

Confronting Complexity

We have met the enemy and he is us

Robert Smallshire

 @robsmallshire

1

Characterising Complexity

What is complexity and how can we measure it?

2

Evolution of Entropy

Can we get out of the game?

3

Predicting the Past

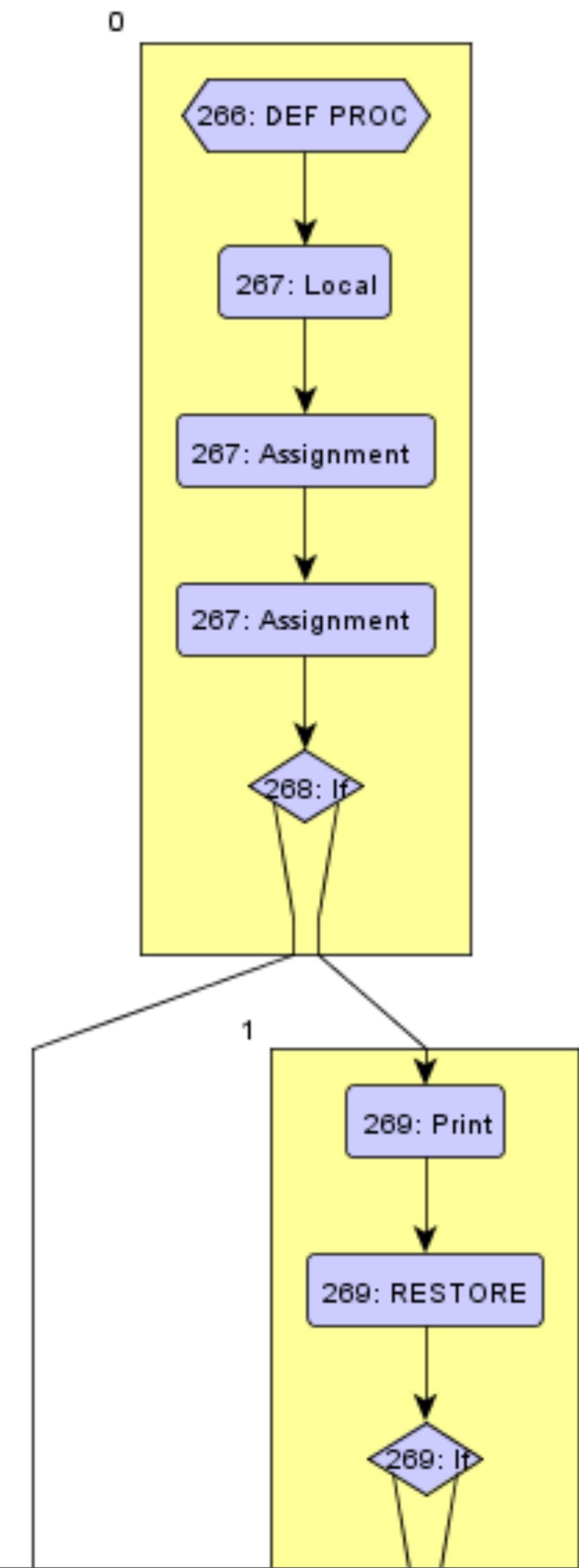
A model driven approach.



**“Measuring programming progress by
lines of code is like measuring aircraft
building progress by weight”**

Bill Gates





Cyclomatic Complexity

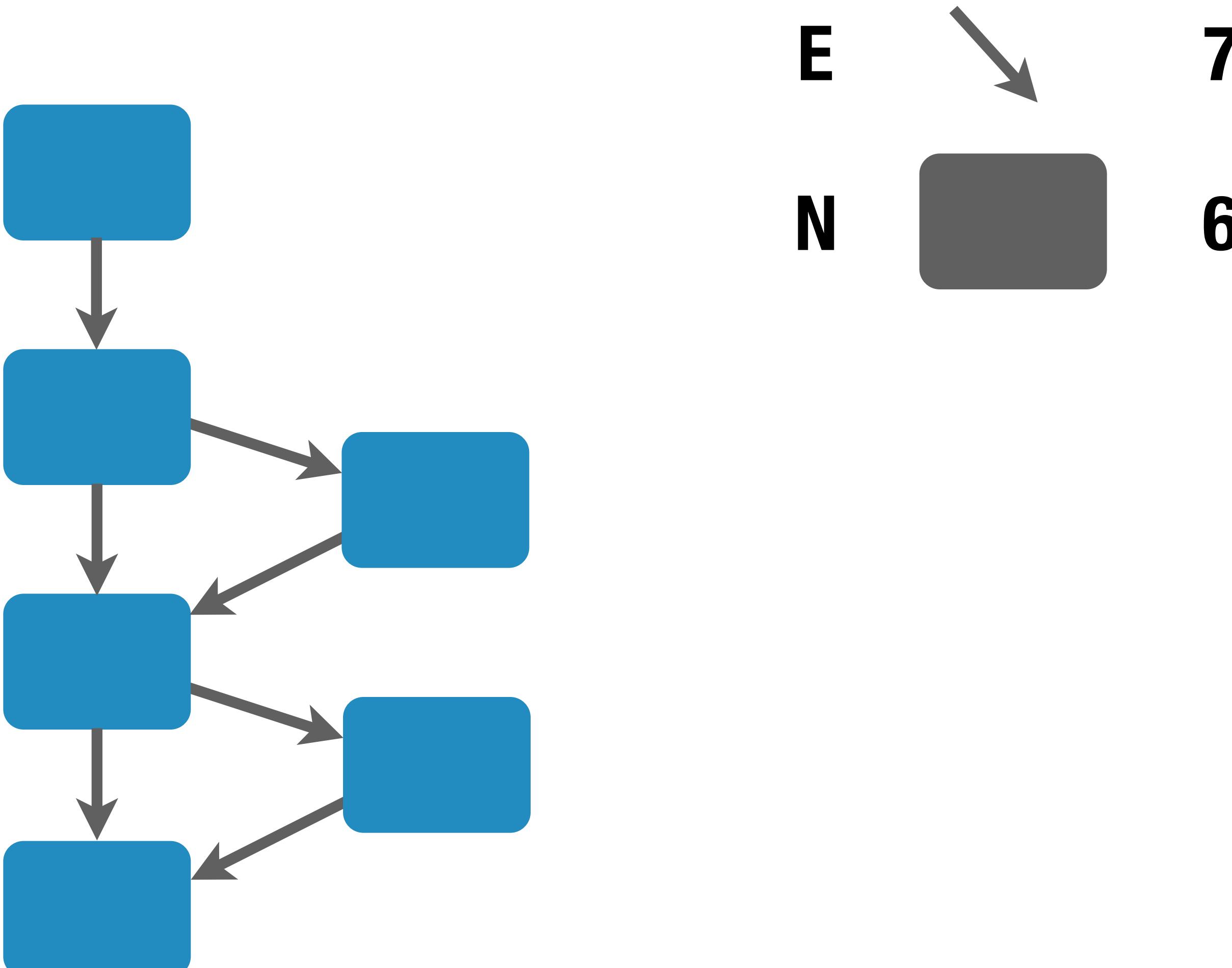
Thomas McCabe 1976

```
def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    print(msg or num)
```

Cyclomatic Complexity

Thomas McCabe 1976

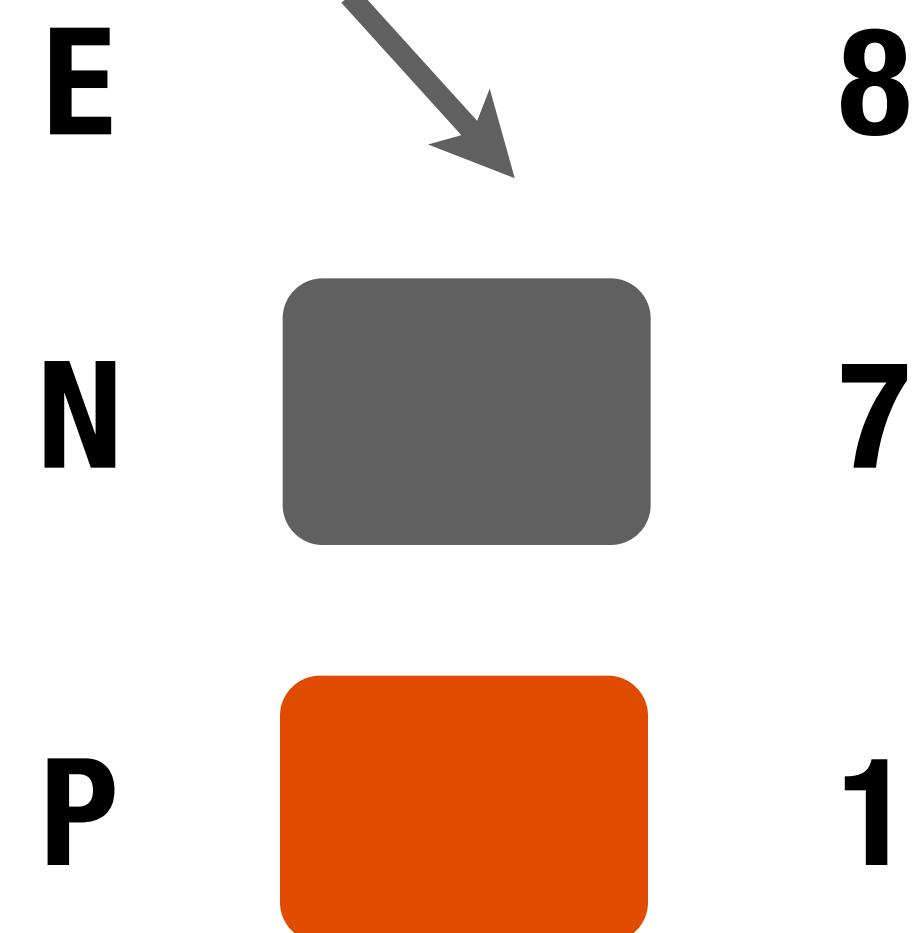
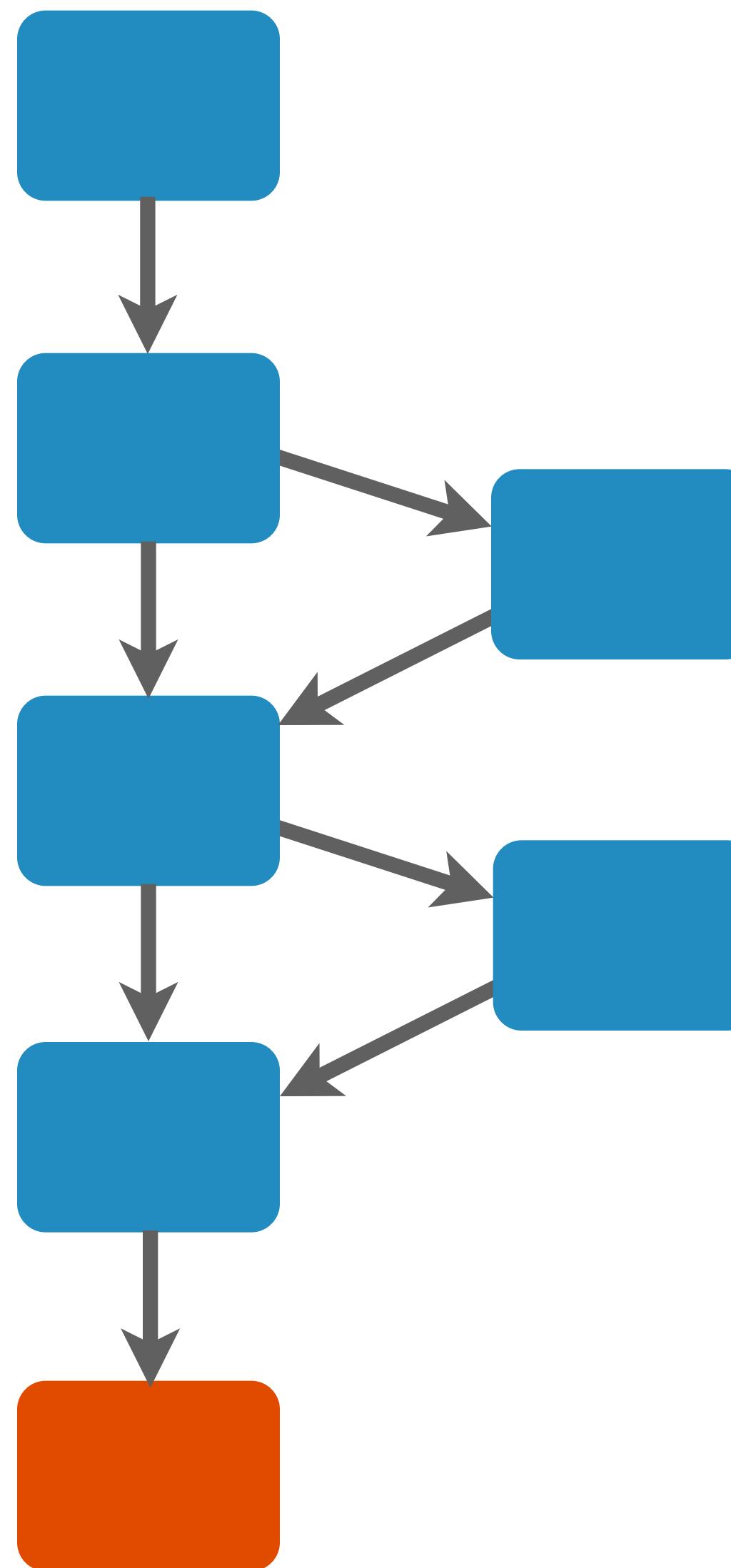
```
def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    print(msg or num)
```



Cyclomatic Complexity

Thomas McCabe 1976

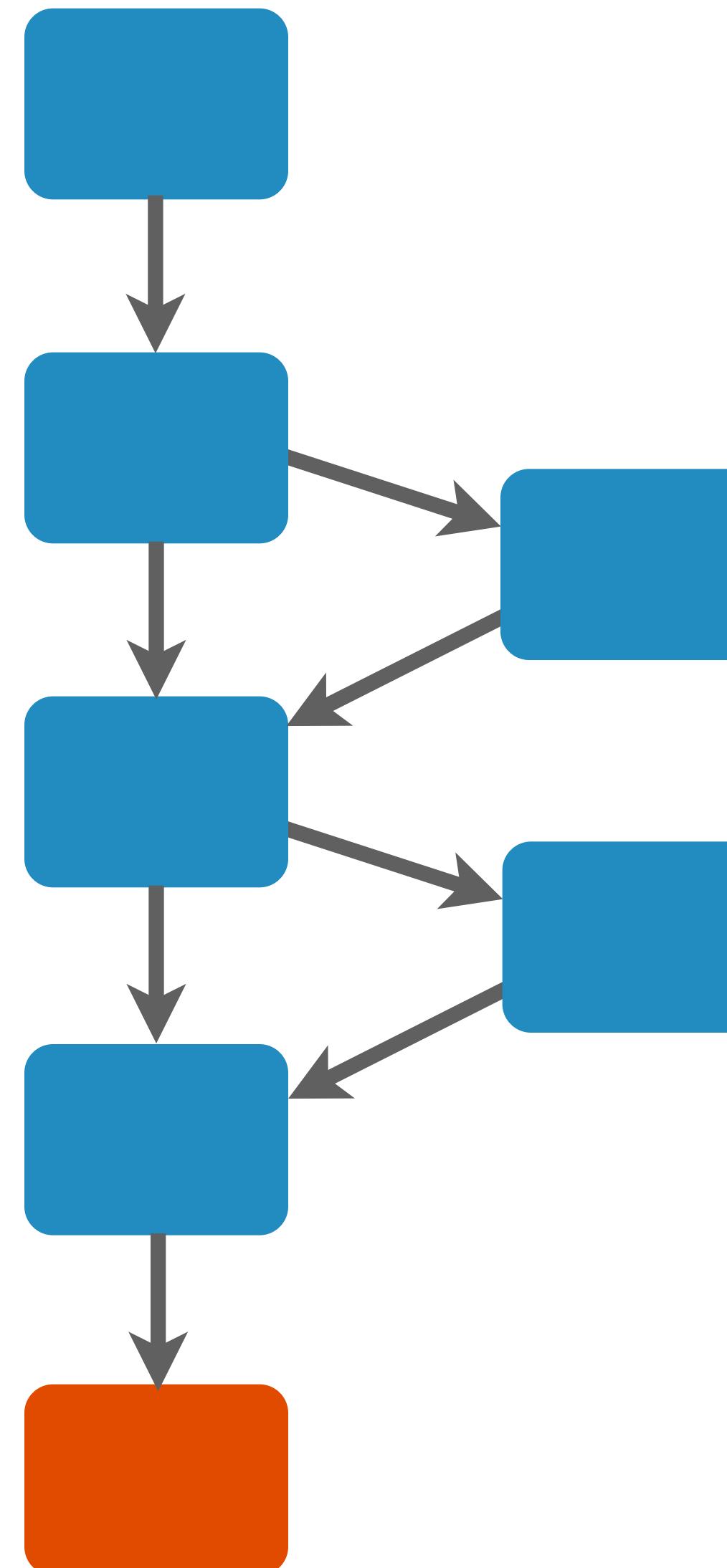
```
def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    print(msg or num)
return
```



Cyclomatic Complexity

Thomas McCabe 1976

```
def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    print(msg or num)
return
```



