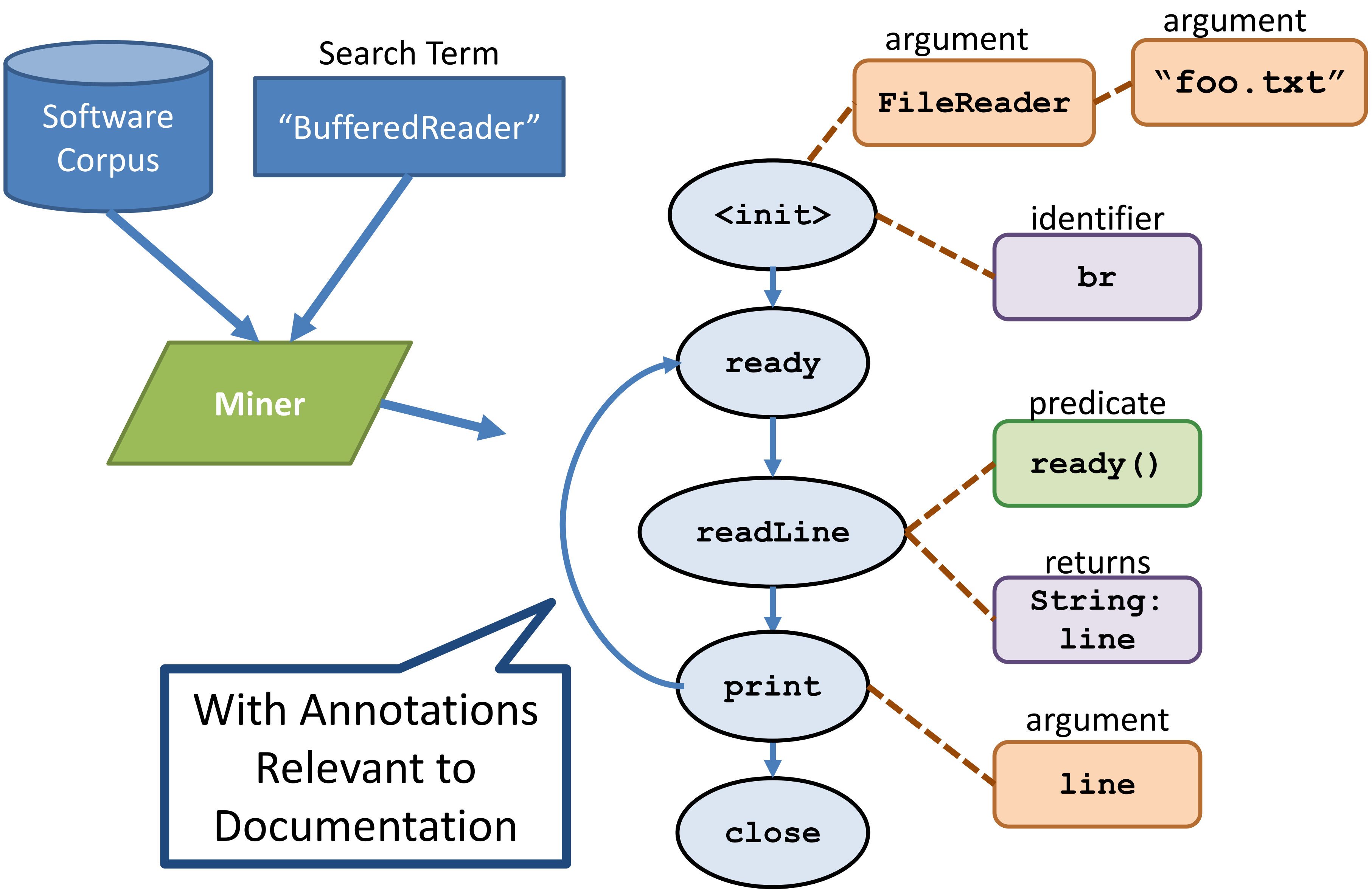
Complete

## Synthesis

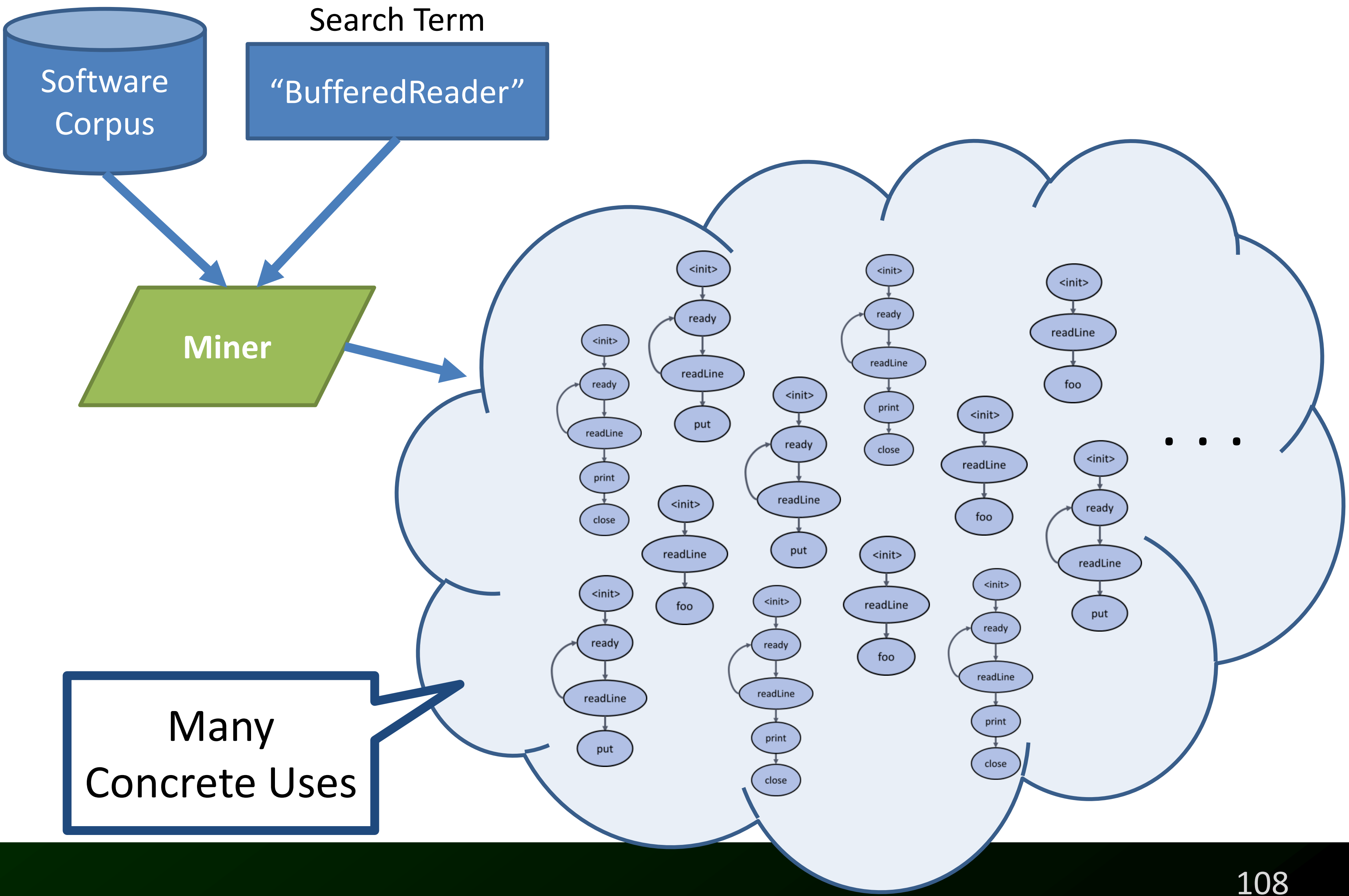
## Approach

- **Mine** usages from an existing program corpus
  - Similar to *Specification Mining*
- **Learn** common patterns
- **Abstract** representative examples


Software Corpus
Search Term
"BufferedReader"
Miner
<init>
ready
readLine
print
close
Control Flow Graph Representation



With Annotations Relevant to Documentation


Search Term
Software Corpus
"BufferedReader"
Miner
<init>
ready
readLine
put
print
close
foo
. . .
Many Concrete Uses