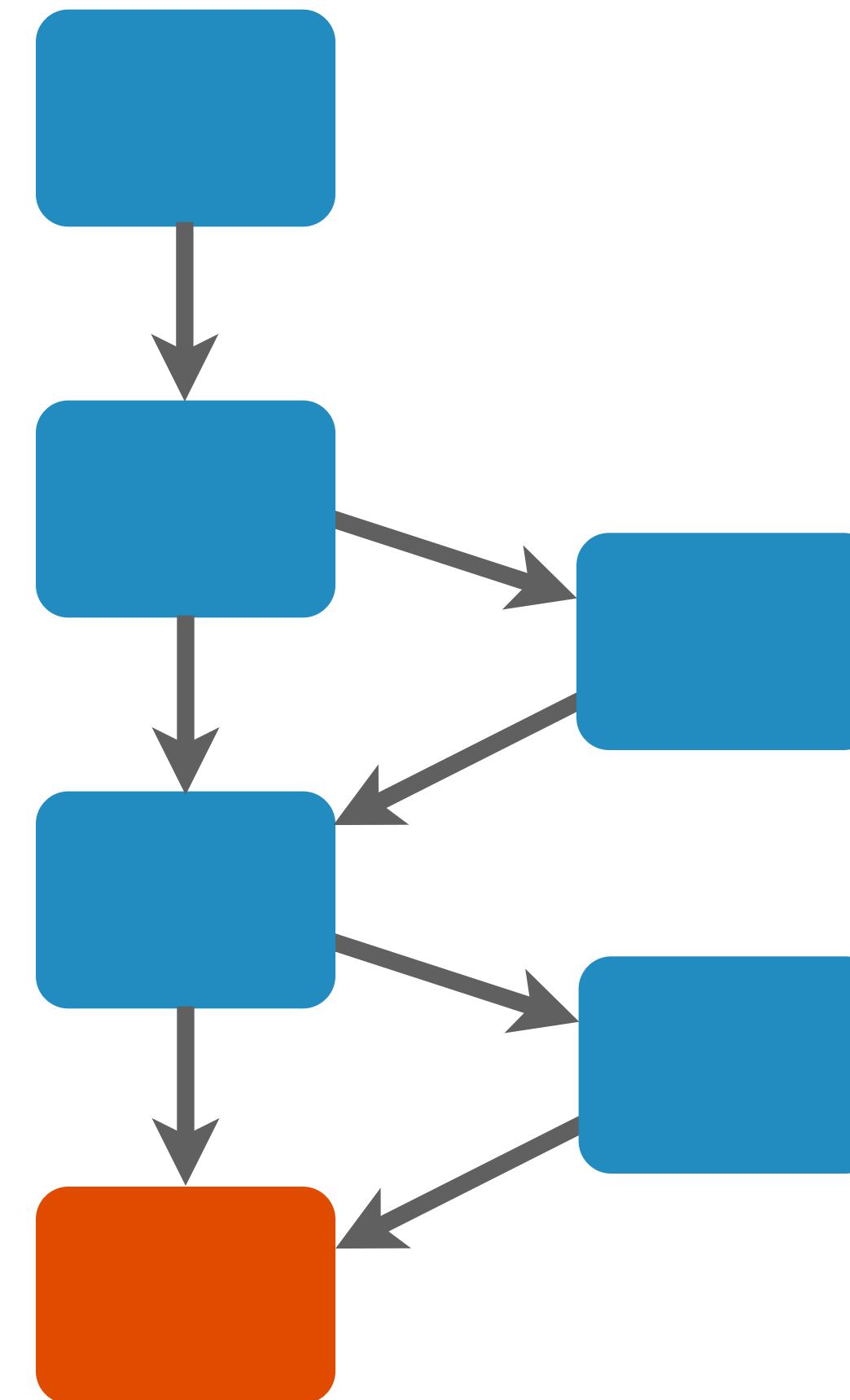
E	8
N	7
P	1

$$M = E - N + 2P$$
$$M = 8 - 7 + 2 \times 1$$
$$M = 3$$

Cyclomatic Complexity

Thomas McCabe 1976

```
def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    return msg or num
```



$$M = 3$$

E	7
N	6
P	1

$$M = E - N + 2P$$

$$M = 7 - 6 + 2 \times 1$$

Cyclomatic Complexity

Thomas McCabe 1976

```
class TestFizzBuzz(unittest.TestCase):

def fizzbuzz(num):
    msg = ''
    if num % 3 == 0:
        msg += 'Fizz'
    if num % 5 == 0:
        msg += 'Buzz'
    return msg or num

    def test_multiple_of_three(self):
        self.assertEqual(fizzbuzz(6), 'Fizz')

    def test_multiple_of_five(self):
        self.assertEqual(fizzbuzz(10), 'Buzz')

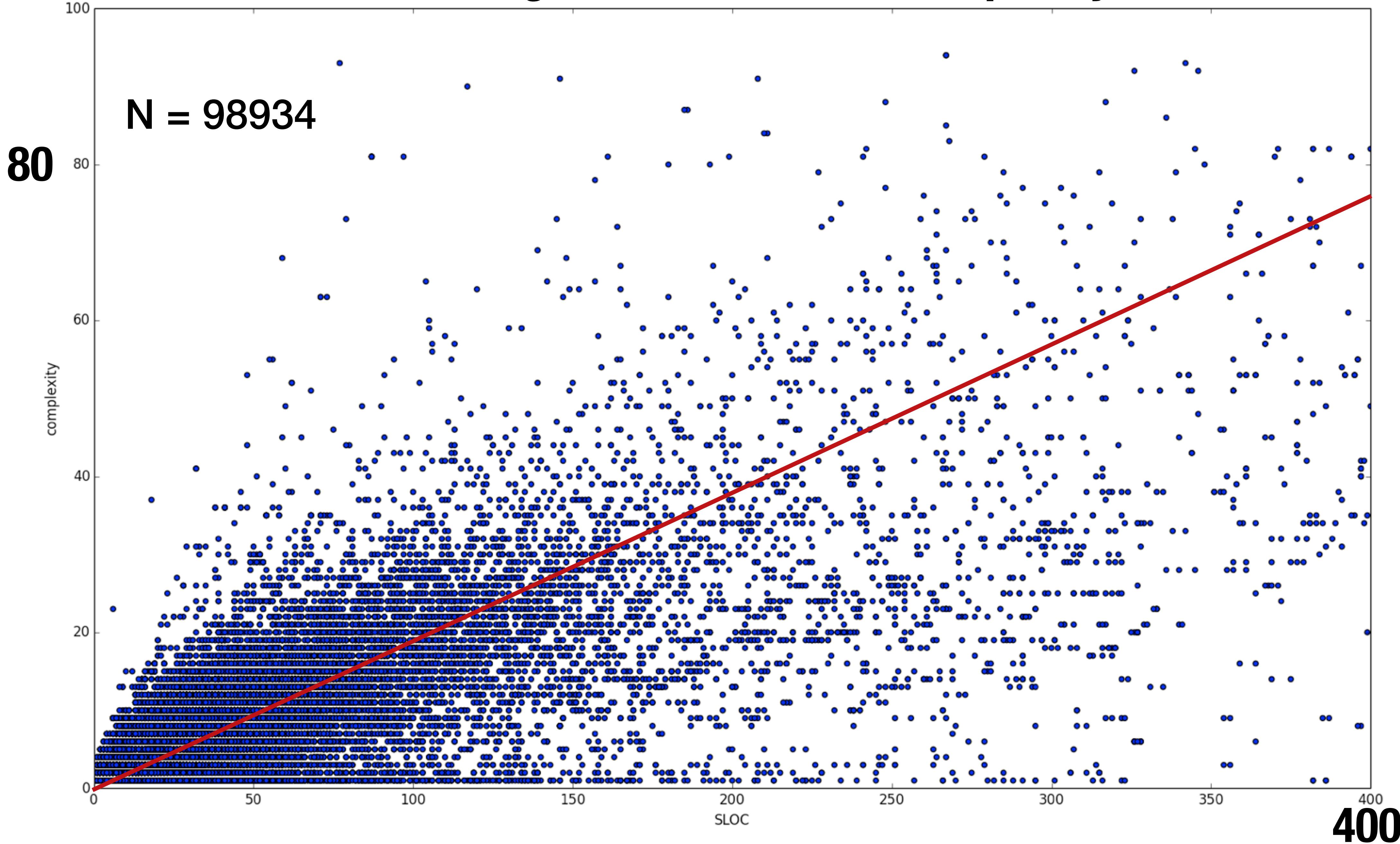
    def test_multiple_of_three_and_five():
        self.assertEqual(fizzbuzz(30), 'FizzBuzz')

    def test_multiple_of_neither_three_nor_five():
        self.assertEqual(fizzbuzz(17), 17)
```

M = 3

lower bound on number of tests

Function length versus function complexity

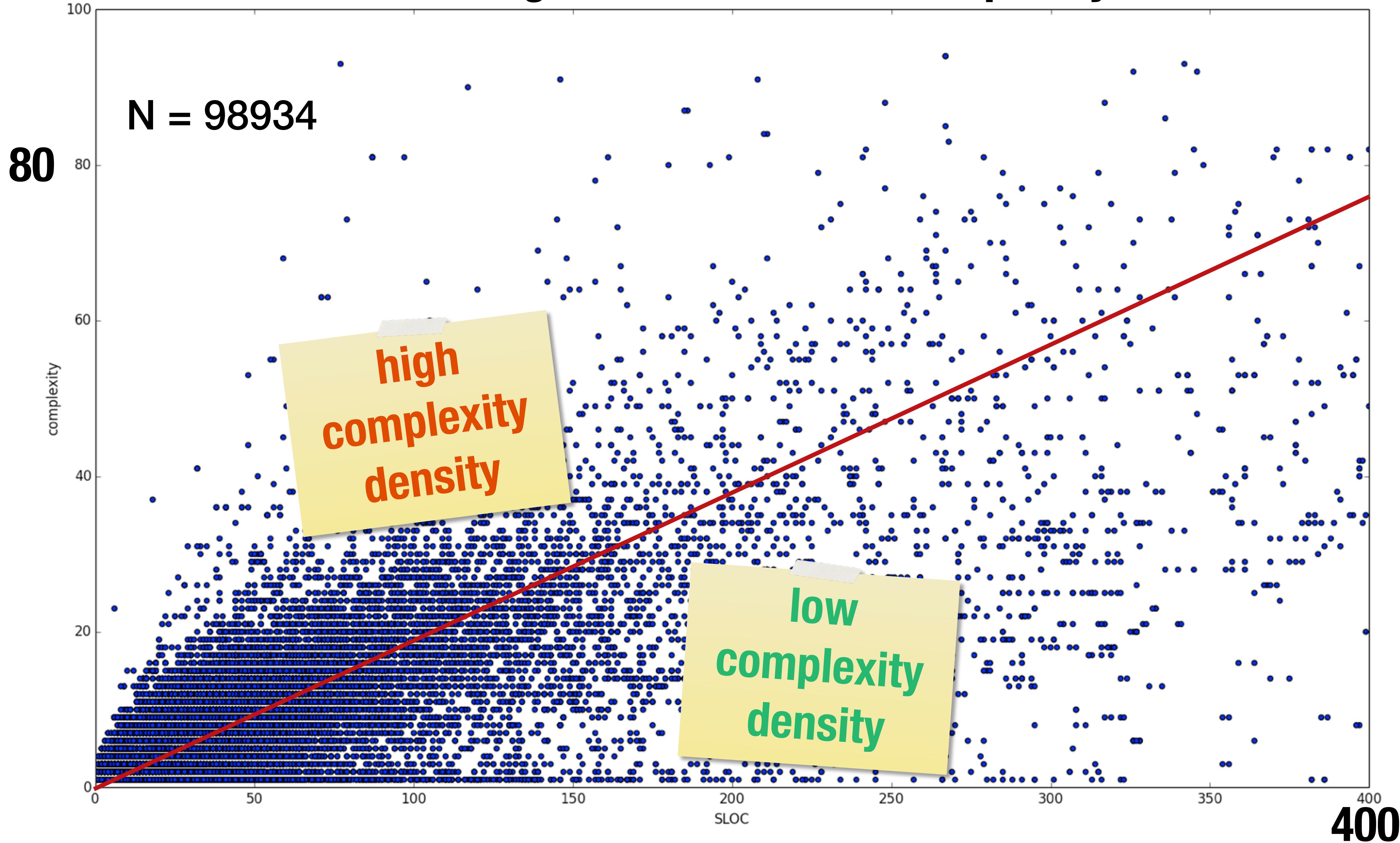


Every fifth line of code is a conditional

so

Each additional five lines of code demands another test

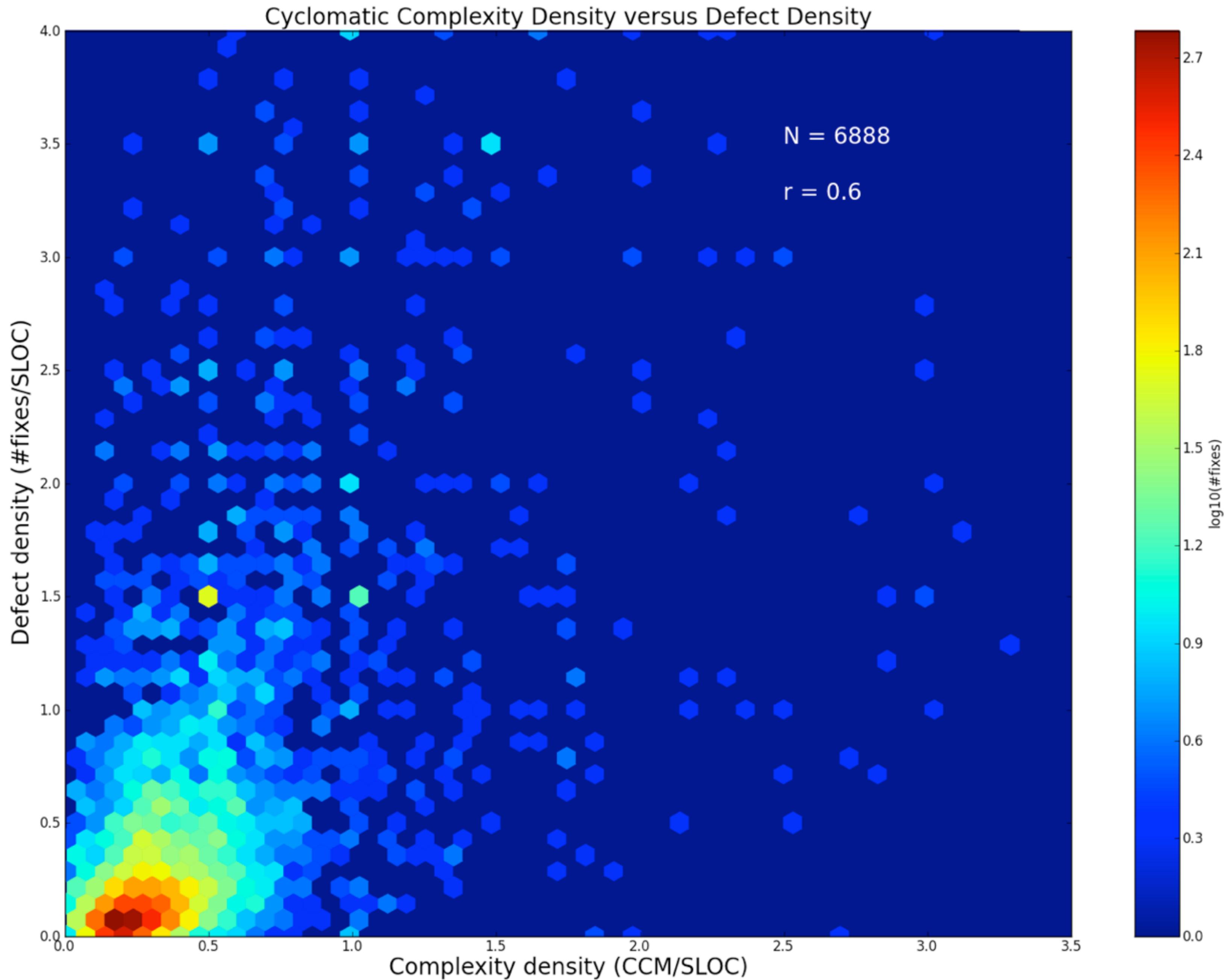
Function length versus function complexity

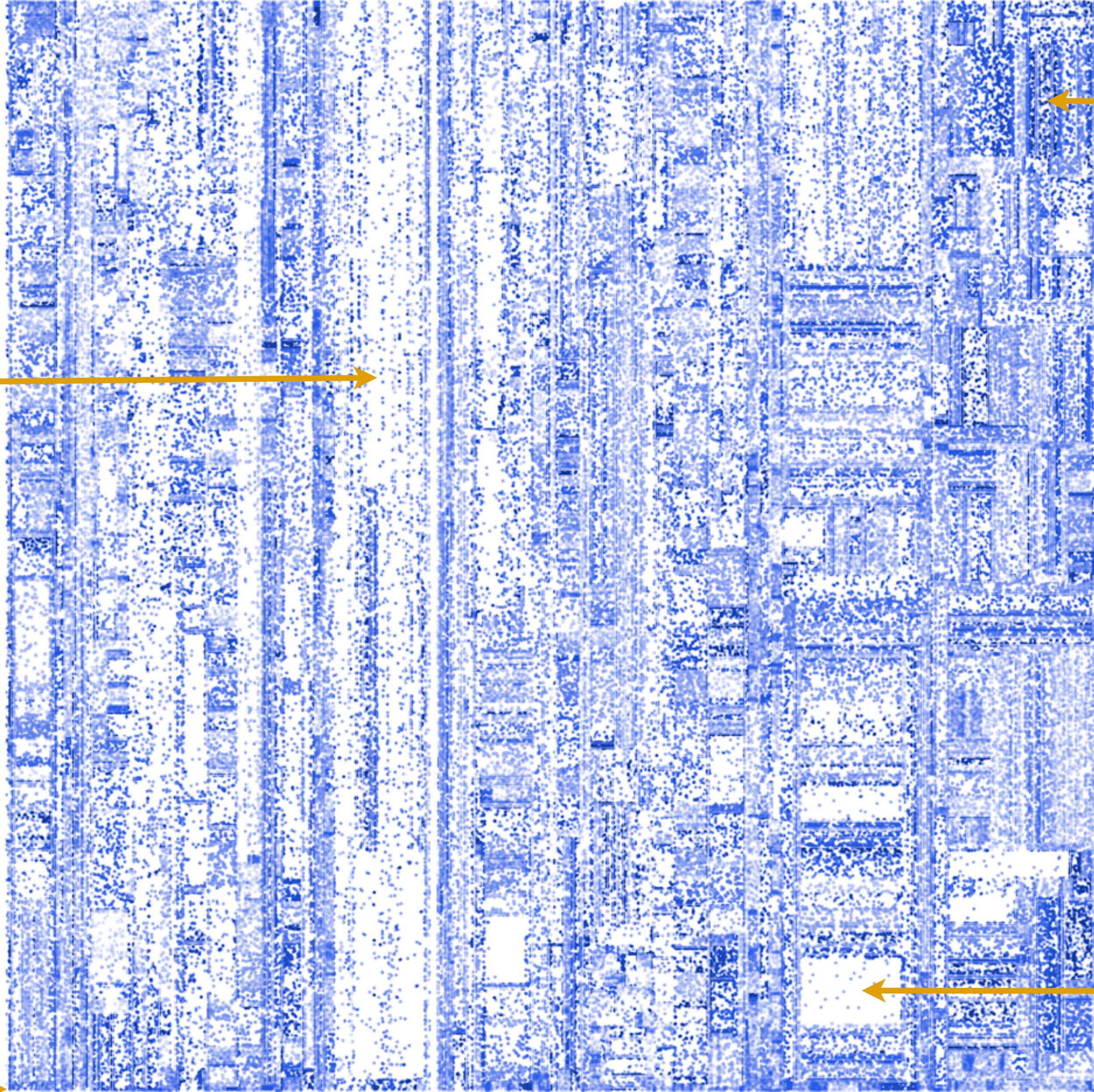


- ▶ Complexity density can be aggregated
- ▶ Mapping!