## Multiple Common Patterns

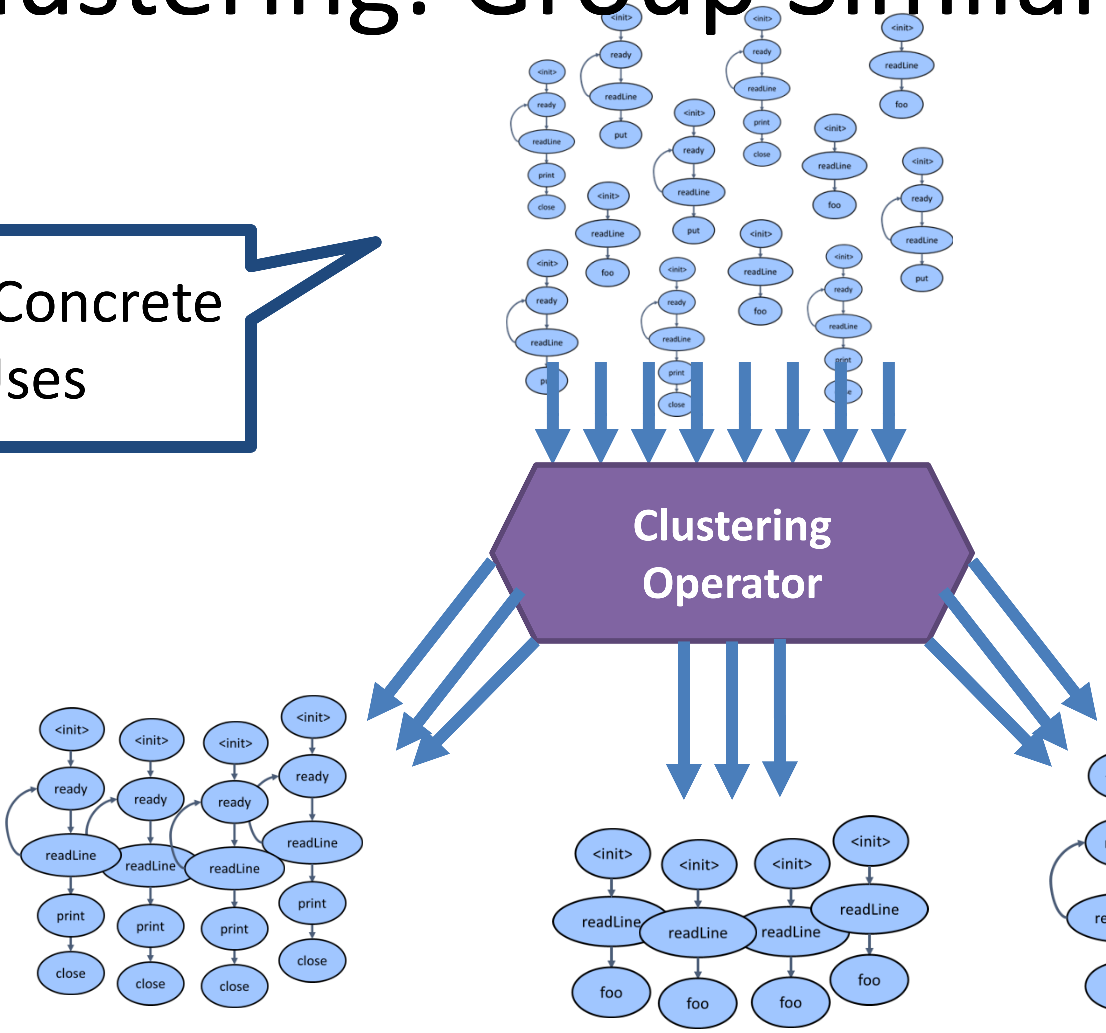
```
BufferedReader br = new BufferedReader(new FileReader("foo.in"));
while( br.ready() )
{
    String s = br.readLine();
    //Do something with s
}
```

```
BufferedReader br = new BufferedReader(new FileReader("foo.in"));
String s;
while( ( s = br.readLine() ) != null )
{
    //Do something with s
}
```

## Clustering: Group Similar Patterns

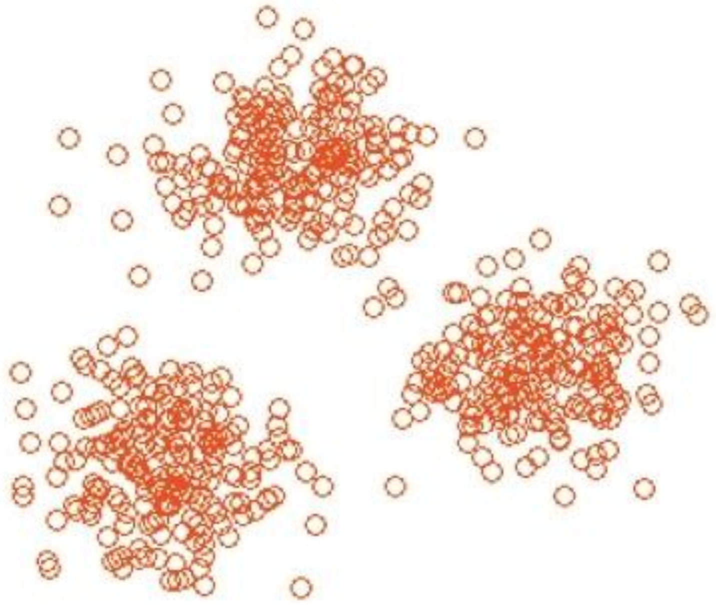
Many Concrete Uses

… Grouped Into A Few Different Patterns



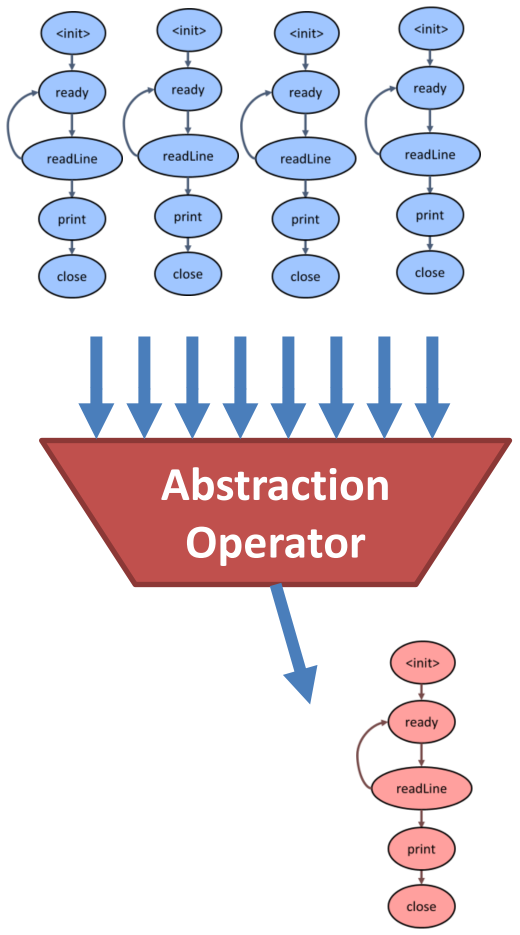
## Clustering

- k-medoids
- Distance metric captures difference in **order of statements** and **types of objects**