- ▶ defect density correlates with complexity density
- ▶ strong signal at CCM=0.20 to 0.25





**code predates
defect tracking**

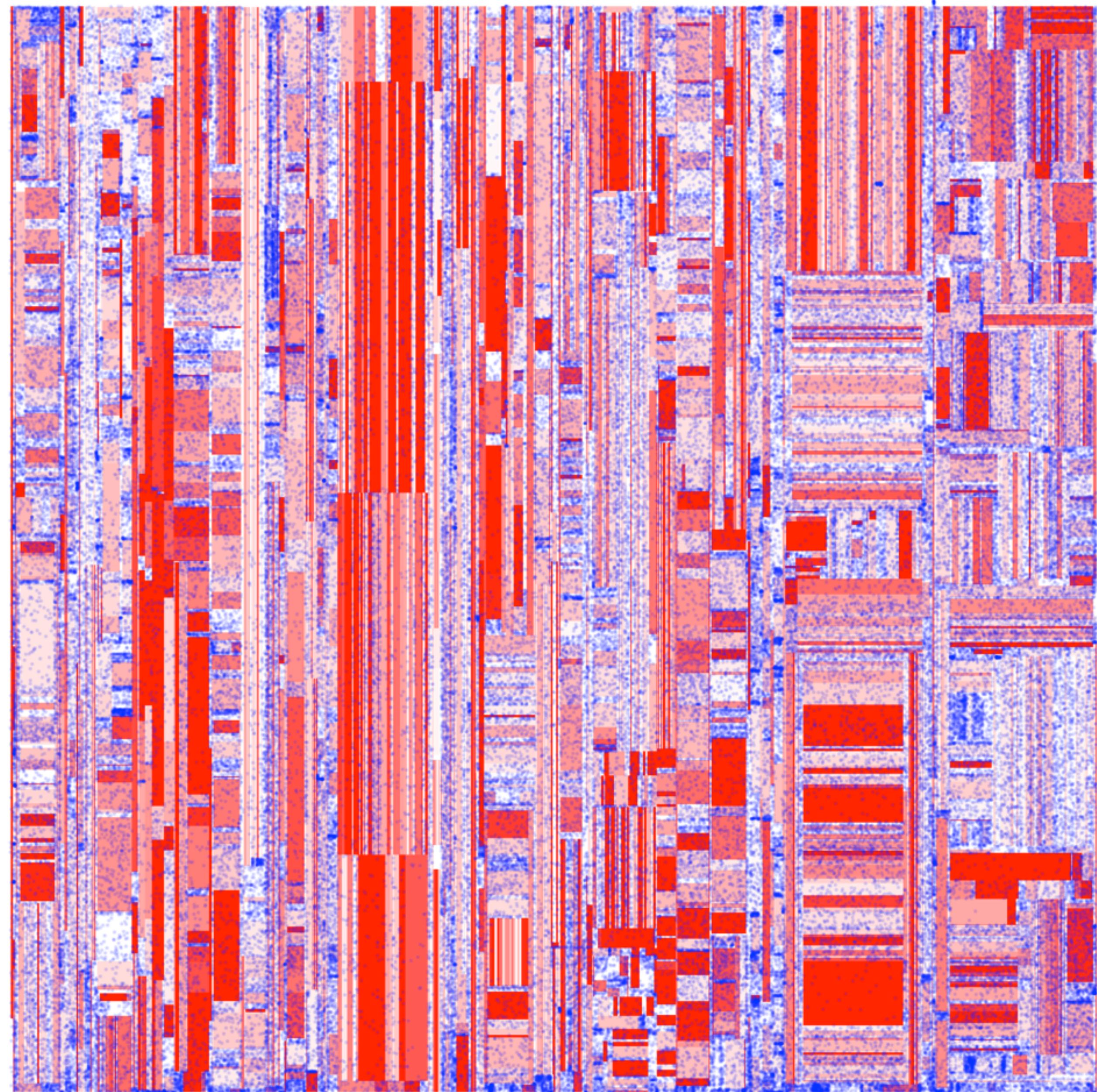
**many fixes in
presentation
layer**

**maintenance
hotspot**

**brilliant
or unused?**



- ▶ Correlations may not be what you expect!
- ▶ Much of the very complex code here is *older* than the defect database
- ▶ Low defect density may indicate unused code





Robert Smallshire

@robsmallshire

Attention person on train! I can see from 4
m away your functions are too long and your
code has too many levels of indentation!
Refactor!



Reply



Delete



Favorite



More

64

RETWEETS

23

FAVORITES



8:43 AM - 28 Feb 13

```
Protected Shared Function dateFromJavaXml(ByVal dt As String) As DateTime
    Dim datetimel As DateTime = DateTime.get_Now()
    If (dt.EndsWith(" GMT-12:00")) Then
        dt = dt.Replace(" GMT-12:00", "-12:00")
    ElseIf (dt.EndsWith(" GMT-11:00")) Then
        dt = dt.Replace(" GMT-11:00", "-11:00")
        GoTo label1
    If (dt.EndsWith(" WSST")) Then
        dt = dt.Replace(" WSST", "-11:00")
    ElseIf (dt.EndsWith(" SDT")) Then
        dt = dt.Replace(" SDT", "-11:00")
        GoTo label2
    If (dt.EndsWith(" NUST")) Then
        dt = dt.Replace(" NUST", "-11:00")
    ElseIf (dt.EndsWith(" GMT-10:00")) Then
        dt = dt.Replace(" GMT-10:00", "-10:00")
        GoTo label3
    If (dt.EndsWith(" HADT")) Then
        dt = dt.Replace(" HADT", "-10:00")
    ElseIf (dt.EndsWith(" HDT")) Then
        dt = dt.Replace(" HDT", "-10:00")
        GoTo label4
    If (dt.EndsWith(" TKST")) Then
        dt = dt.Replace(" TKST", "-10:00")
    ElseIf (dt.EndsWith(" CKHST")) Then
        dt = dt.Replace(" CKHST", "-10:00")
        GoTo label5
    If (dt.EndsWith(" TAHST")) Then
        dt = dt.Replace(" TAHST", "-10:00")
    ElseIf (dt.EndsWith(" MARST")) Then
        dt = dt.Replace(" MARST", "-09:00")
        GoTo label6
    If (dt.EndsWith(" AKDT")) Then
        dt = dt.Replace(" AKDT", "-09:00")
    ElseIf (dt.EndsWith(" GMT-09:00")) Then
        dt = dt.Replace(" GMT-09:00", "-09:00")
        GoTo label7
    If (dt.EndsWith(" GAMST")) Then
        dt = dt.Replace(" GAMST", "-09:00")
    ElseIf (dt.EndsWith(" PDT")) Then
        dt = dt.Replace(" PDT", "-08:00")
```

357
spaces
89¼
indents

```
static int tree_key_search(
    size_t *at_pos, git_vector *entries, const char *filename, size_t filename_len)
{
    struct tree_key_search ksearch;
    const git_tree_entry *entry;
    size_t homing, i;

    ksearch.filename = filename;
    ksearch.filename_len = filename_len;

    /* Initial homing search; find an entry on the tree with
     * the same prefix as the filename we're looking for */
    if (git_vector_bsearch2(&homing, entries, &homing_search_cmp, &ksearch) < 0)
        return GIT_ENOTFOUND; /* just a signal error; not passed back to user */

    /* We found a common prefix. Look forward as long as
     * there are entries that share the common prefix */
    for (i = homing; i < entries->length; ++i) {
        entry = entries->contents[i];

        if (homing_search_cmp(&ksearch, entry) < 0)
            break;

        if (entry->filename_len == filename_len &&
            memcmp(filename, entry->filename, filename_len) == 0) {
            if (at_pos)
                *at_pos = i;
        }
    }

    /* If we haven't found our filename yet, look backwards
     * too as long as we have entries with the same prefix */
    if (homing > 0) {
        i = homing - 1;

        do {
            entry = entries->contents[i];

            if (homing_search_cmp(&ksearch, entry) > 0)
                break;

            if (entry->filename_len == filename_len &&
                memcmp(filename, entry->filename, filename_len) == 0) {
                if (at_pos)
                    *at_pos = i;
            }
        } while (i-- > 0);
    }

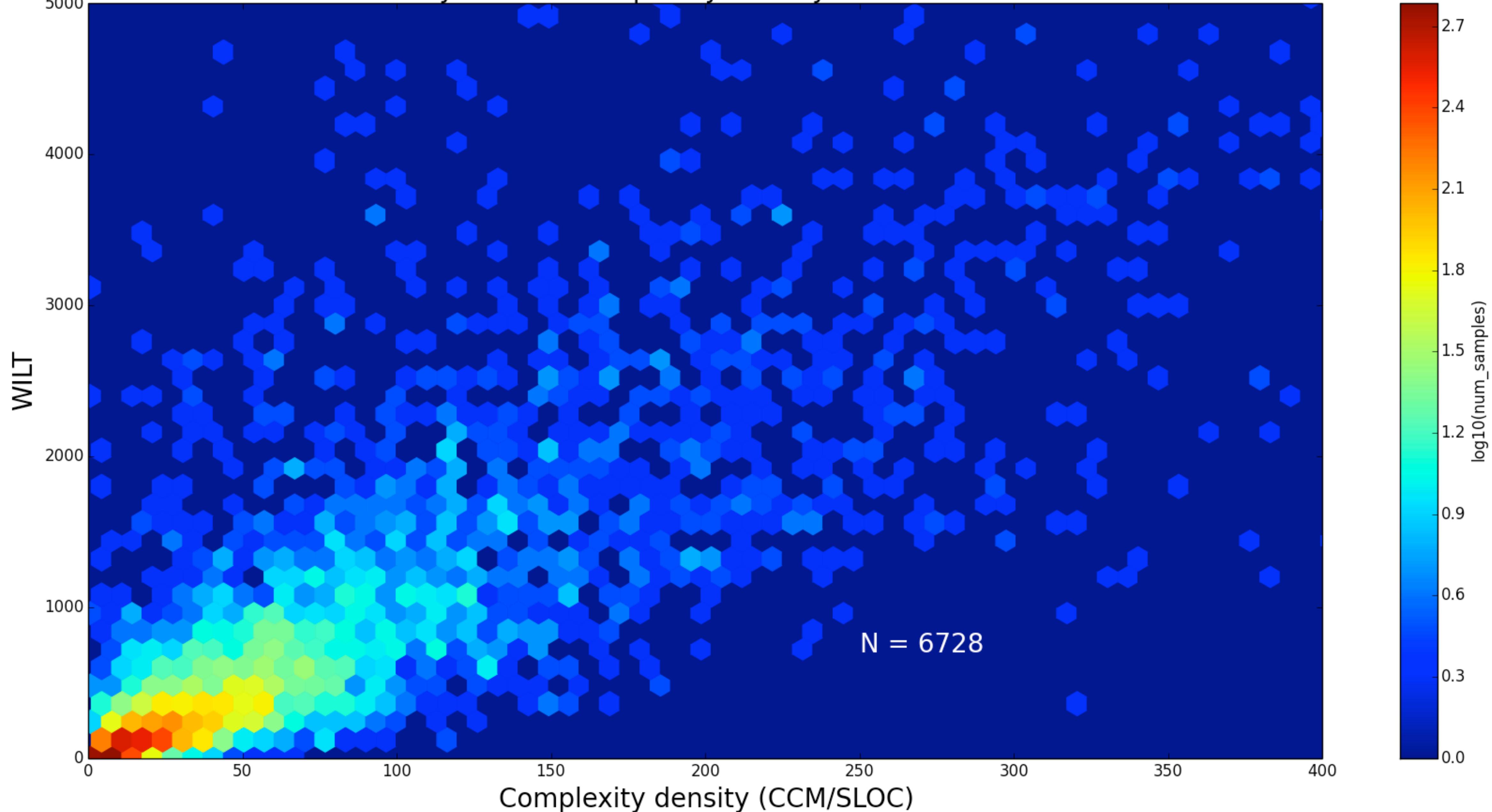
    /* The filename doesn't exist at all */
    return GIT_ENOTFOUND;
}
```

Whitespace
Integrated over
Lines of
Text

W I L T



Cyclomatic Complexity Density versus WILT

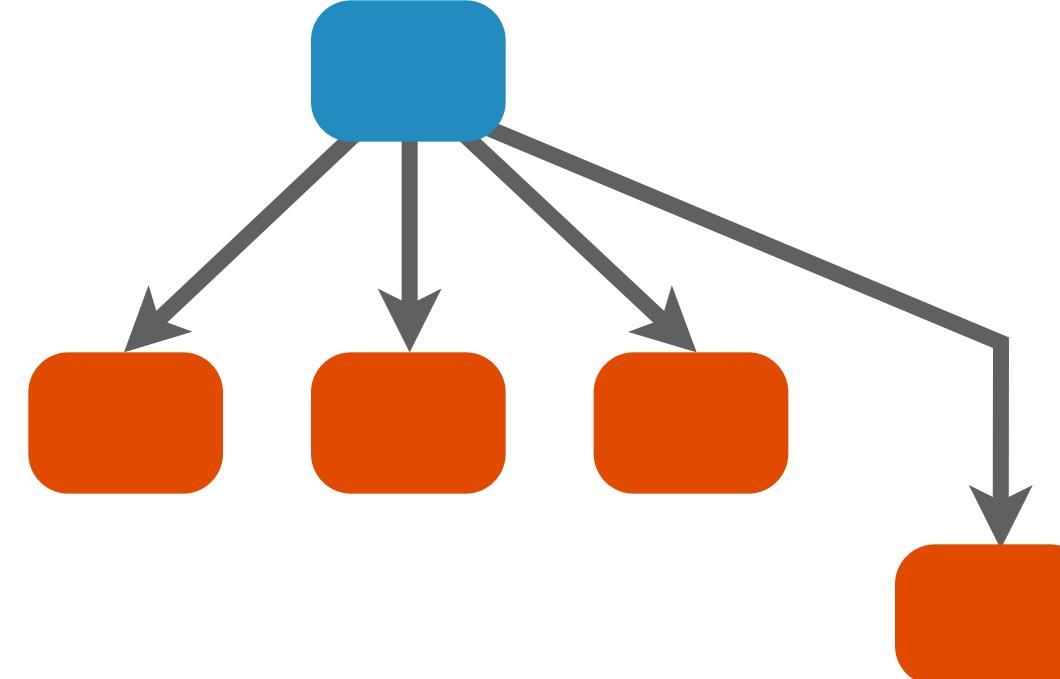


```
7  for (var i = 0; i < arr.length; i++) {  
8    if (arr[i] == 'x') {  
9      k++;  
0      var k = 0;  
1      for (var i = 0; i < arr.length; i++) {  
2        if (arr[i] == 'x') {  
3          k++;  
4          var k = 0;  
5          for (var i = 0; i < arr.length; i++) {  
6            if (arr[i] == 'x') {  
7              k++;  
8              var k = 0;  
9              for (var i = 0; i < arr.length; i++) {  
0                if (arr[i] == 'x') {  
1                  k++;  
2                  }  
3                }  
4              }  
5            }  
6          }  
7        }  
8      }  
9    }
```