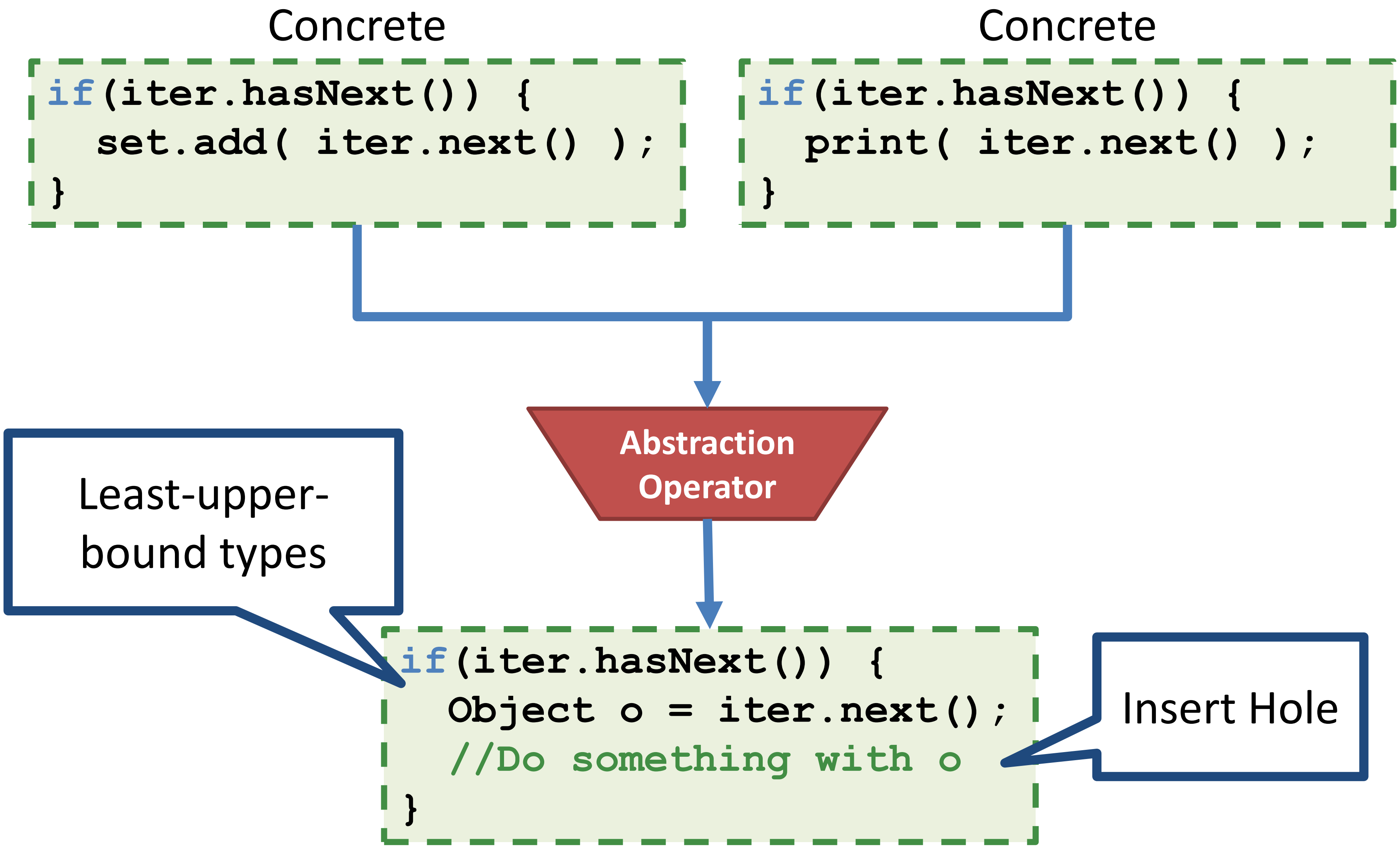


# Abstraction

Many Concrete Examples



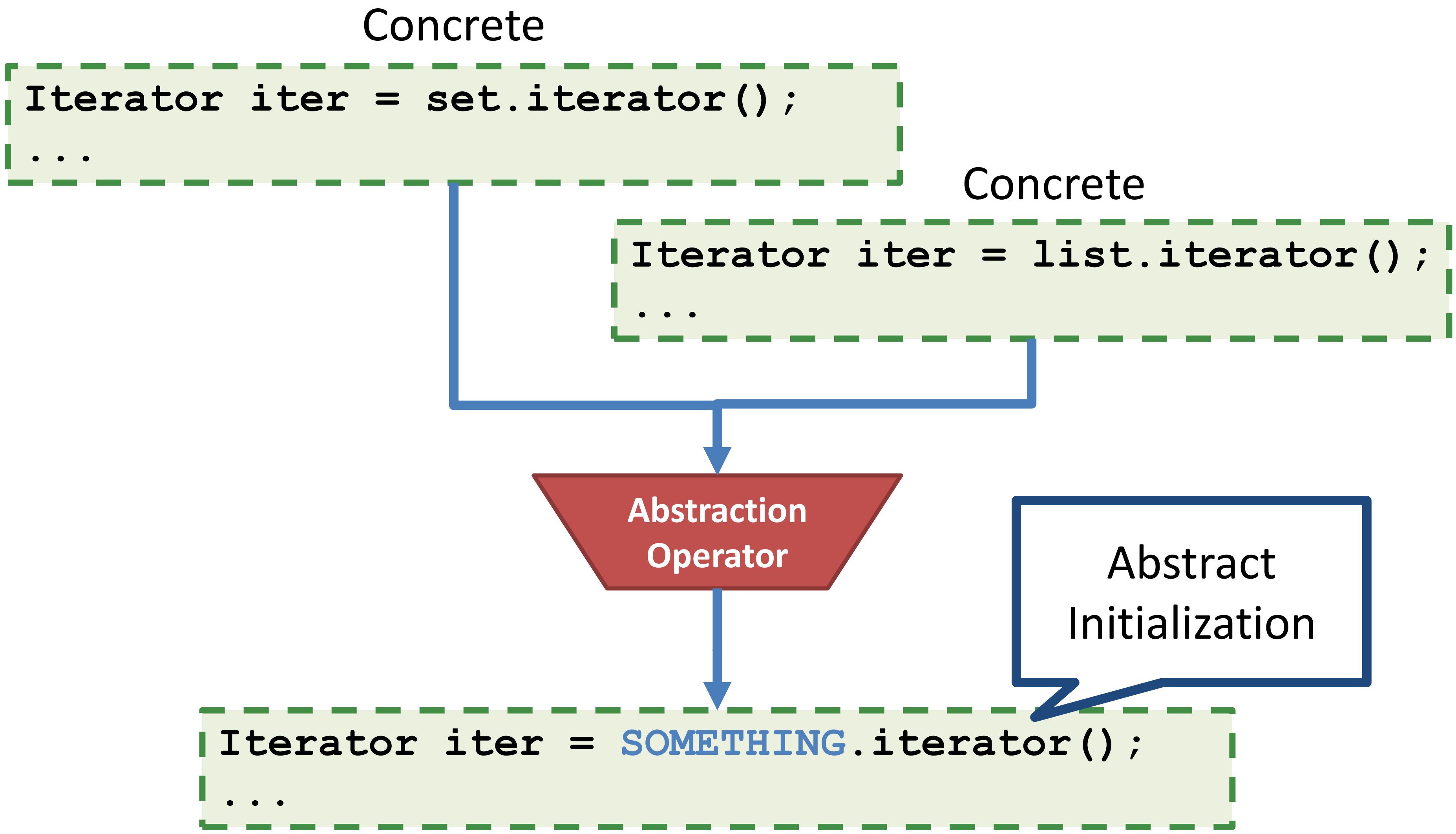
... Into One Abstract Example



```
if(iter.hasNext()) {
  set.add( iter.next() );
}
```

```
if(iter.hasNext()) {
  print( iter.next() );
}
```

```
if(iter.hasNext()) {
  Object o = iter.next();
  //Do something with o
}
```

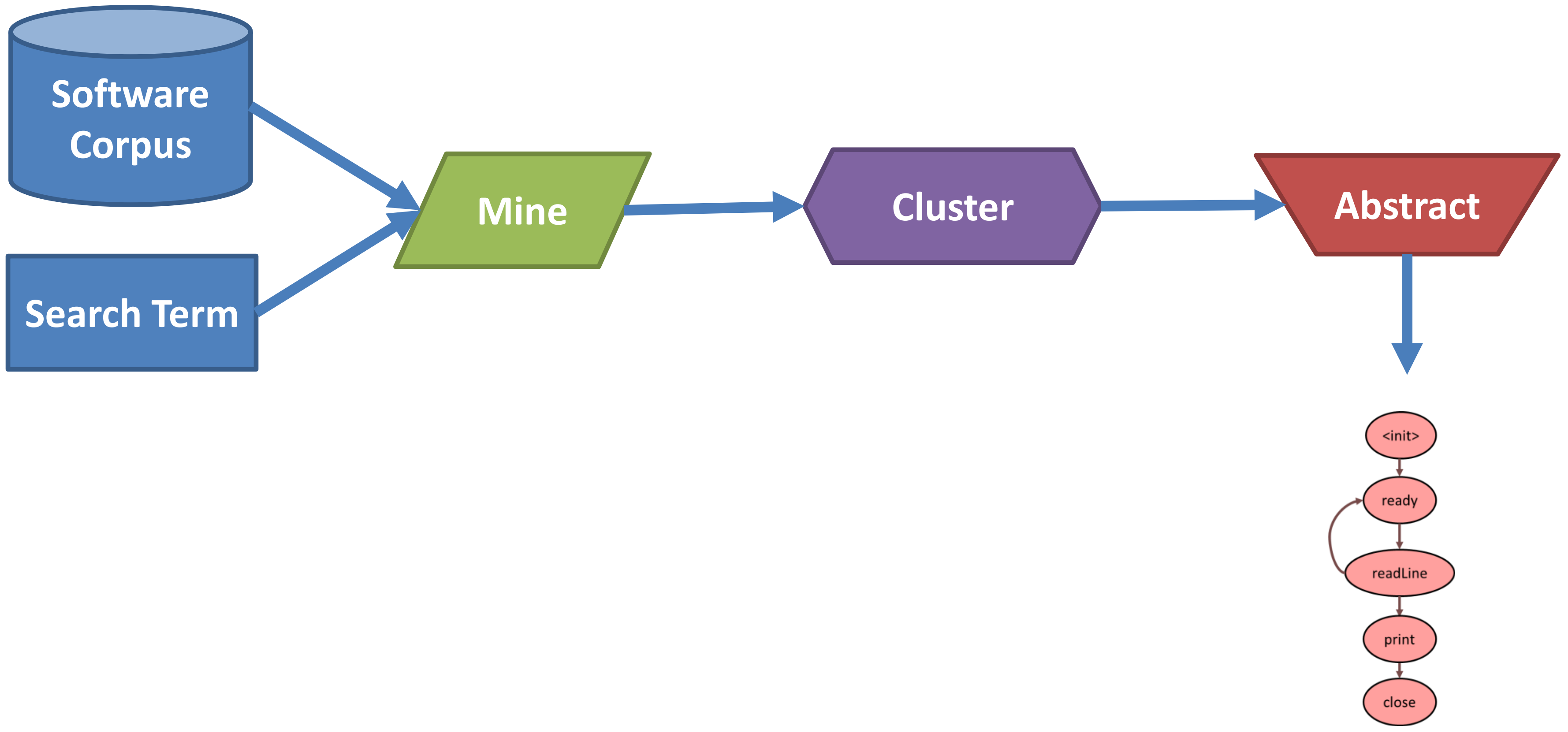


```
Iterator iter = set.iterator();
...
```

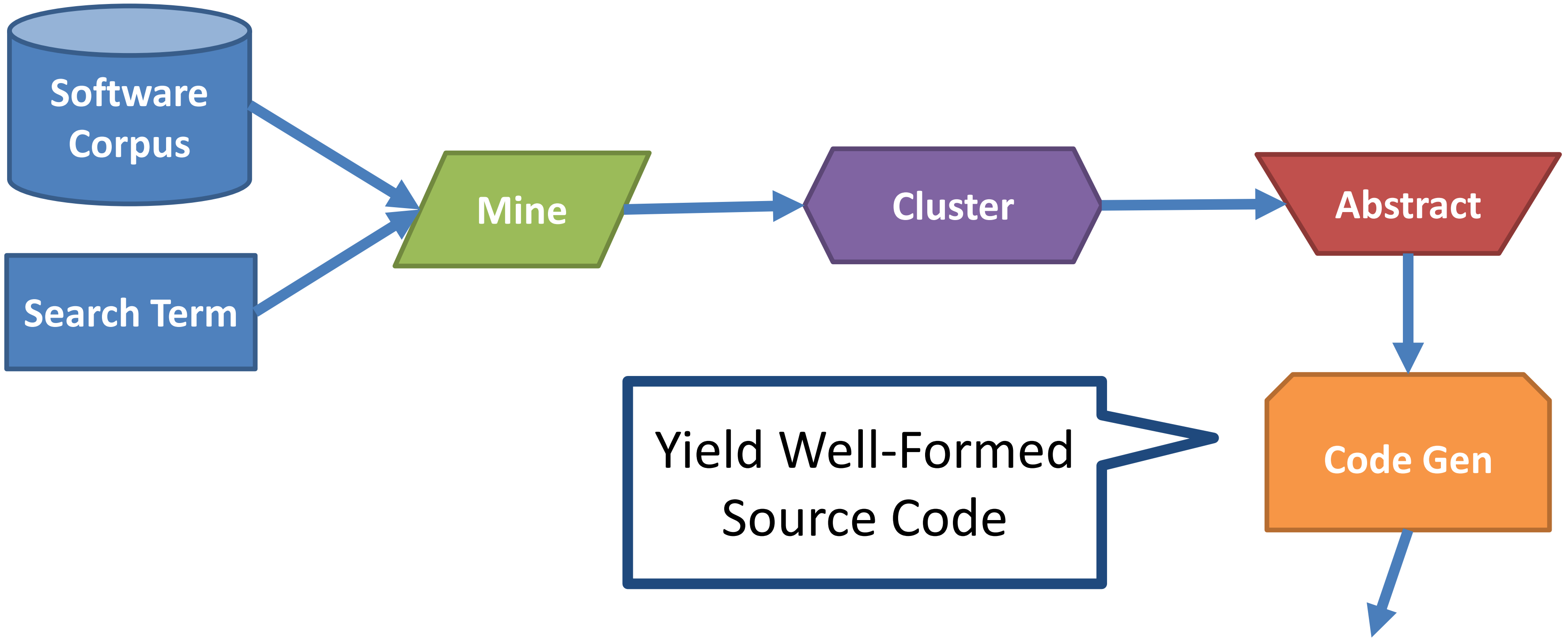
```
Iterator iter = list.iterator();
...
```

```
Iterator iter = SOMETHING.iterator();
...
```