Refactorings reduce complexity

e.g. replace conditional with polymorphism

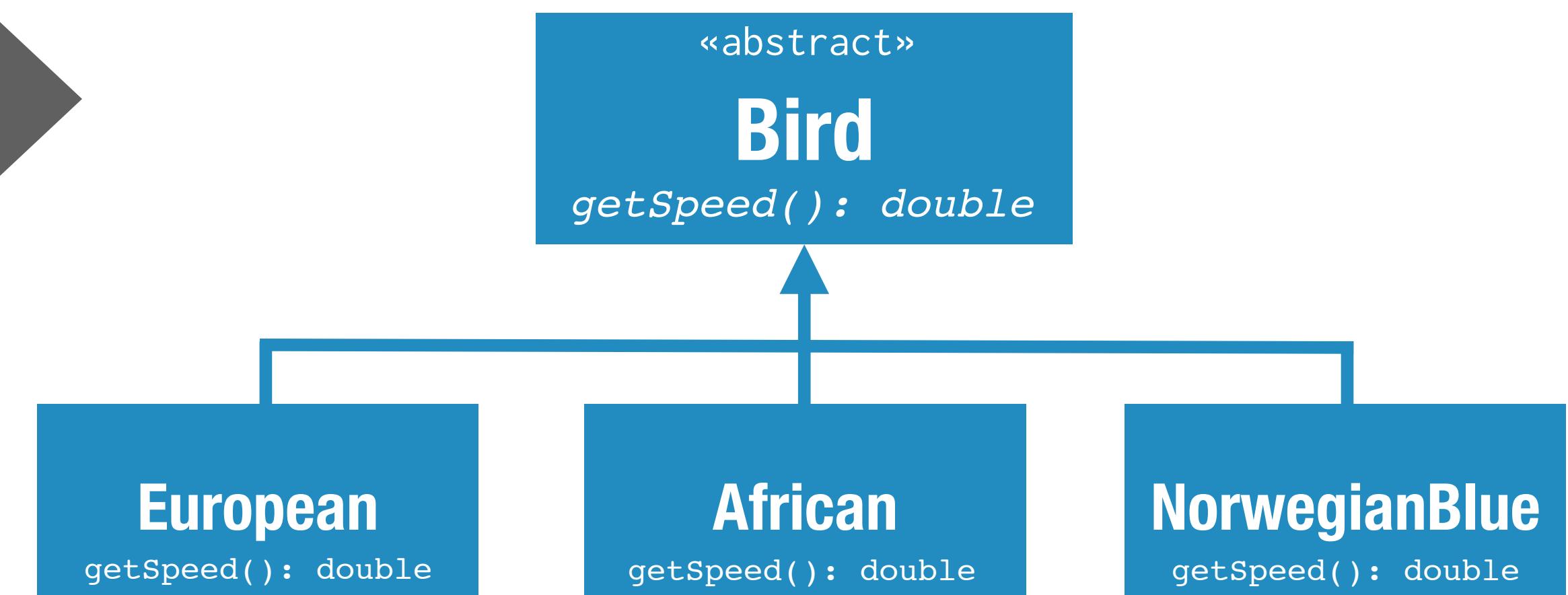
```
double getSpeed() {  
    switch (_type) {  
        case EUROPEAN:  
            return getBaseSpeed();  
        case AFRICAN:  
            return getBaseSpeed() - getLoadFactor() * _numberOfCoconuts;  
        case NORWEGIAN_BLUE:  
            return (_isExParrot) ? 0 : getBaseSpeed(_voltage);  
    }  
    throw new RuntimeException("Should be unreachable");  
}
```



$$M = E - N + 2P$$

$$M = 4 - 5 + 2 \times 4$$

$$M = 7$$



$$M = 1$$

$$M = 1$$

$$M = 1$$

$$\Sigma M = 3$$

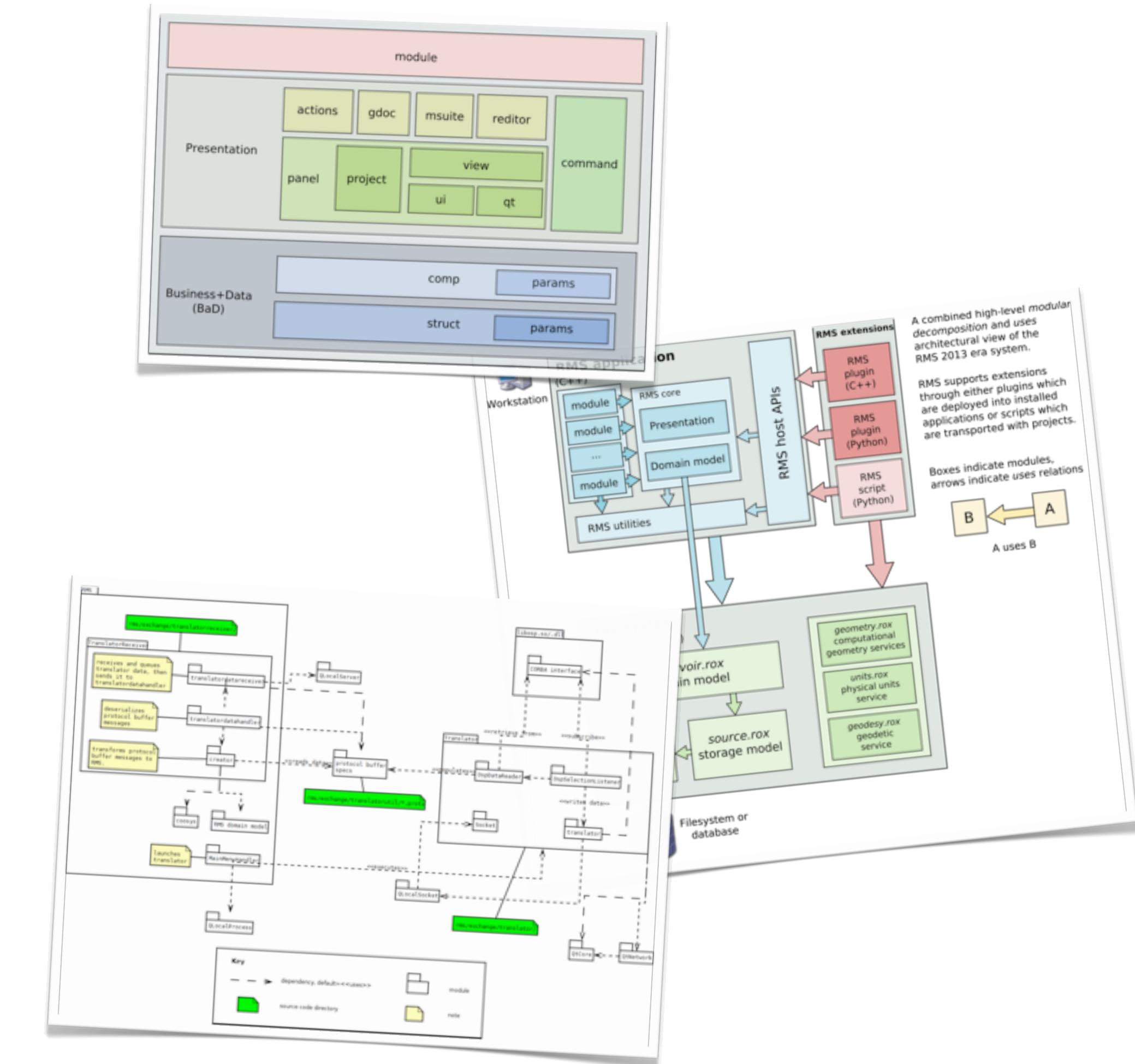


“How many if statements does it take to add a feature?”

Rob Galanakis, Technical Director, Eve Online

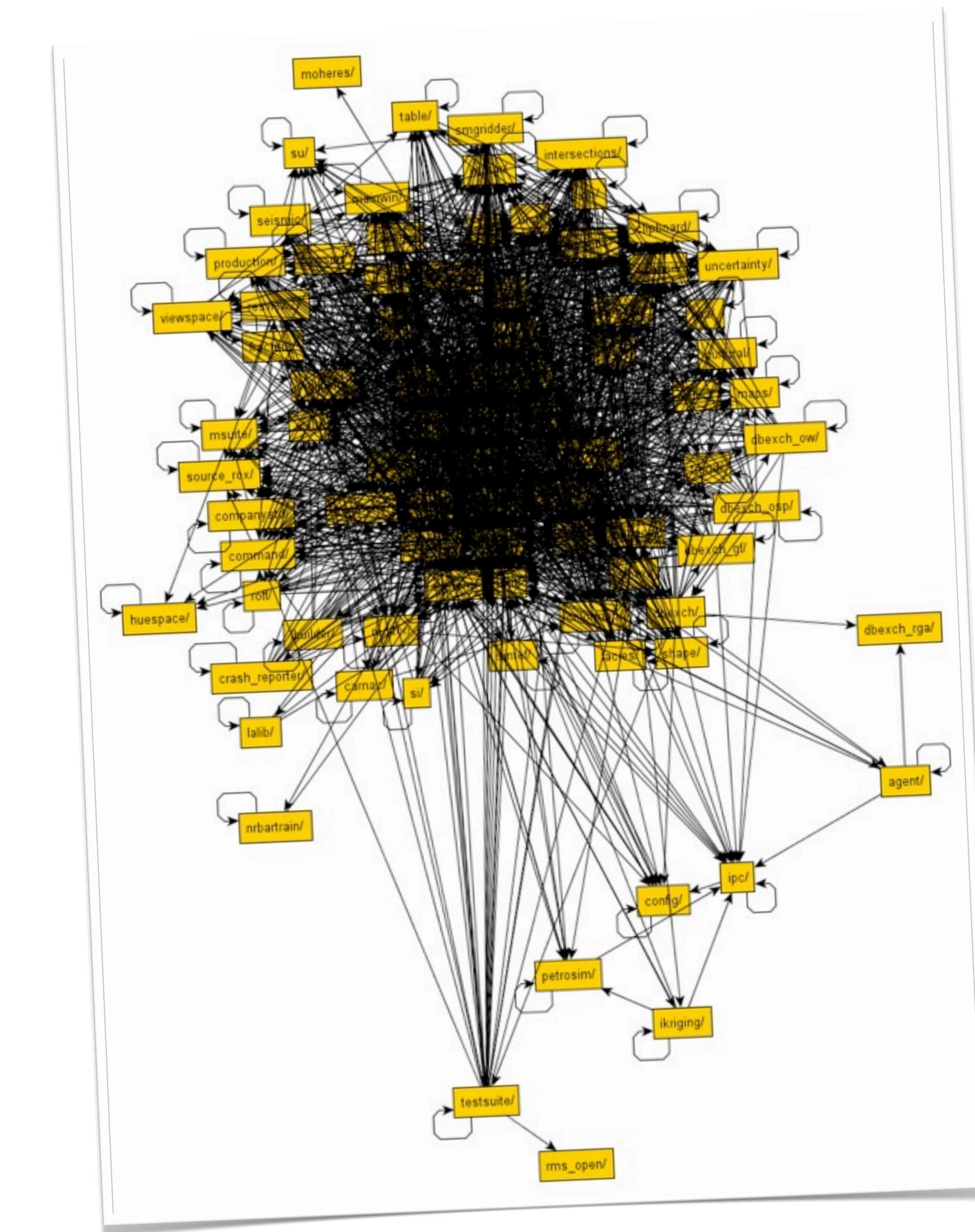
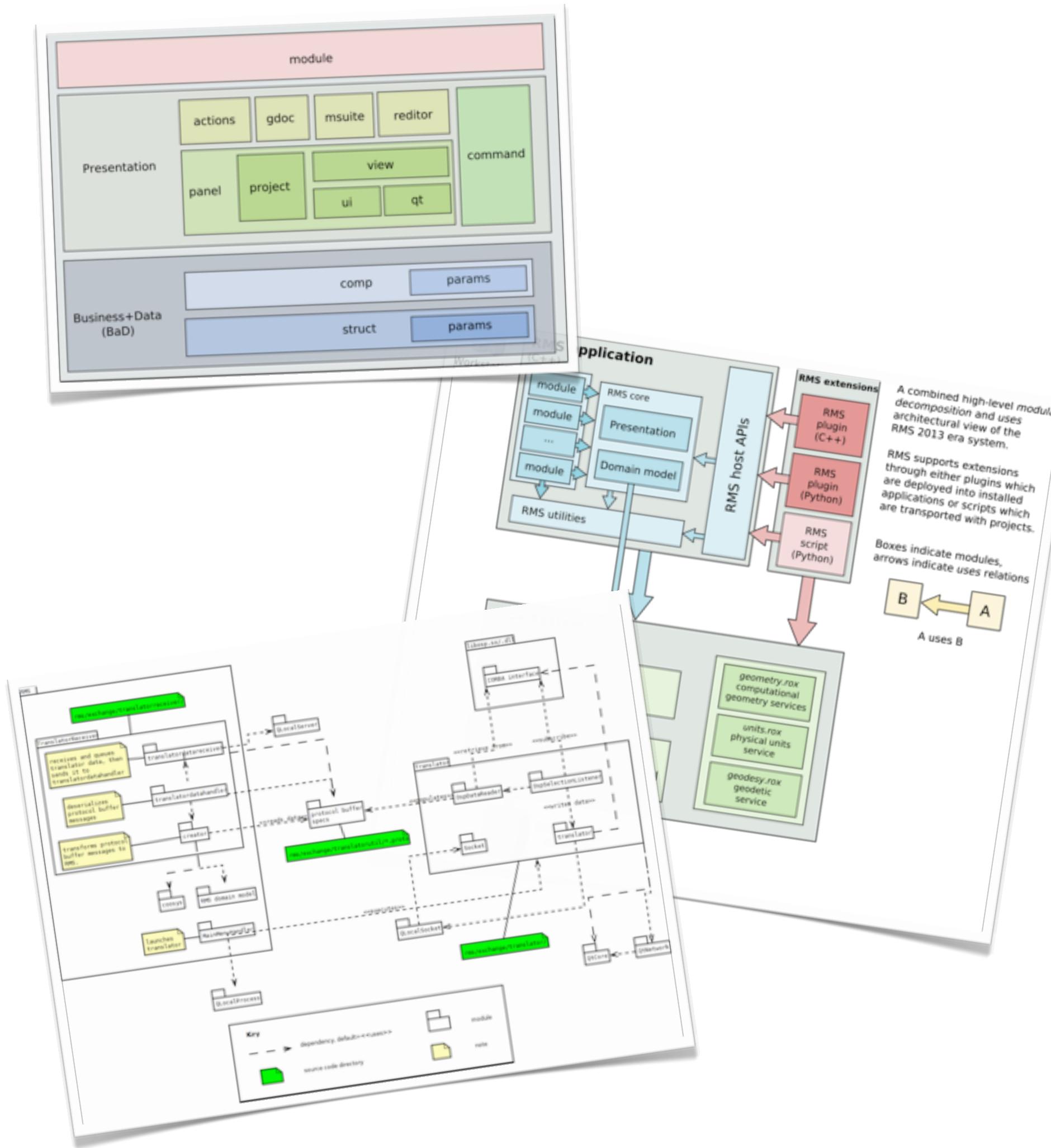
How does this happen?

What takes us from this ...



How does this happen?

... to this?



How does this happen?

The tragedy unfolds inexorably, one dependency at a time.

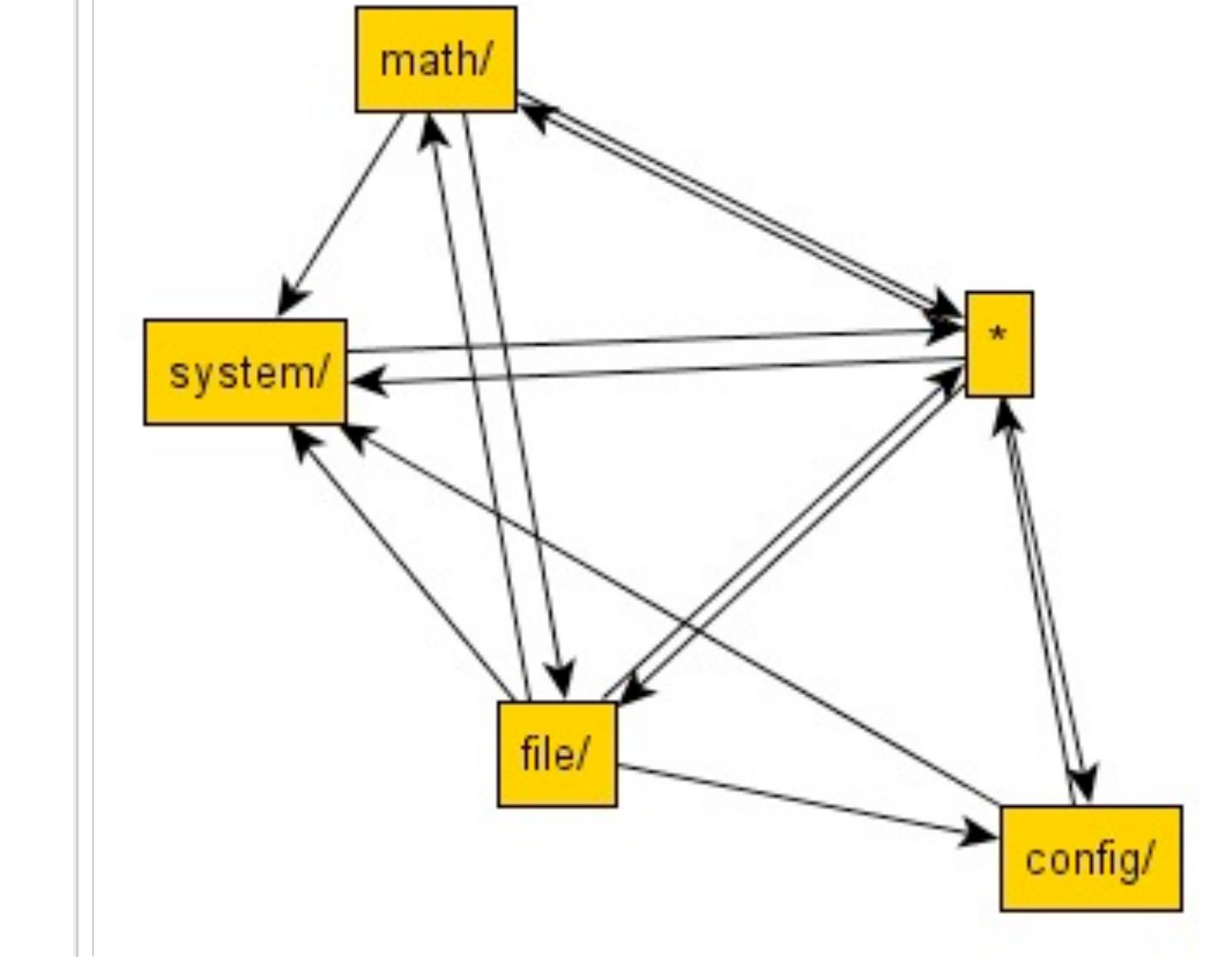
```
using App.Foo.Bar
```

```
import App.Foo.Bar
```

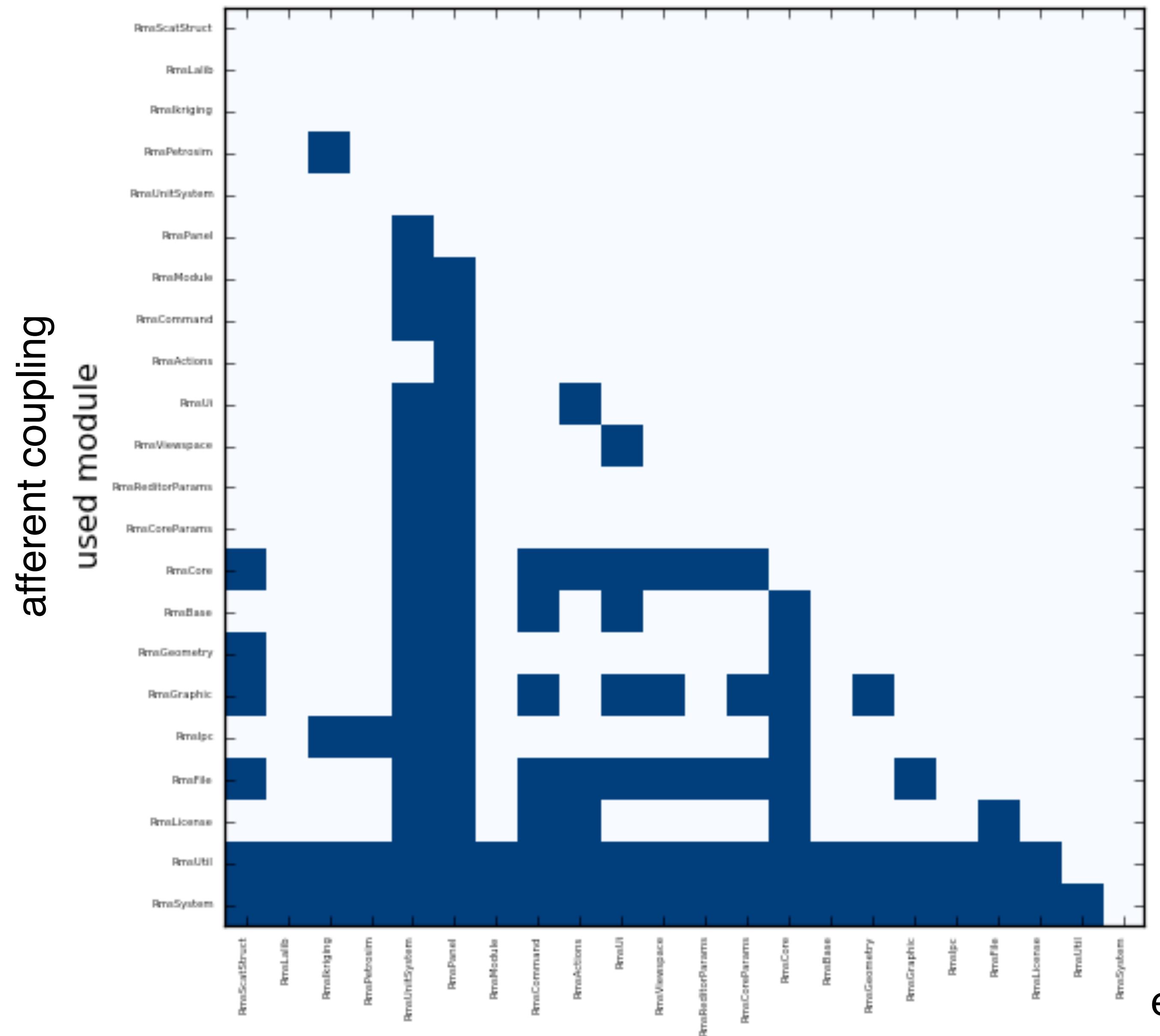
```
from App.Foo import Bar
```

```
#include <app/foo/bar.h>
```

```
require "app/foo/bar"
```



Dependency Structure Matrix



using module
efferent coupling

1

Characterising Complexity

What is complexity and how can we measure it?

2

Evolution of Entropy

Can we get out of the game?

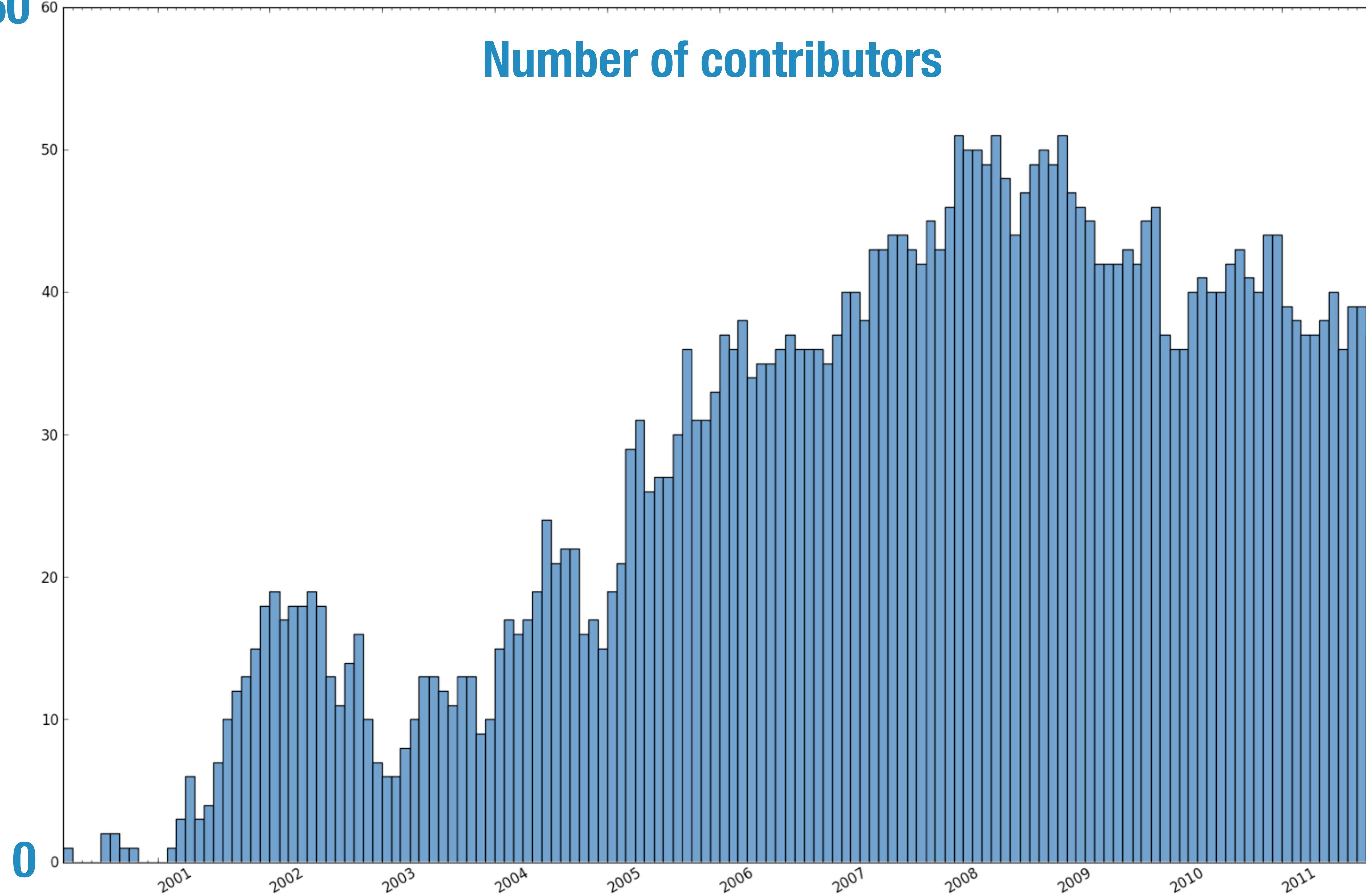
3

Predicting the Past

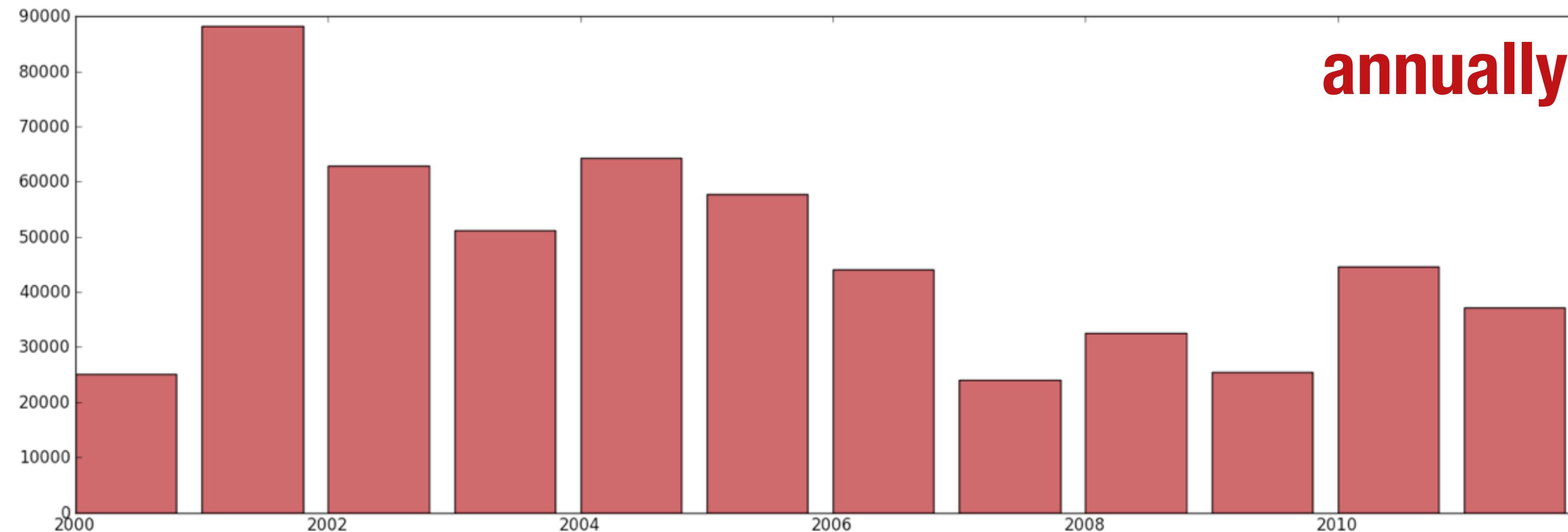
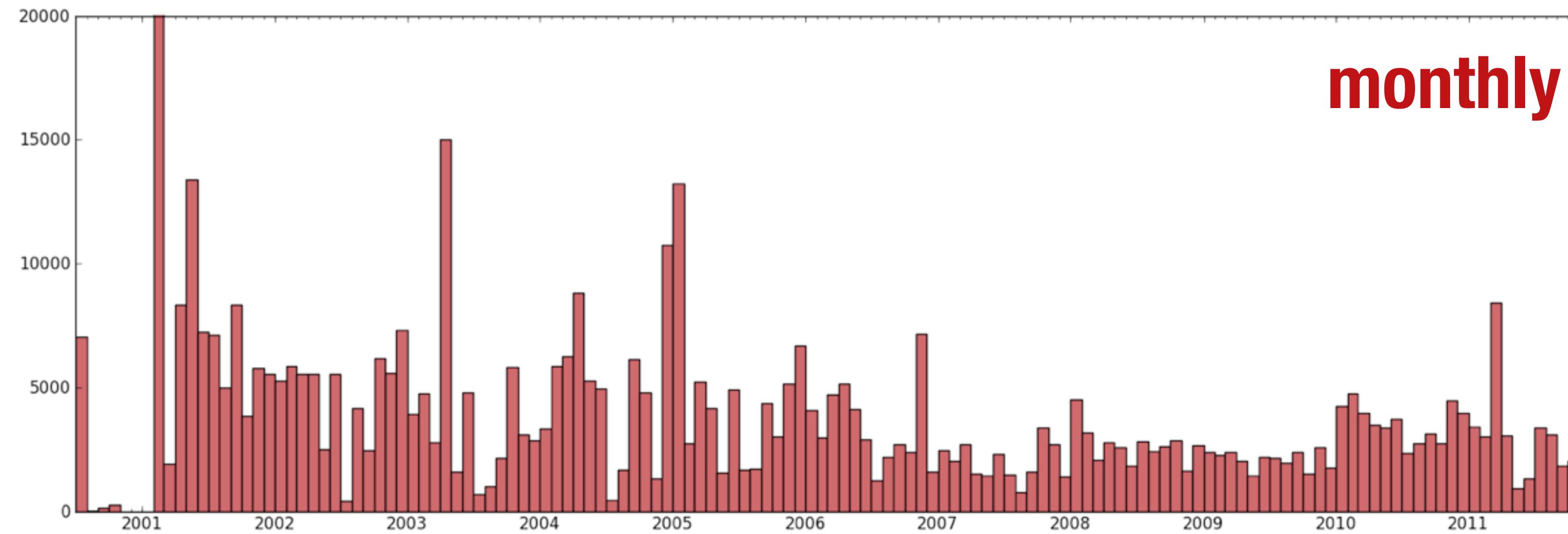
A model driven approach.

60

Number of contributors



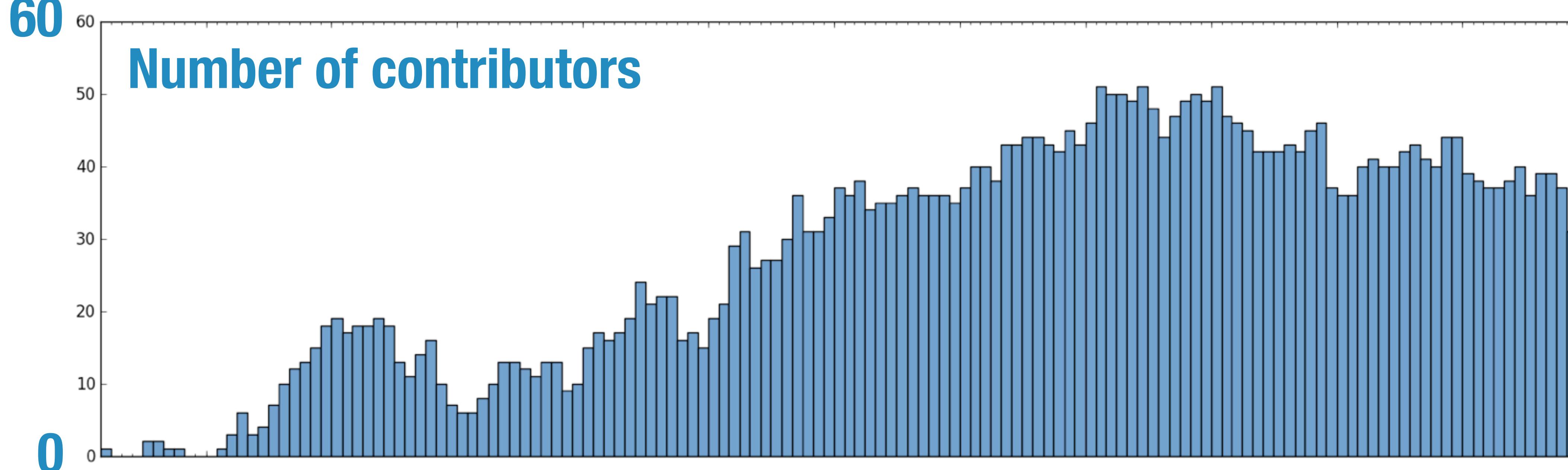
"Productivity" $\text{abs}(\text{diff_size})$