## Recap


Software Corpus
Search Term
Mine
Cluster
Abstract
<init>
ready
readLine
print
close

## Recap



```
Calendar calendar = Calendar.getInstance();
Date d = calendar.getTime();
//Do something with d
```

## Examples

```
Calendar calendar = Calendar.getInstance();
Date d = calendar.getTime();
//Do something with d
```

Query: java.util.Calendar

## Examples

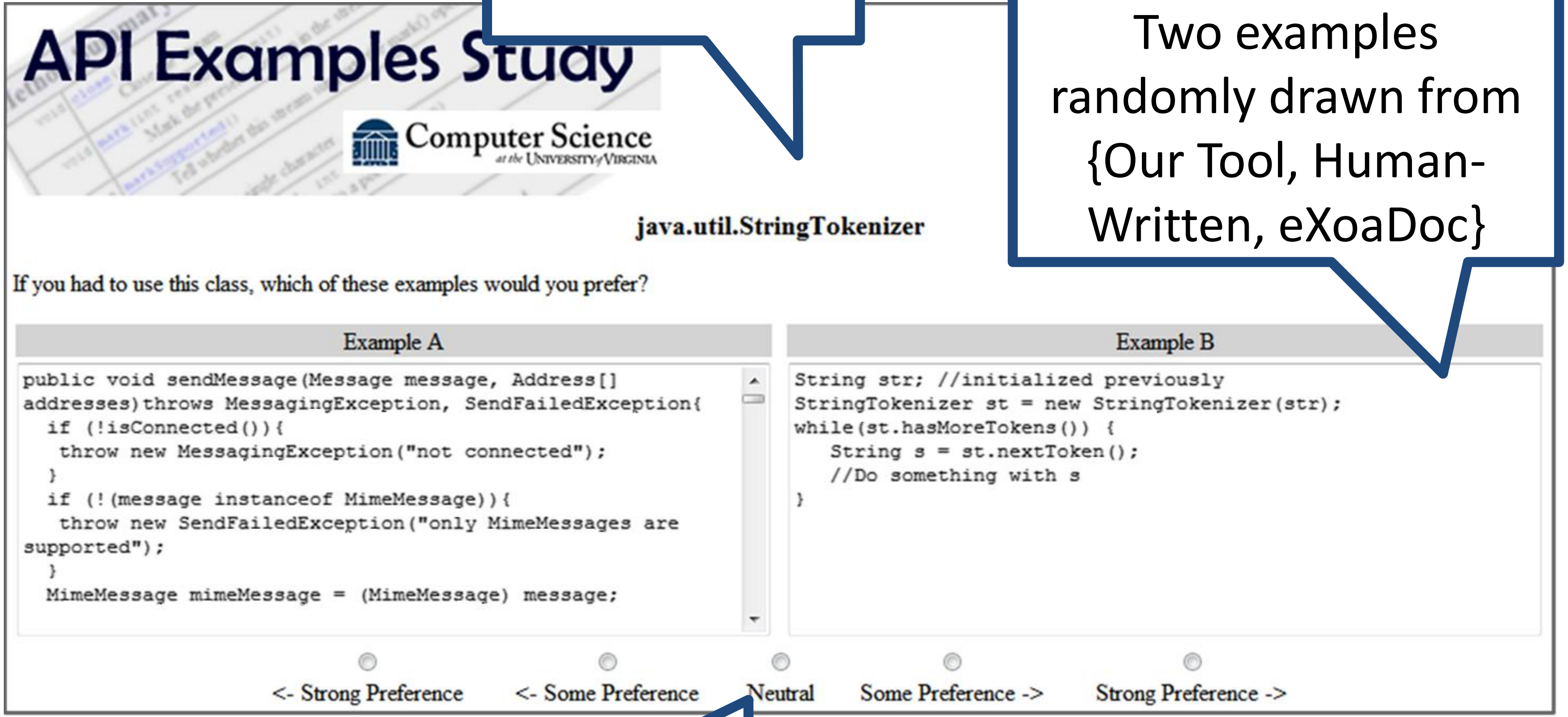
```java
String regex; //initialized previously
String input; //initialized previously
Pattern pattern = Pattern.compile(regex);
Matcher m = pattern.matcher(input);
//Do something with m
```

Query: java.util.regex.Pattern

## Limitations

- Can’t always be perfectly precise
  - E.g., Aliasing, Types
  - Conservative analysis preserves correctness
- Common usage is not always best
  - E.g., poor exception handling
  - Guarantee representative examples
- Not all APIs have indicative patterns
- Some patterns are difficult to find
  - Message passing over network etc.

## Evaluation



API Examples Study

Computer Science at the University of Virginia

java.util.StringTokenizer

If you had to use this class, which of these examples would you prefer?

Example A

```
public void sendMessage(Message message, Address[]
addresses)throws MessagingException, SendFailedException{
  if (!isConnected()){
   throw new MessagingException("not connected");
  }
  if (!(message instanceof MimeMessage)){
   throw new SendFailedException("only MimeMessages are
supported");
  }
  MimeMessage mimeMessage = (MimeMessage) message;
```

Example B

```
String str; //initialized previously
StringTokenizer st = new StringTokenizer(str);
while(st.hasMoreTokens()) {
   String s = st.nextToken();
   //Do something with s
}
```

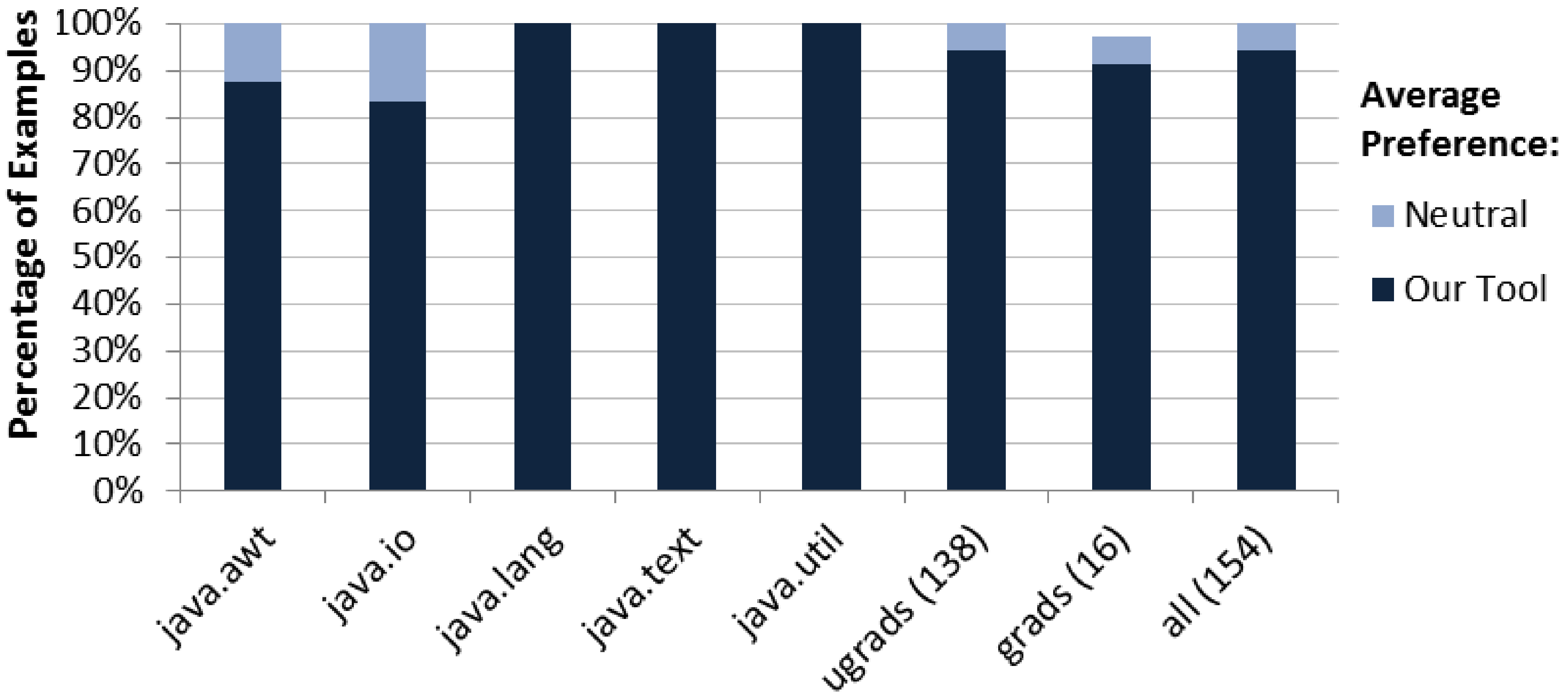
<- Strong Preference | <- Some Preference | Neutral | Some Preference -> | Strong Preference ->