60

Number of contributors

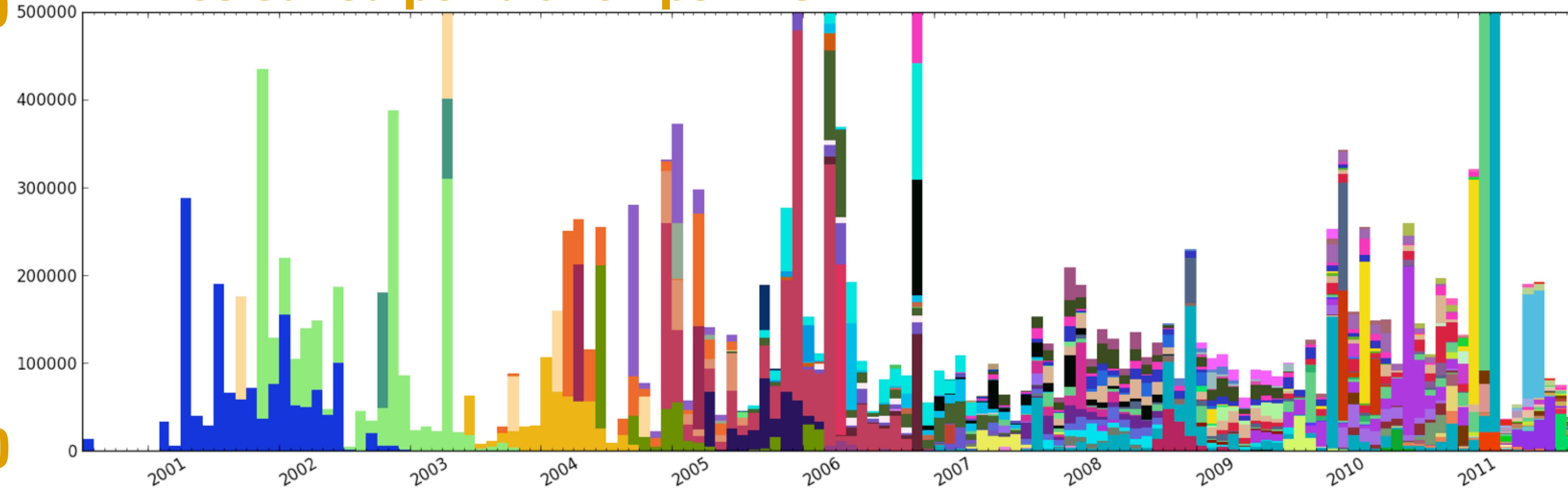
0



500000

Lines edited per branch per month

0



20000801



2000

20000801



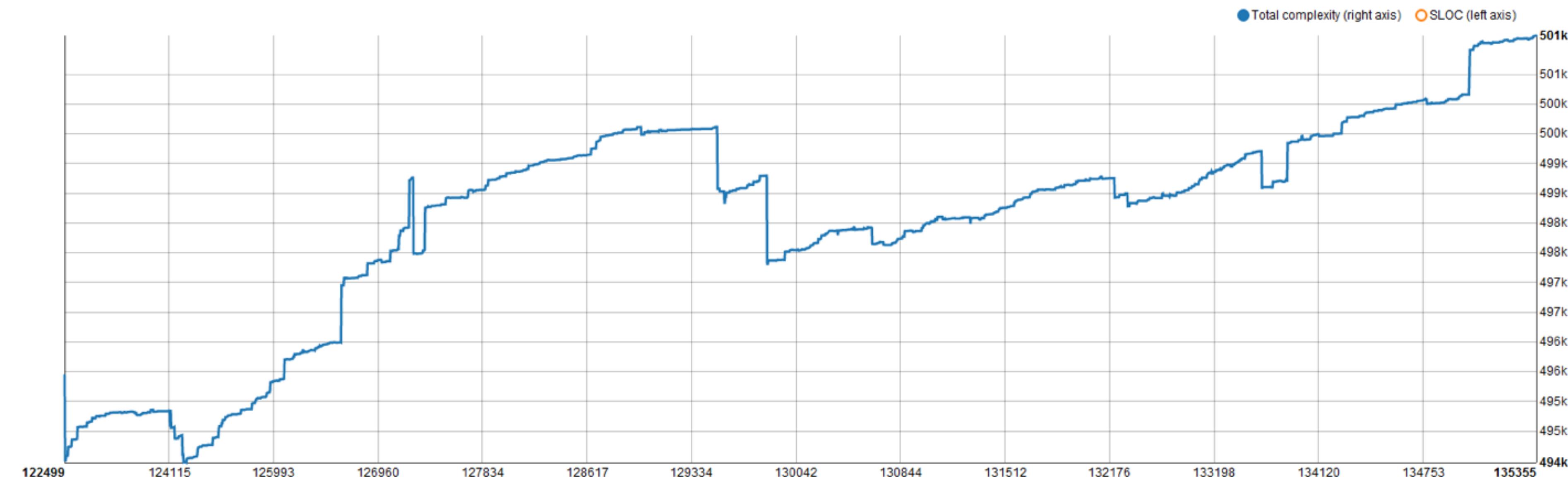
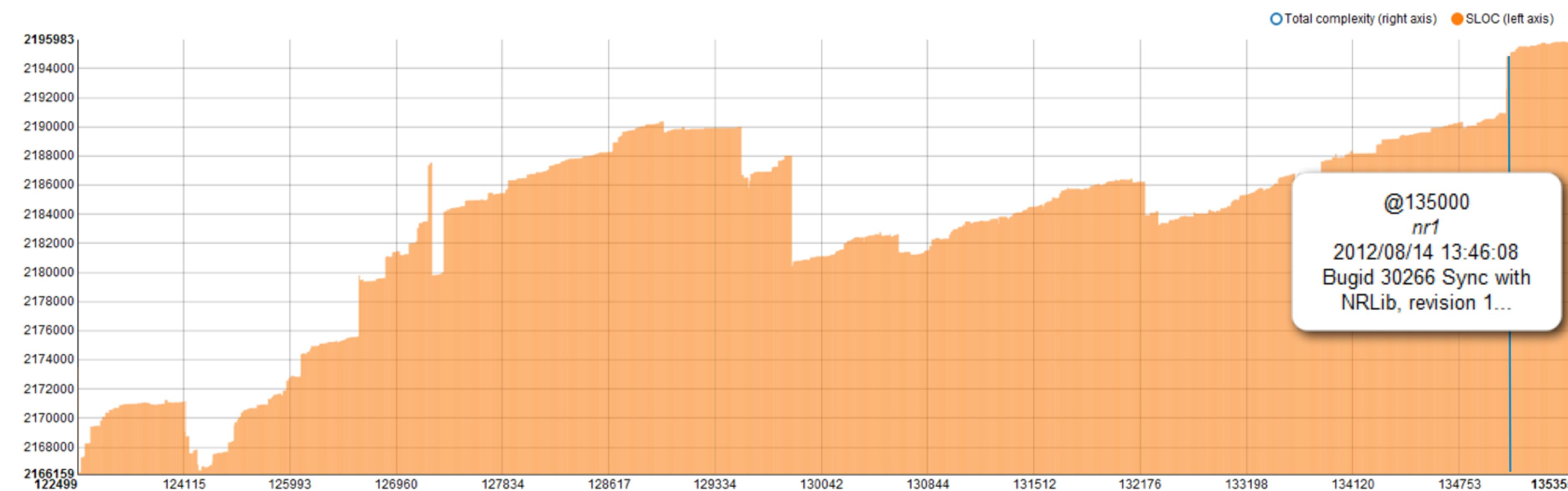
800 000 LOC
complexity density = 0.24

2012

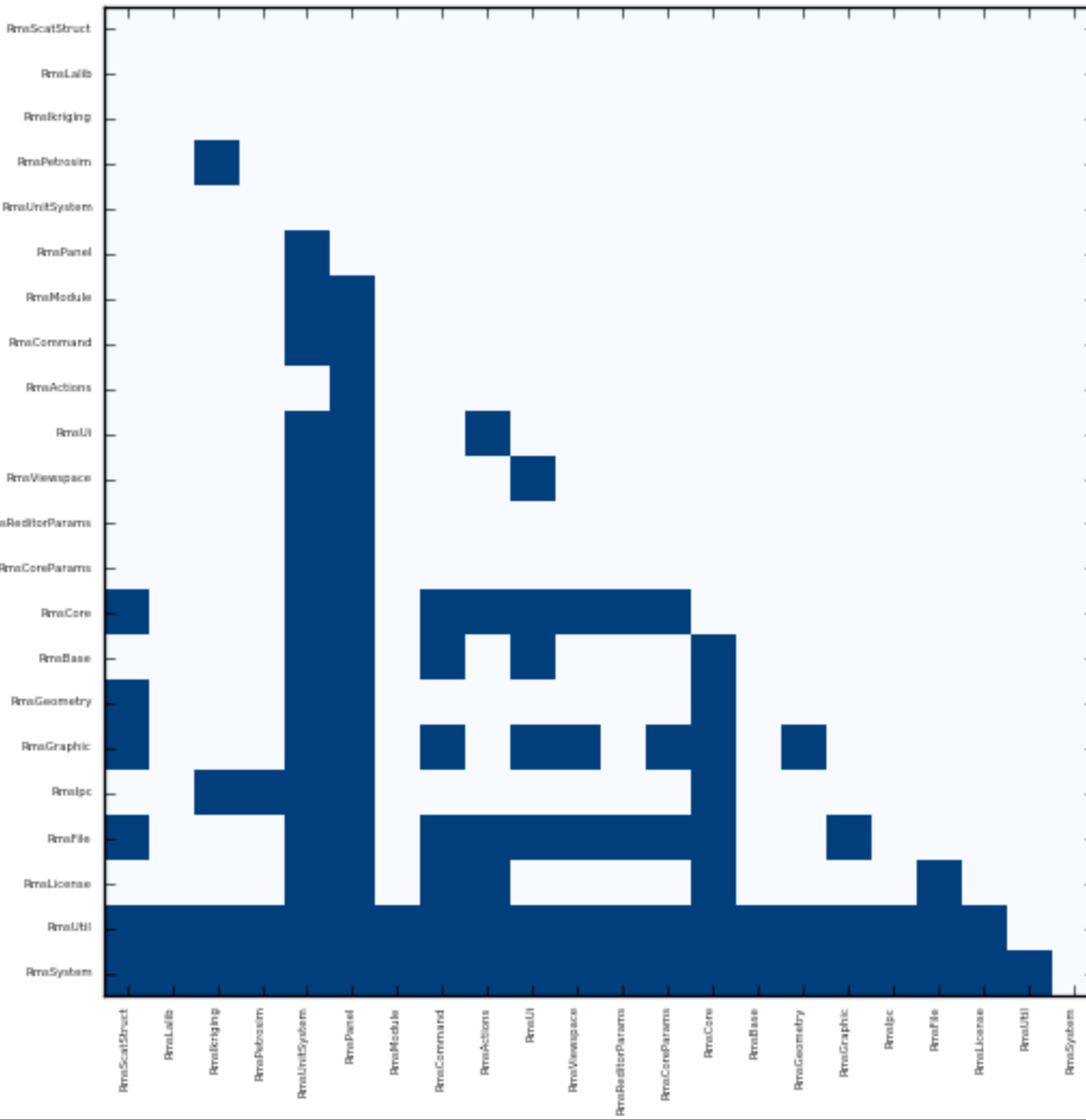
20120319



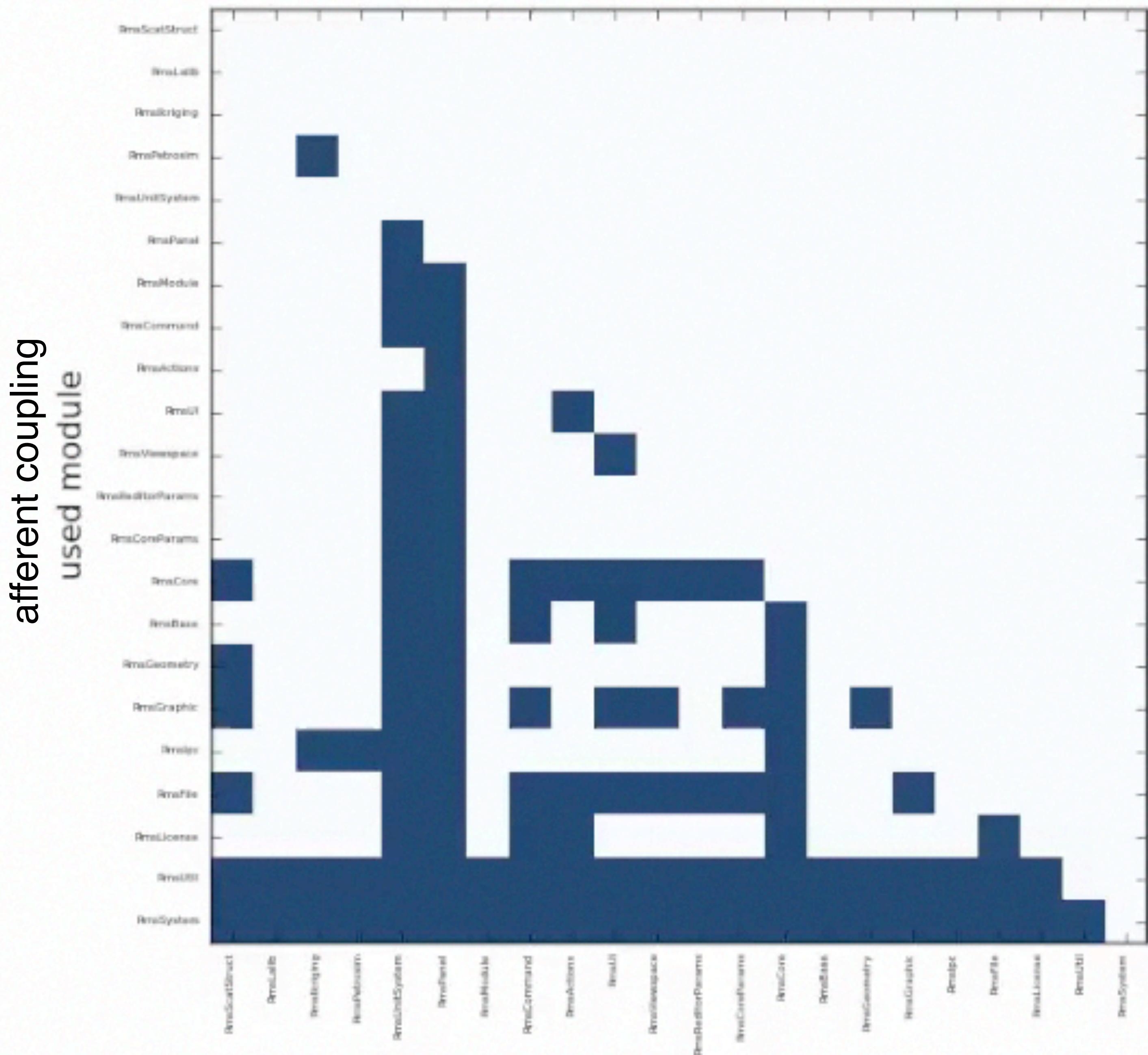
2 200 000 LOC
complexity density = 0.22



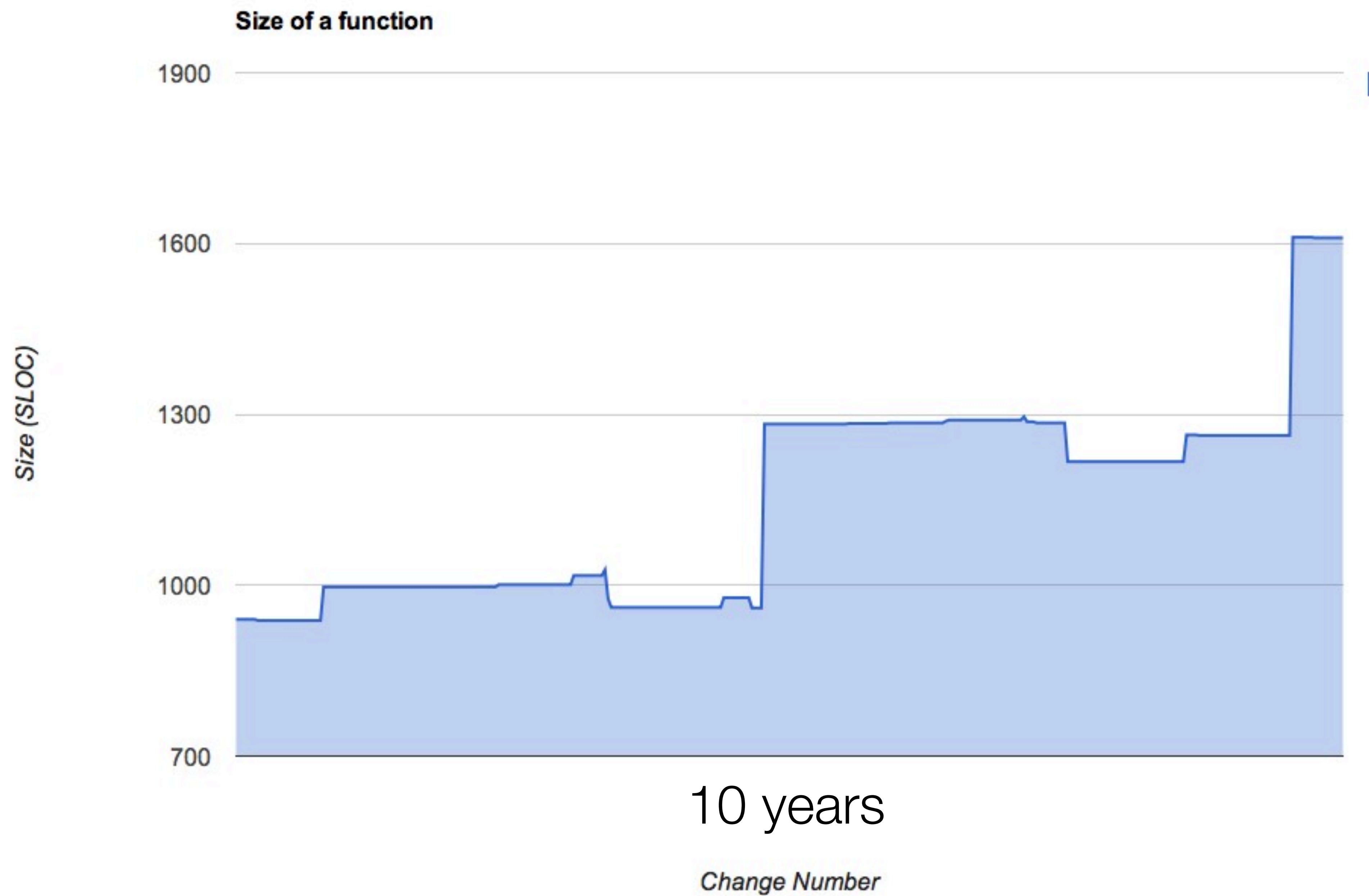
**afferent coupling
used module**



**using module
efferent coupling**



using module efferent coupling



Transparent Metrics to Raise Awareness.

Beware targets!



1

Characterising Complexity

What is complexity and how can we measure it?

2

Evolution of Entropy

Can we get out of the game?

3

Predicting the Past

A model driven approach.

Scientific Method (1 serving)

1. Ask a question.
2. Formulate a hypothesis.
3. Perform experiment.
4. Collect data.
5. Draw conclusions.

Bake until thoroughly cooked.

Garnish with additional observations.

Too simple!

Experimental Science

Randomised controlled trials

- Developers don't like to be watched
- Eliminating extraneous factors
- Toy problems aren't realistic
- No two projects are the same
- Can't do double-blind
- Students have little experience
- Time and money





How can we know?

Prediction

Formulate a hypothesis.

Modelling

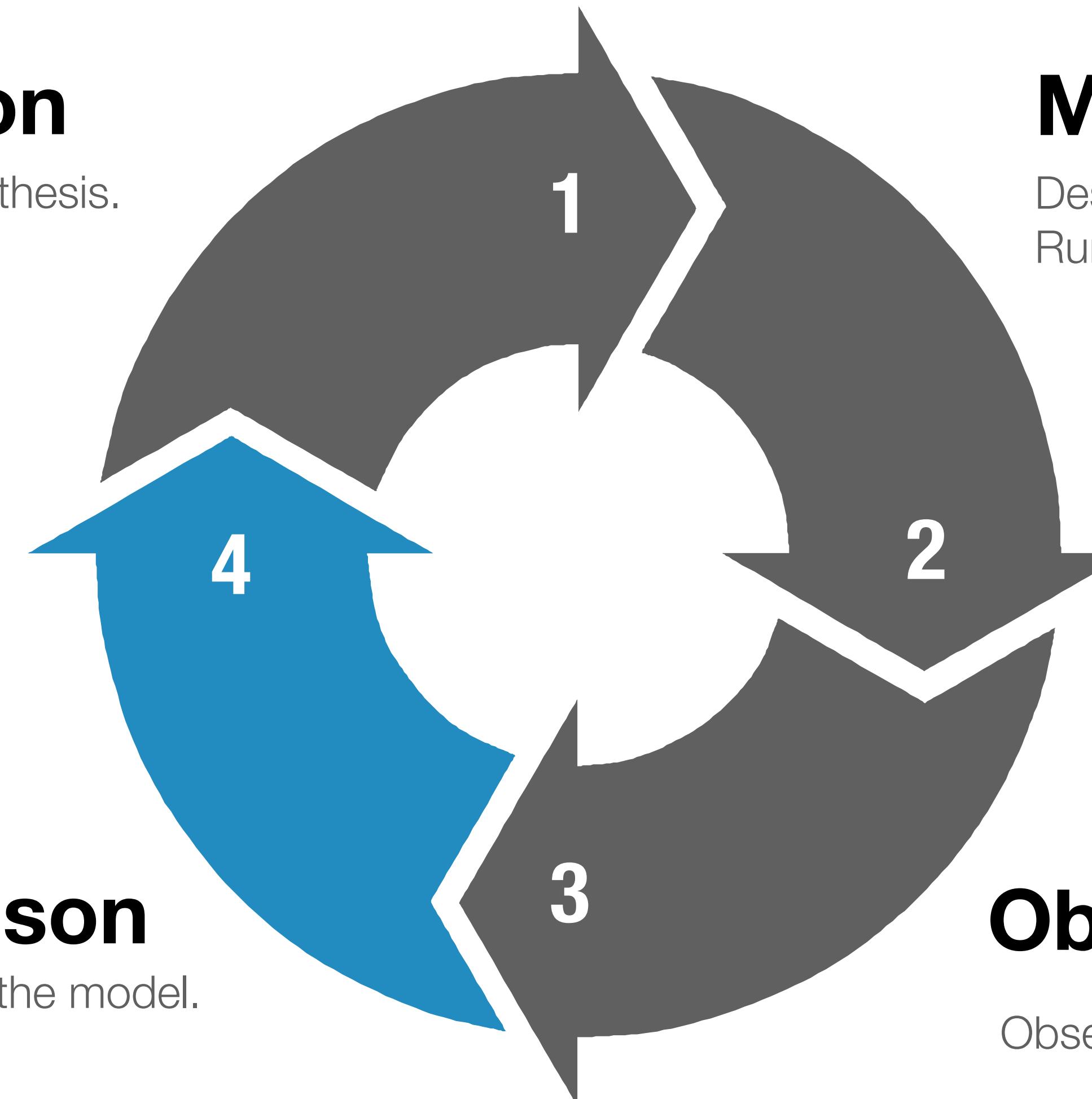
Design a conceptual model.
Run simulations.

Comparison

Validate or refute the model.

Observation

Observe and record reality.

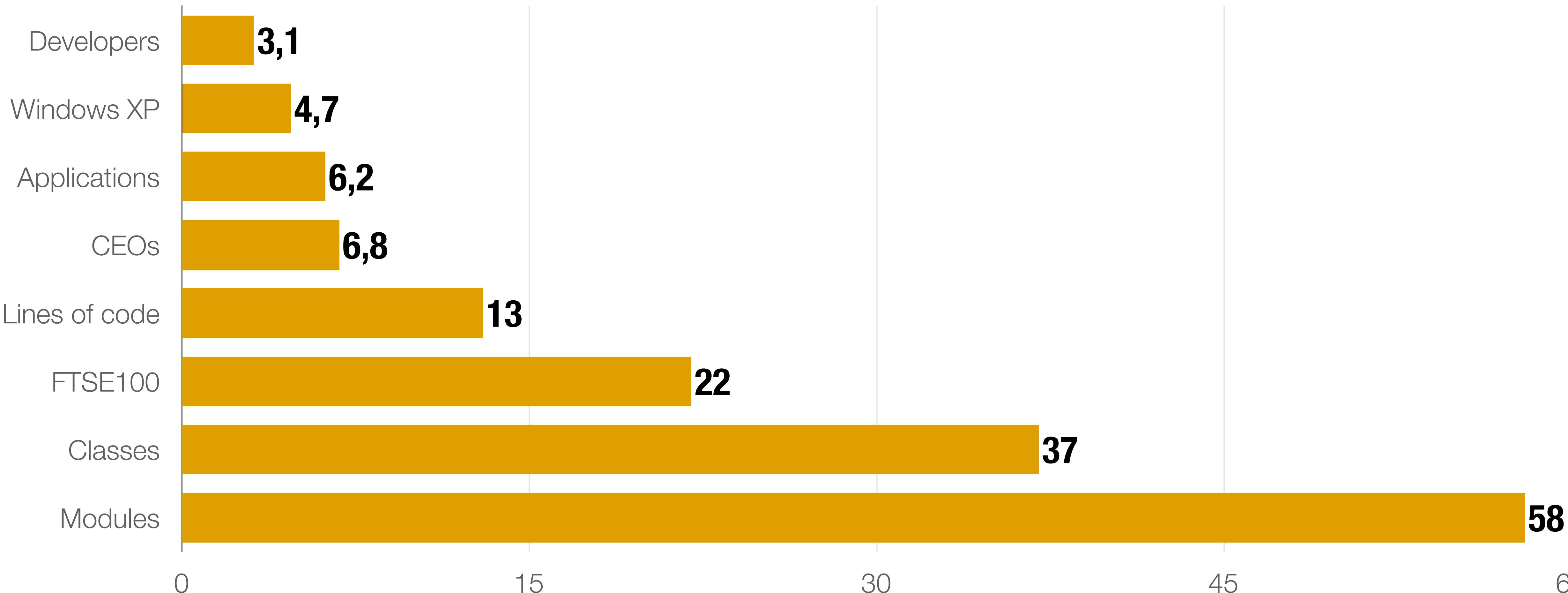


Lifetimes in the software industry

Systems and their architectures are long lived

Half-lives of software related entities

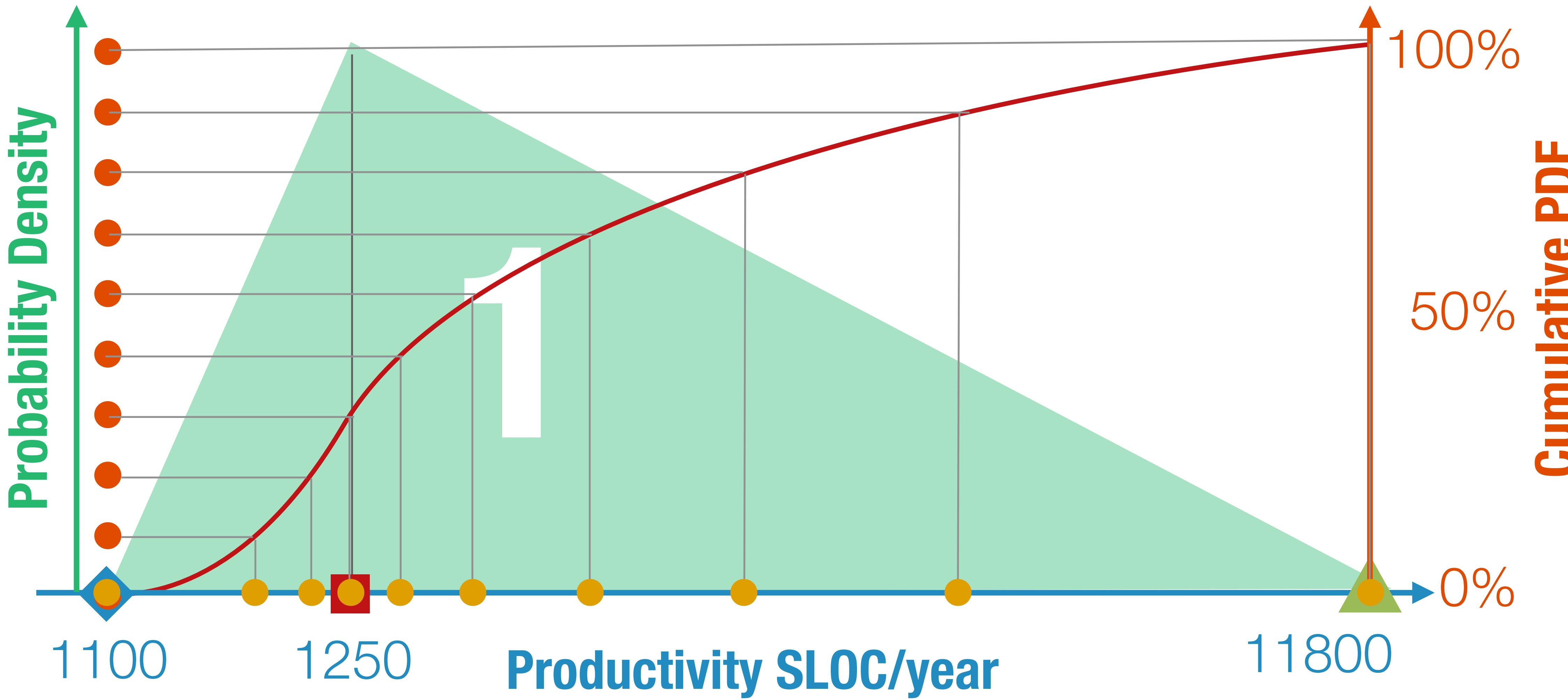
The number of years over which half the entities are replaced