Name of Class

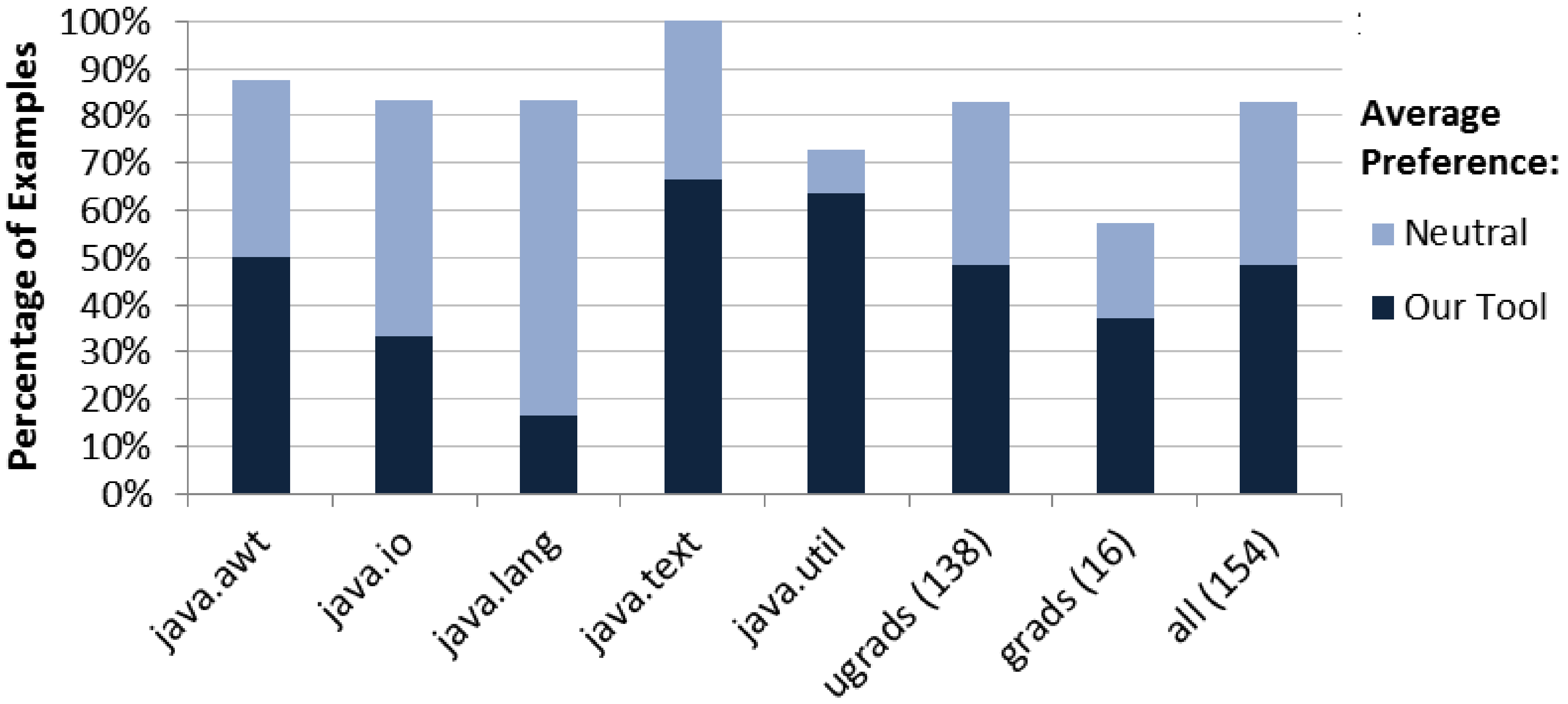
Two examples randomly drawn from {Our Tool, Human-Written, eXoaDoc}

Participant specifies preference

## Comparison to Code Search

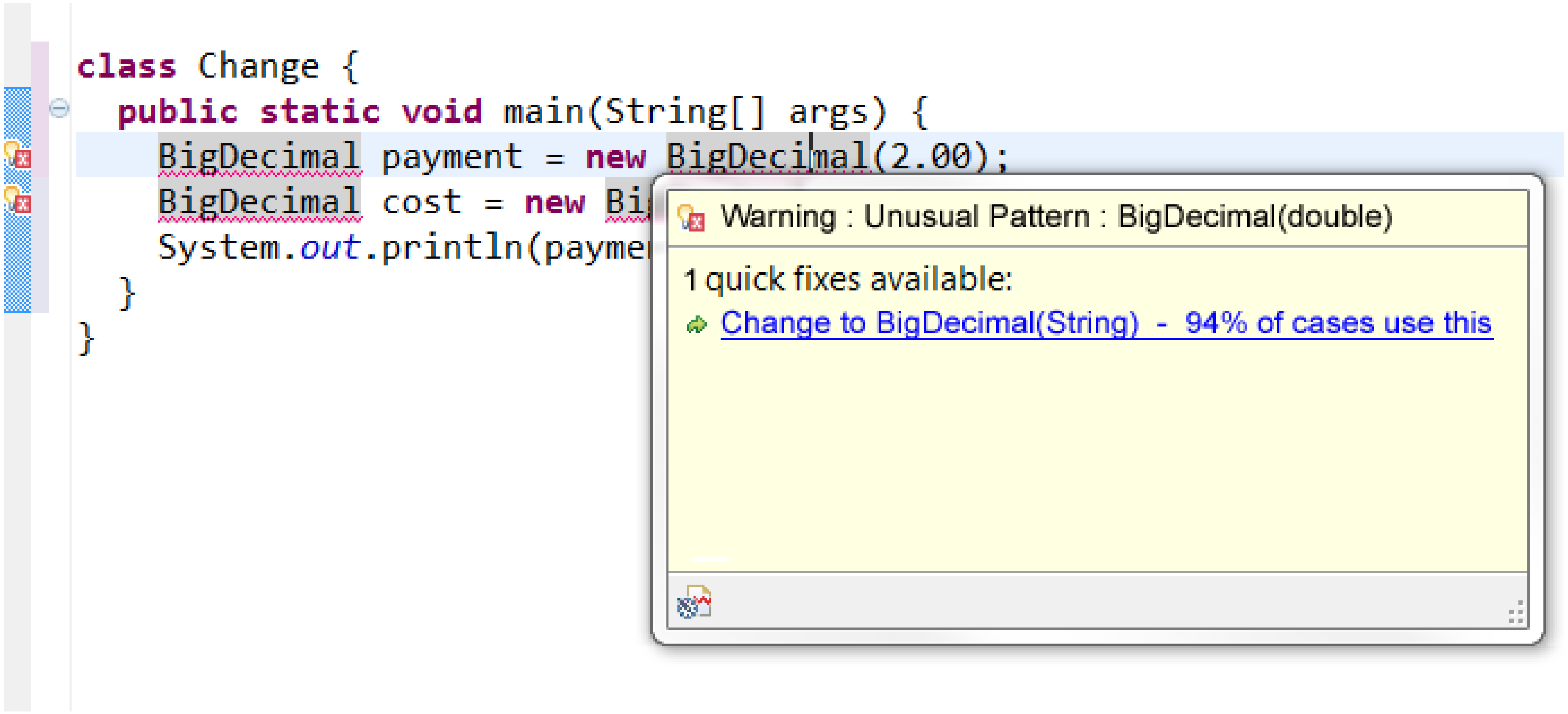


# Comparison to Human-Written



## Use Cases

# Warnings for Likely Mistakes



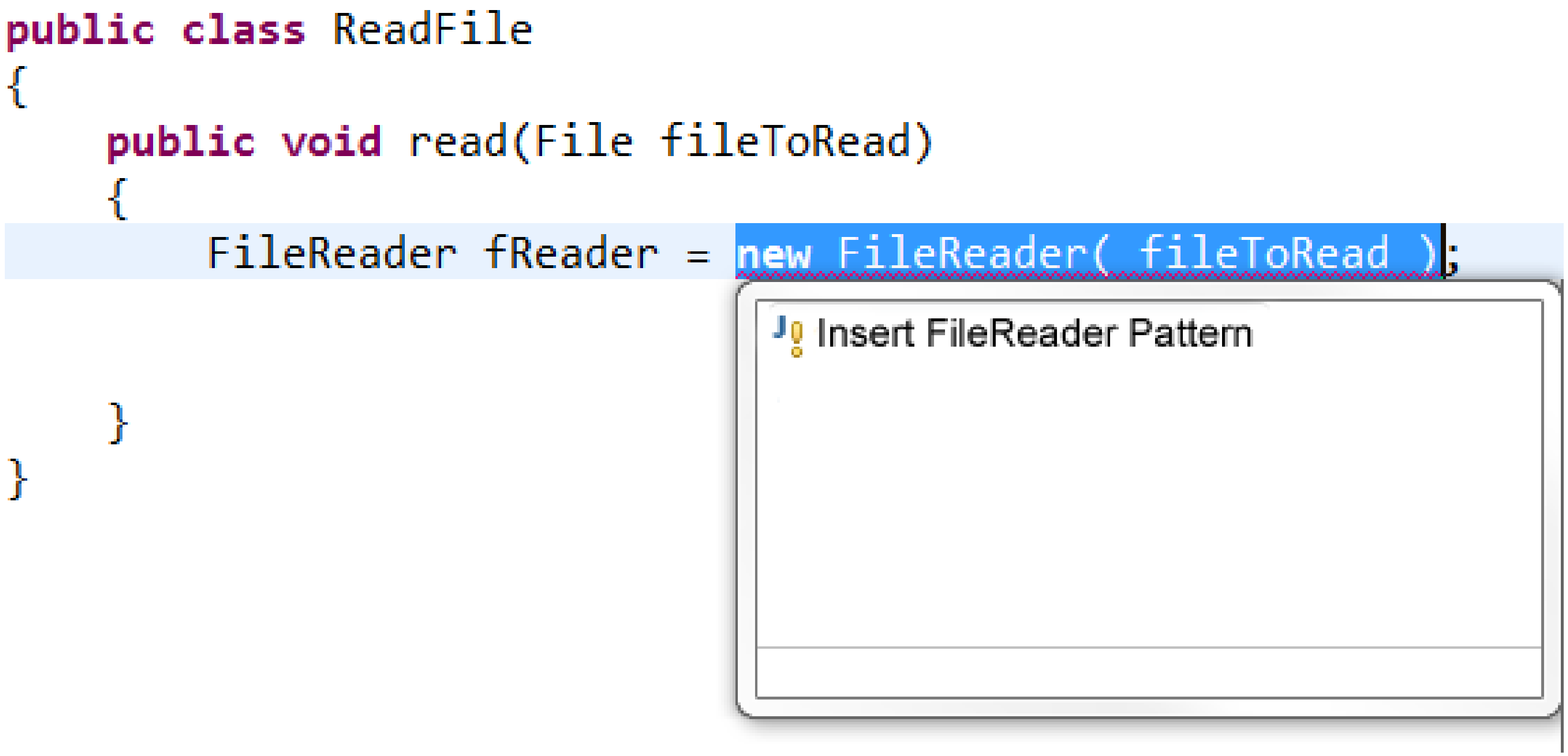
```
class Change {
  public static void main(String[] args) {
    BigDecimal payment = new BigDecimal(2.00);
    BigDecimal cost = new Bi
    System.out.println(payme
  }
}
```

Warning : Unusual Pattern : BigDecimal(double)

1 quick fixes available:

Change to BigDecimal(String) - 94% of cases use this

# Auto Completion

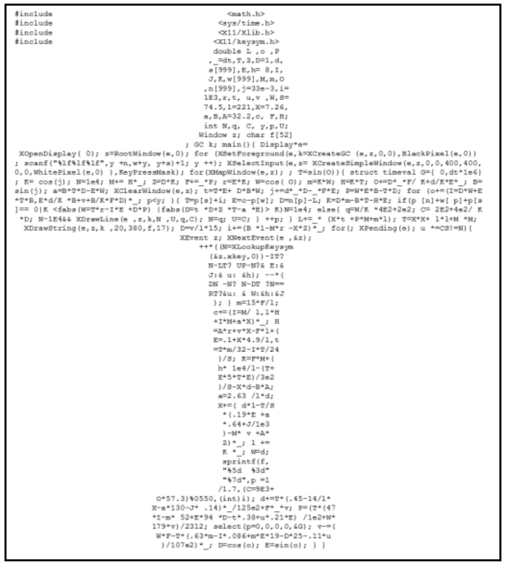


```
public class ReadFile
{
    public void read(File fileToRead)
    {
        FileReader fReader = new FileReader( fileToRead );


    }
}
```

Insert FileReader Pattern

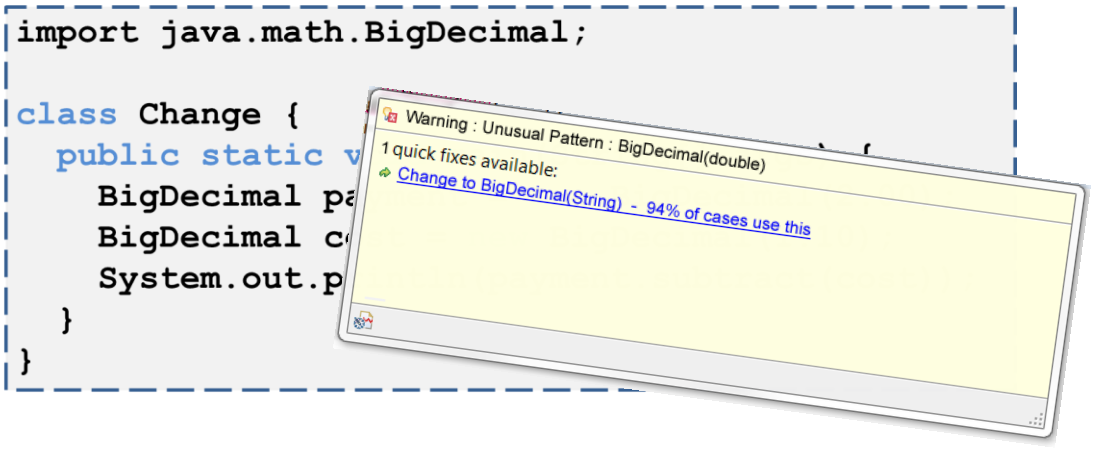
## Conclusion

0.0014



```
#include                                        <math.h>
#include                                      <sys/time.h>
#include                                      <X11/Xlib.h>
#include                                     <X11/keysym.h>
                                             double L ,o ,P
                                            ,_=dt,T,Z,D=1,d,
                                            s[999],E,h= 8,I,
                                            J,K,w[999],M,m,O
                                           ,n[999],j=33e-3,i=
                                           1E3,r,t, u,v ,W,S=
                                           74.5,l=221,X=7.26,
                                           a,B,A=32.2,c, F,H;
                                           int N,q, C, y,p,U;
                                          Window z; char f[52]
                                        ; GC k; main(){ Display*e=
 XOpenDisplay( 0); z=RootWindow(e,0); for (XSetForeground(e,k=XCreateGC (e,z,0,0),BlackPixel(e,0))
; scanf("%lf%lf%lf",y +n,w+y, y+s)+1; y ++); XSelectInput(e,z= XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0) ),KeyPressMask); for(XMapWindow(e,z); ; T=sin(O)){ struct timeval G={ 0,dt*1e6}
; K= cos(j); N=1e4; M+= H*_; Z=D*K; F+=_*P; r=E*K; W=cos( O); m=K*W; H=K*T; O+=D*_*F/ K+d/K*E*_; B=
sin(j); a=B*T*D-E*W; XClearWindow(e,z); t=T*E+ D*B*W; j+=d*_*D-_*F*E; P=W*E*B-T*D; for (o+=(I=D*W+E
*T*B,E*d/K *B+v+B/K*F*D)*_; p<y; ){ T=p[s]+i; E=c-p[w]; D=n[p]-L; K=D*m-B*T-H*E; if(p [n]+w[ p]+p[s
]== 0|K <fabs(W=T*r-I*E +D*P) |fabs(D=t *D+Z *T-a *E)> K)N=1e4; else{ q=W/K *4E2+2e2; C= 2E2+4e2/ K
 *D; N-1E4&& XDrawLine(e ,z,k,N ,U,q,C); N=q; U=C; } ++p; } L+=_* (X*t +P*M+m*l); T=X*X+ l*l+M *M;
  XDrawString(e,z,k ,20,380,f,17); D=v/l*15; i+=(B *l-M*r -X*Z)*_; for(; XPending(e); u *=CS!=N){
                                       XEvent z; XNextEvent(e ,&z);
                                           ++*((N=XLookupKeysym
                                             (&z.xkey,0))-IT?
                                             N-LT? UP-N?& E:&
                                             J:& u: &h); --*(
                                             DN -N? N-DT ?N==
                                             RT?&u: & W:&h:&J
                                              ); } m=15*F/l;
                                              c+=(I=M/ l,l*H
                                              +I*M+a*X)*_; H
                                              =A*r+v*X-F*l+(
                                              E=.1+X*4.9/l,t
                                              =T*m/32-I*T/24
                                               )/S; K=F*M+(
                                               h* 1e4/l-(T+
                                               E*5*T*E)/3e2
                                               )/S-X*d-B*A;
                                               a=2.63 /l*d;
                                               X+=( d*l-T/S
                                                *(.19*E +a
                                                *.64+J/1e3
                                                )-M* v +A*
                                                Z)*_; l +=
                                                K *_; W=d;
                                                sprintf(f,
                                                "%5d  %3d"
                                                "%7d",p =l
                                               /1.7,(C=9E3+
                              O*57.3)%0550,(int)i); d+=T*(.45-14/l*
                             X-a*130-J* .14)*_/125e2+F*_*v; P=(T*(47
                             *I-m* 52+E*94 *D-t*.38+u*.21*E) /1e2+W*
                             179*v)/2312; select(p=0,0,0,0,&G); v-=(
                              W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
                               )/107e2)*_; D=cos(o); E=sin(o); } }
```

Hard to Read



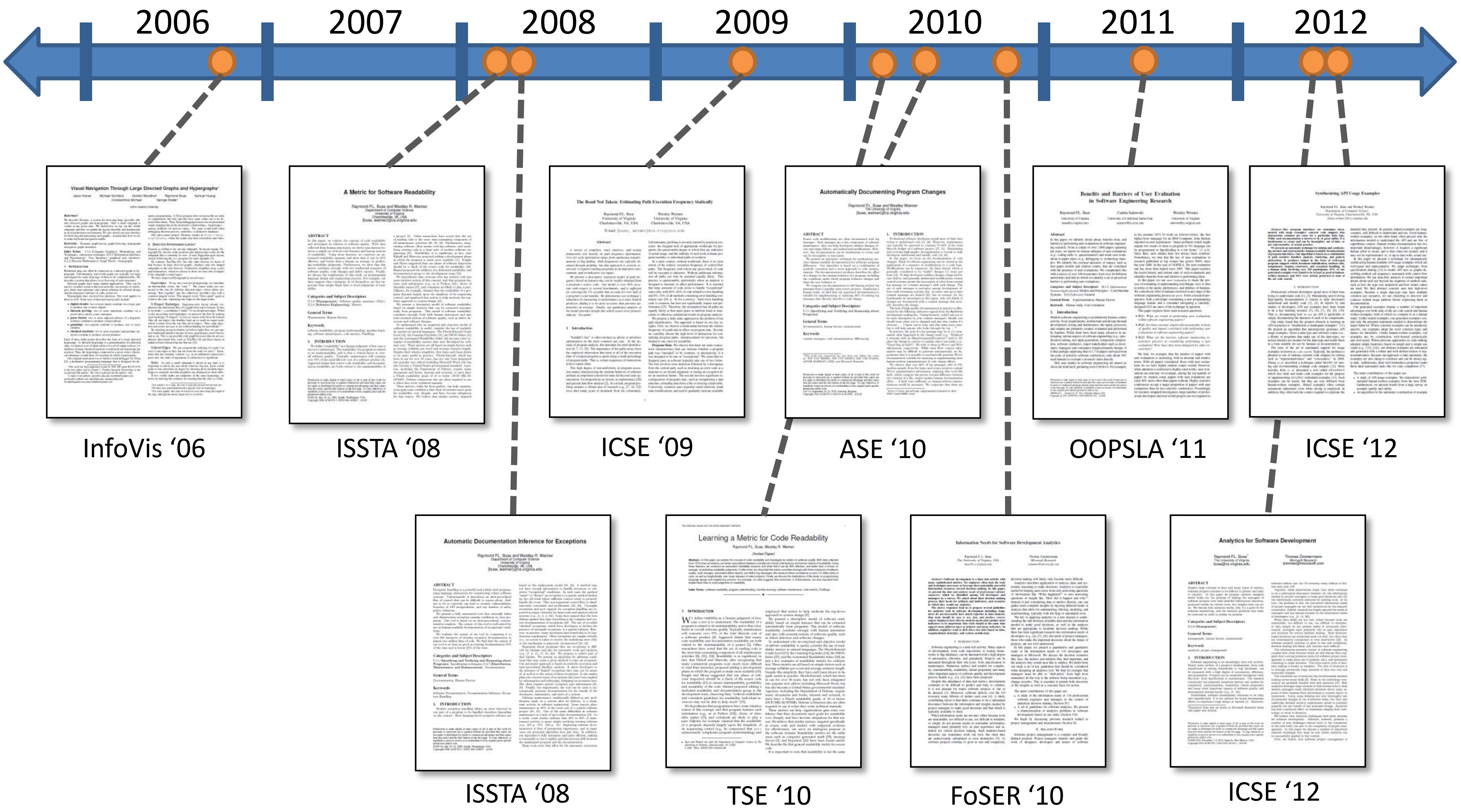
```
import java.math.BigDecimal;

class Change {
  public static v
    BigDecimal pa
    BigDecimal c
    System.out.p
  }
}
```

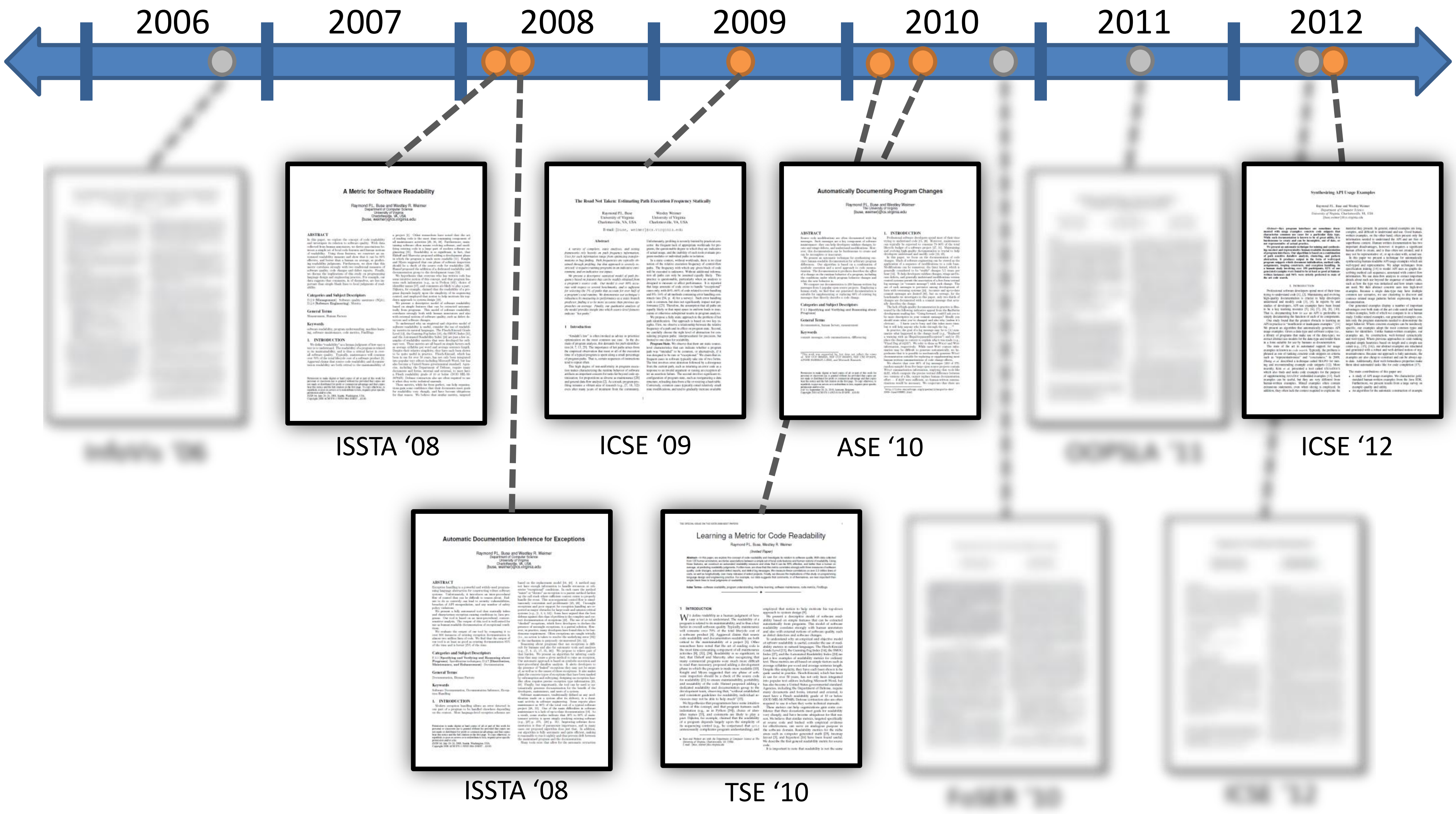
Warning : Unusual Pattern : BigDecimal(double)

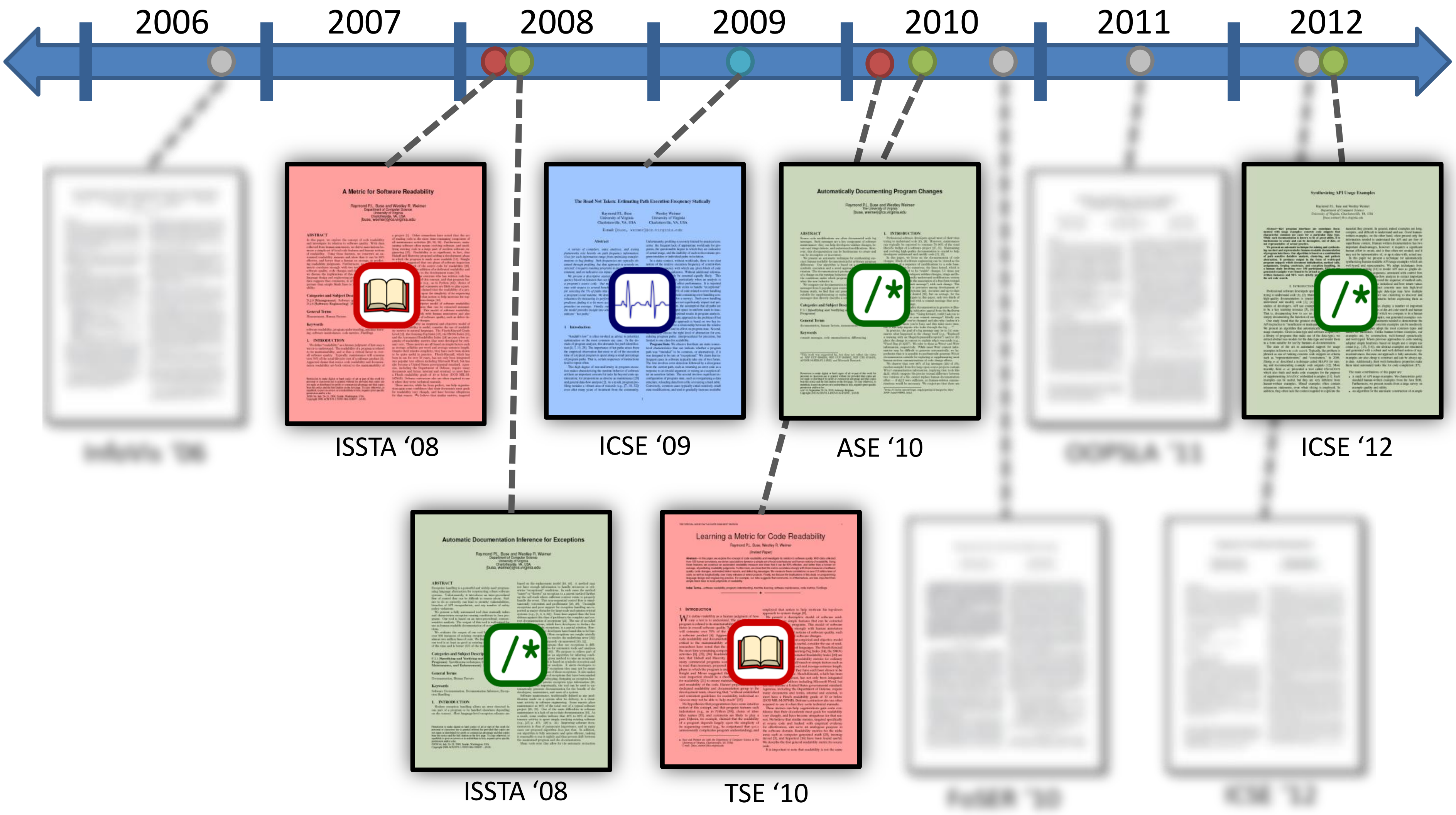
1 quick fixes available:

Change to BigDecimal(String) - 94% of cases use this

Confusing


2006
2007
2008
2009
2010
2011
2012
Visual Navigation Through Large Directed Graphs and Hypergraphs
InfoVis '06
A Metric for Software Readability
ISSTA '08
The Road Not Taken: Estimating Path Execution Frequency Statically
ICSE '09
Automatically Documenting Program Changes
ASE '10
Benefits and Barriers of User Evaluation in Software Engineering Research
OOPSLA '11
Synthesizing API Usage Examples
ICSE '12
Automatic Documentation Inference for Exceptions
ISSTA '08
Learning a Metric for Code Readability
TSE '10
Information Needs for Software Development Analytics
FoSER '10
Analytics for Software Development
ICSE '12


2006
2007
2008
2009
2010
2011
2012
A Metric for Software Readability
ISSTA '08
The Road Not Taken: Estimating Path Execution Frequency Statically
ICSE '09
Automatically Documenting Program Changes
ASE '10
Synthesizing API Usage Examples
ICSE '12
Automatic Documentation Inference for Exceptions
ISSTA '08
Learning a Metric for Code Readability
TSE '10


2006
2007
2008
2009
2010
2011
2012
A Metric for Software Readability
ISSTA '08
The Road Not Taken: Estimating Path Execution Frequency Statically
ICSE '09
Automatically Documenting Program Changes
ASE '10
Synthesizing API Usage Examples
ICSE '12
Automatic Documentation Inference for Exceptions
ISSTA '08
Learning a Metric for Code Readability
TSE '10

## Readability

- Raymond P.L. Buse and Westley Weimer. *Learning a Metric for Code Readability.* IEEE Transactions on Software Engineering, 36(4):546–558, 2010.
- Raymond P.L. Buse and Westley Weimer. *A Metric for Software Readability.* In International Symposium on Software Testing and Analysis, pages 121–130, Seattle, WA, USA, 2008. **ACM Distinguished Paper Award**

## Runtime Behavior

- Raymond P.L. Buse and Westley Weimer. *The Road Not Taken*: Estimating path execution frequency statically. In International Conference on Software Engineering, Vancouver, CA, 2009.

## Documentation

- Raymond P.L. Buse and Westley Weimer. *Synthesizing API Usage Examples*. In International Conference on Software Engineering [To Appear], Zurich, Switzerland, 2012.
- Raymond P.L. Buse and Westley Weimer. *Automatically Documenting Program Changes*. In International Conference on Automated Software Engineering, Antwerp, Belgium, 2010.
- Raymond P.L. Buse and Westley Weimer. *Automatic Documentation Inference for Exceptions*. In International Symposium on Software Testing and Analysis, Seattle, WA, USA, 2008.

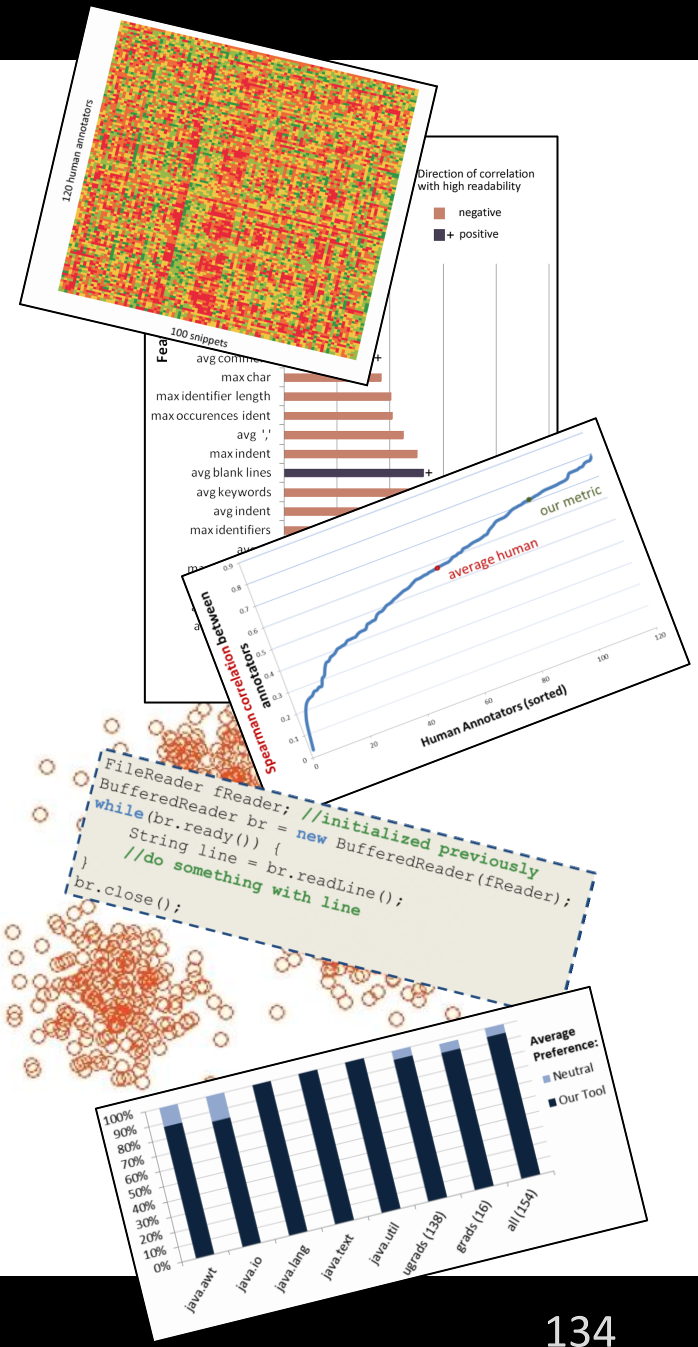
**http://arrestedcomputing.com**

## Thank You!

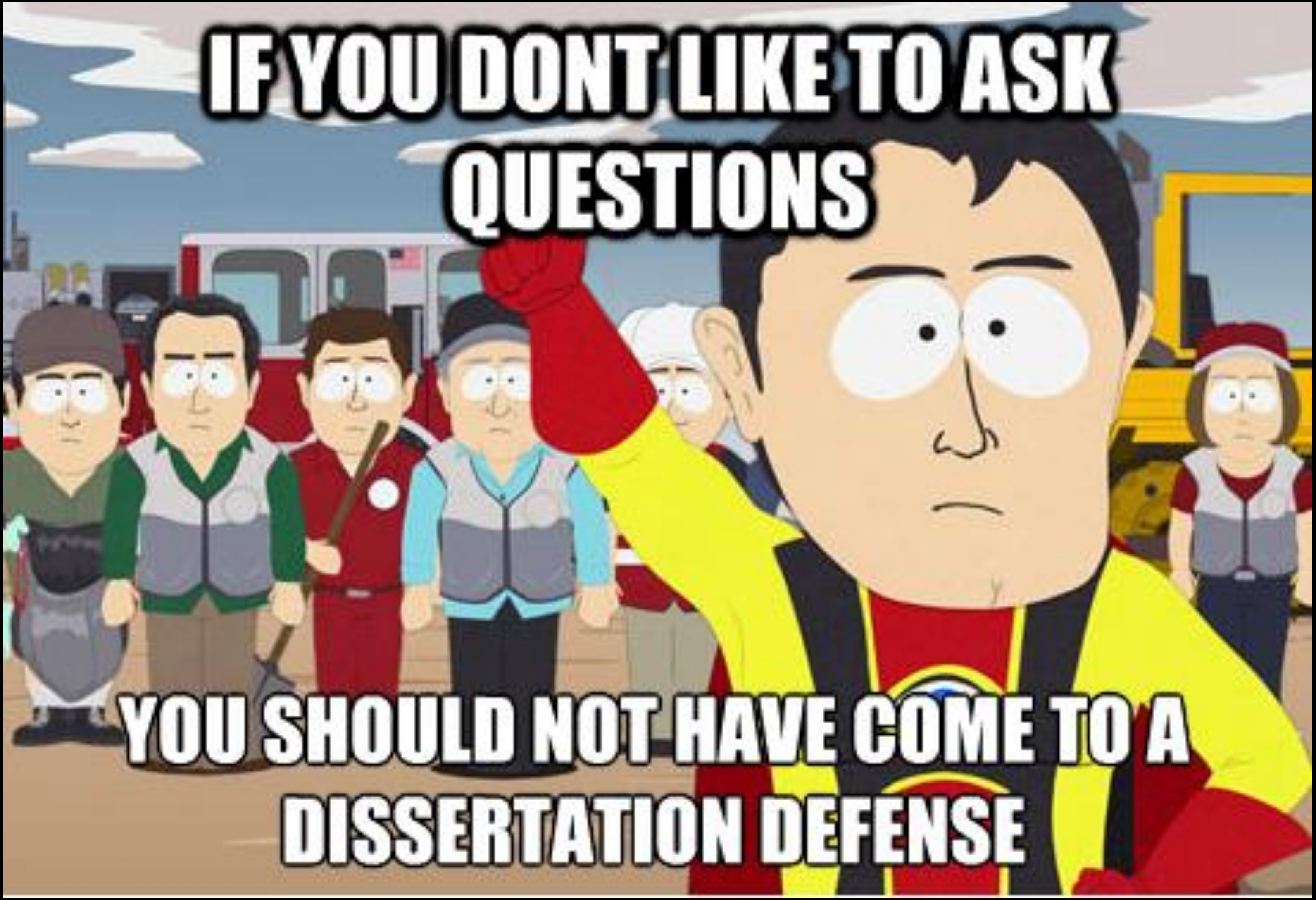
# Questions?

Also ask me about:

- Documenting Exceptions
- Generating Commit Messages
- Conducting Human Studies



```
FileReader fReader; //initialized previously
BufferedReader br = new BufferedReader(fReader);
while(br.ready()) {
  String line = br.readLine();
  //do something with line
}
br.close();
```


IF YOU DONT LIKE TO ASK QUESTIONS
YOU SHOULD NOT HAVE COME TO A DISSERTATION DEFENSE