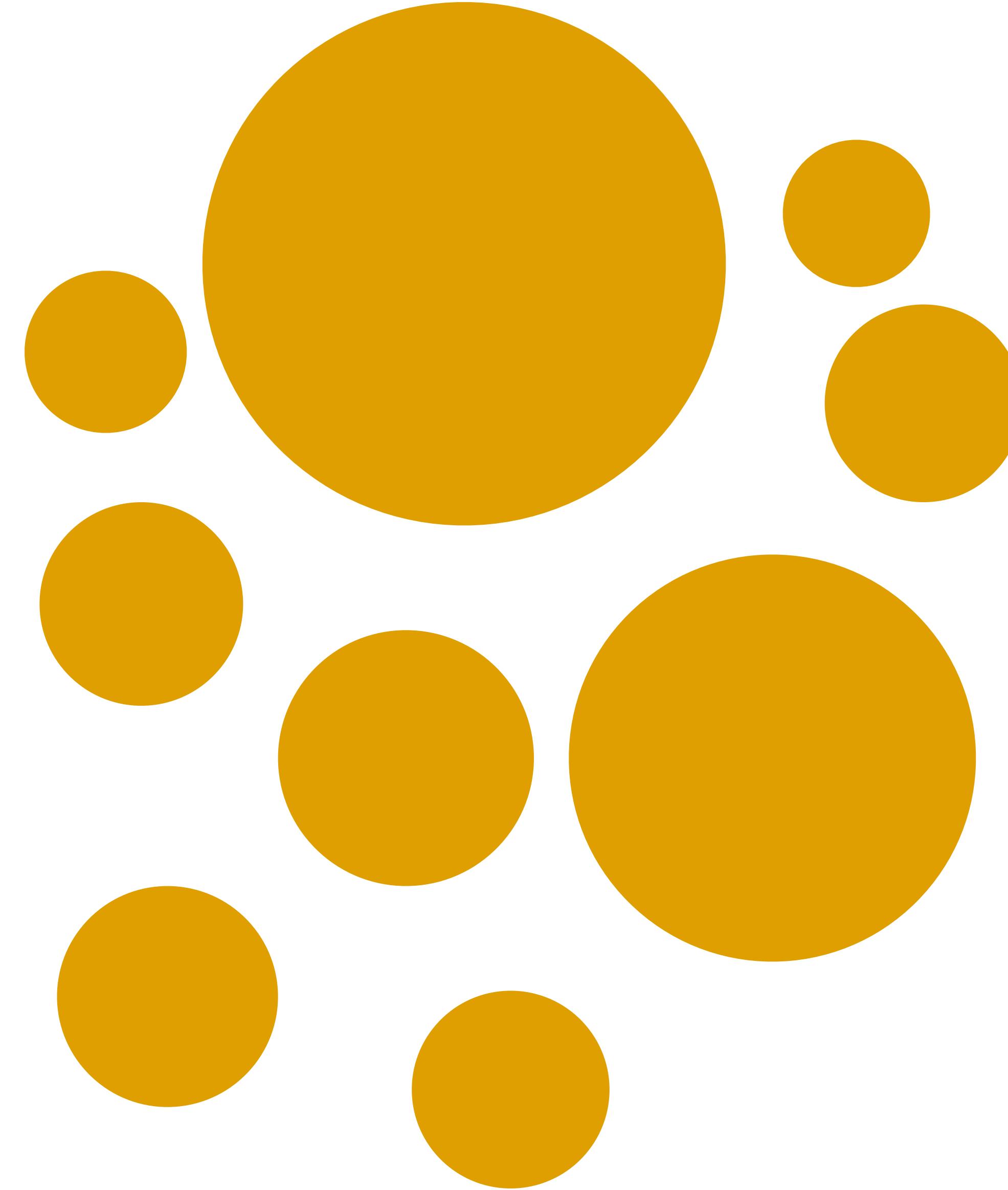


Monte Carlo simulation

Perform a large number of random simulations of developments teams

Productivity on 10000 SLOC codebase

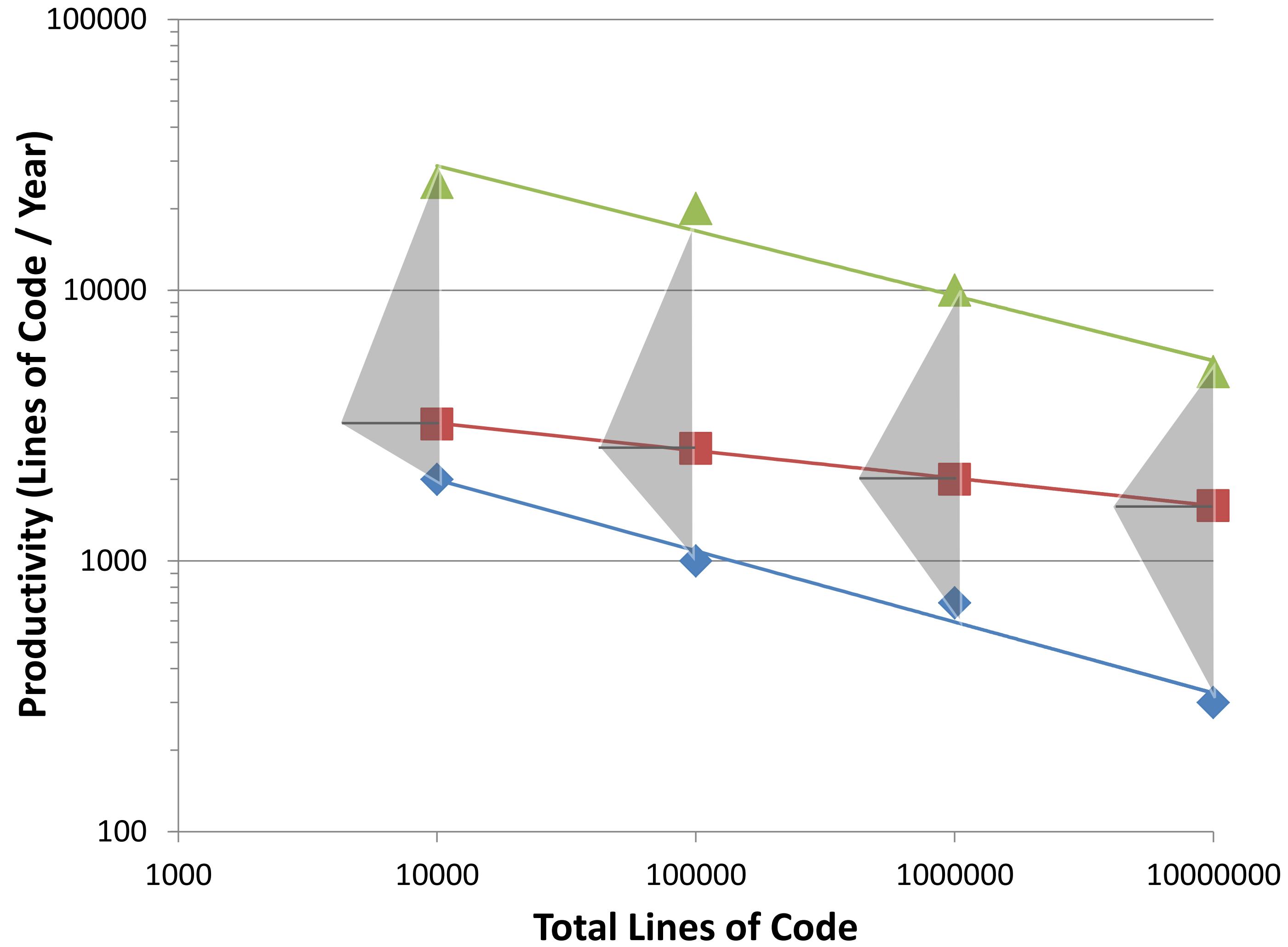


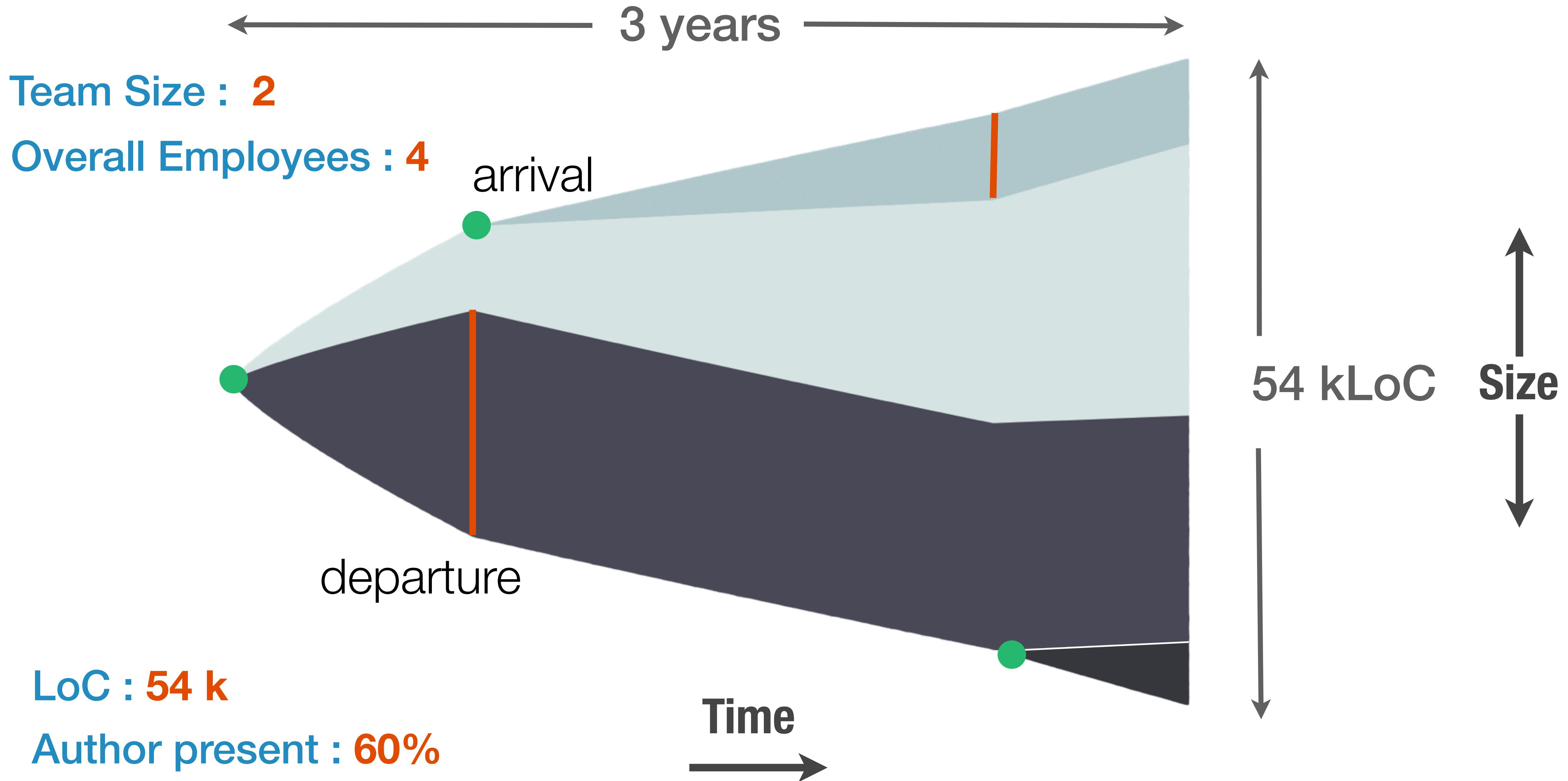


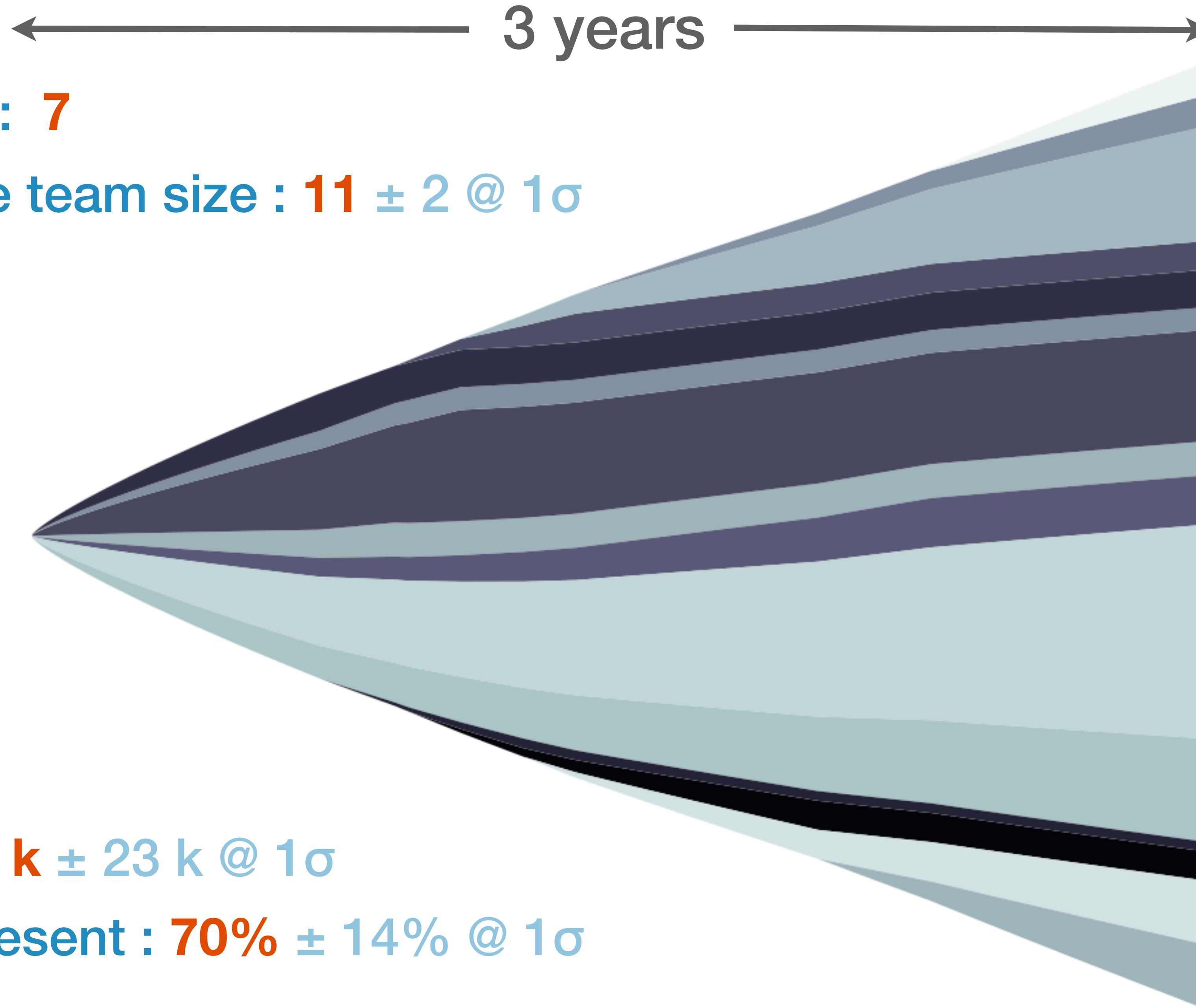
Modelling team and code evolution

Use published productivity data to forward model code size.

At any given system size we can predict a distribution for developer productivity.



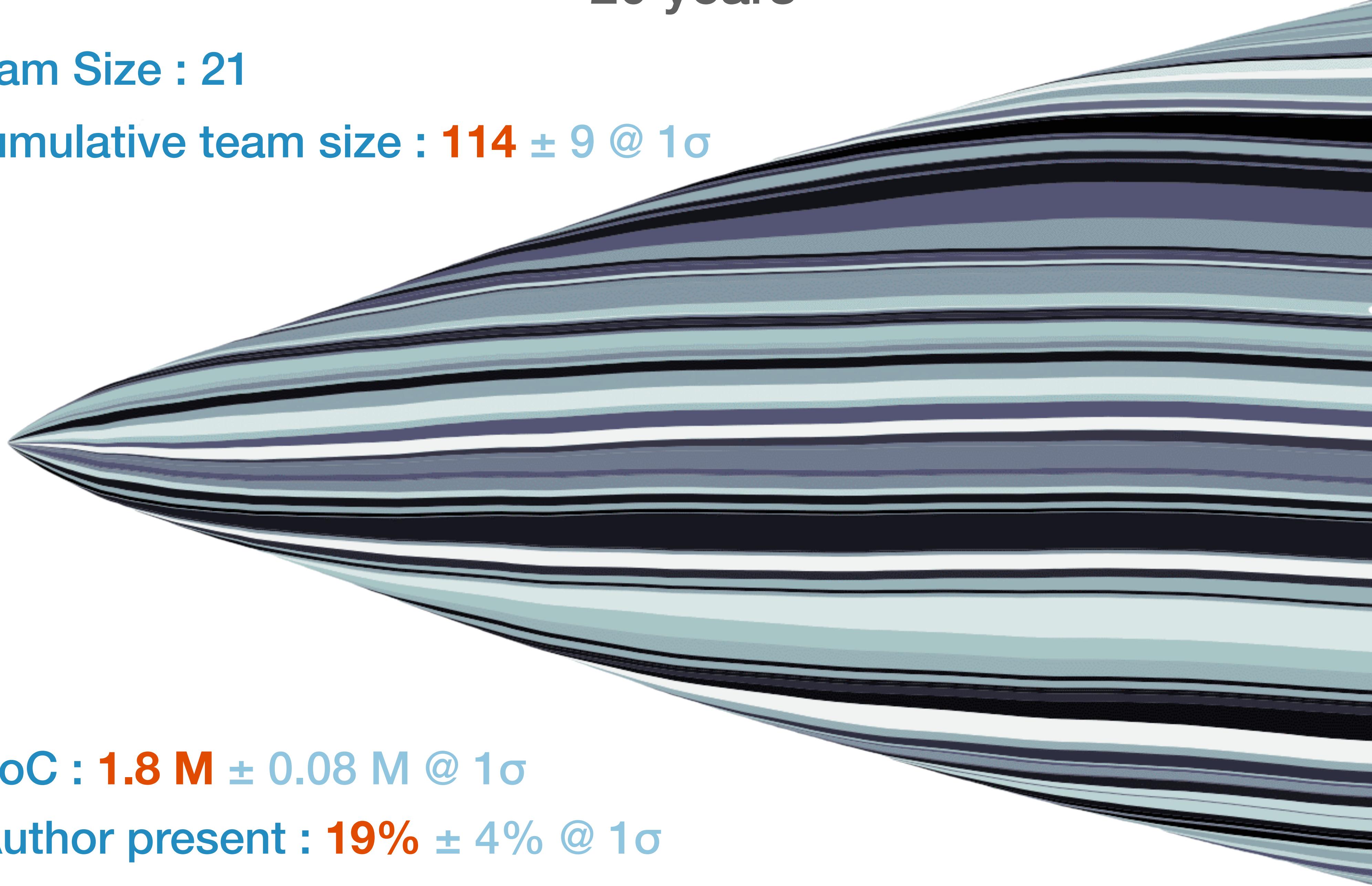




← → 20 years

Team Size : 21

Cumulative team size : **114 ± 9 @ 1 σ**



LoC : **1.8 M ± 0.08 M @ 1 σ**

Author present : **19% ± 4% @ 1 σ**

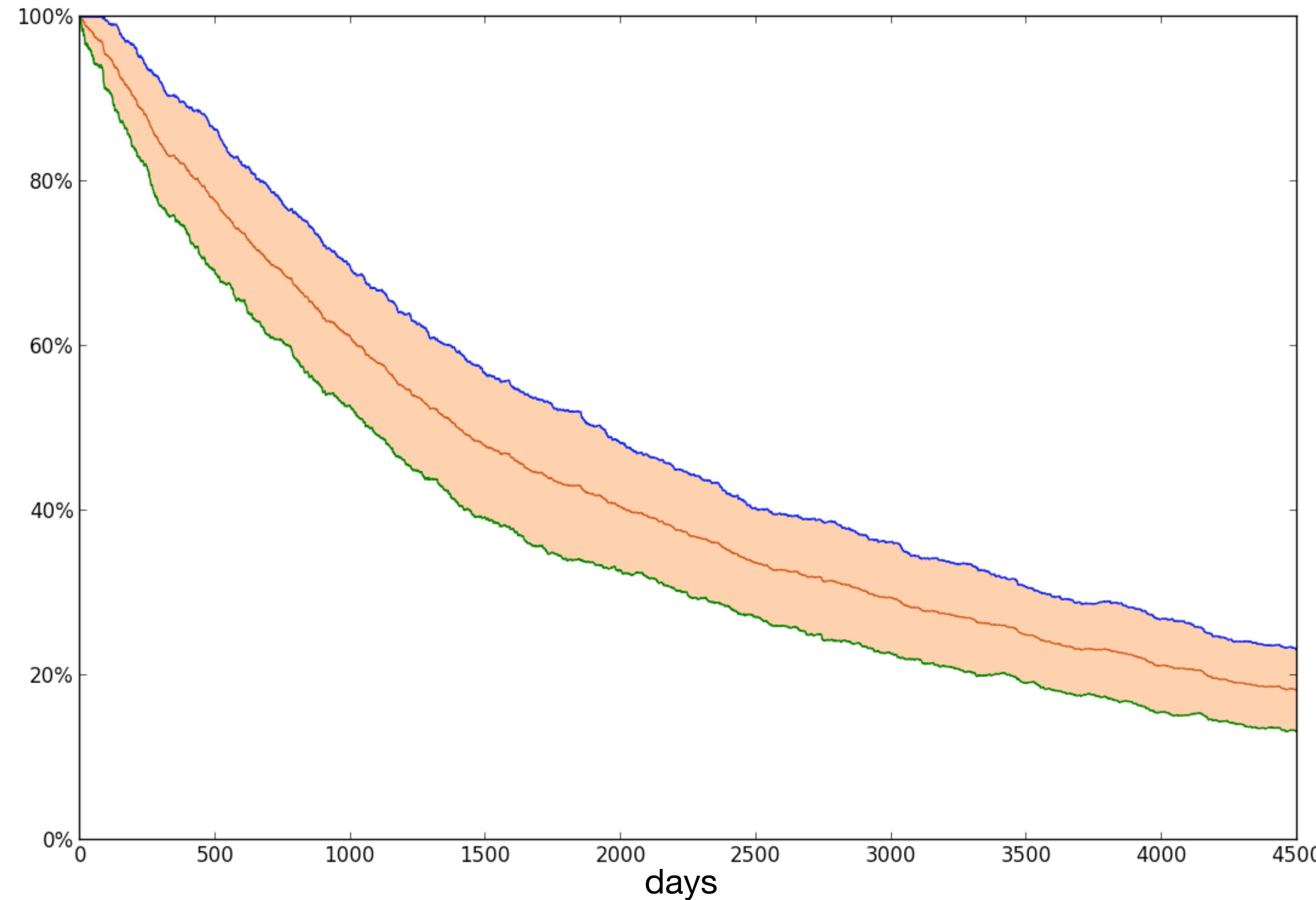
1.8 MLoC



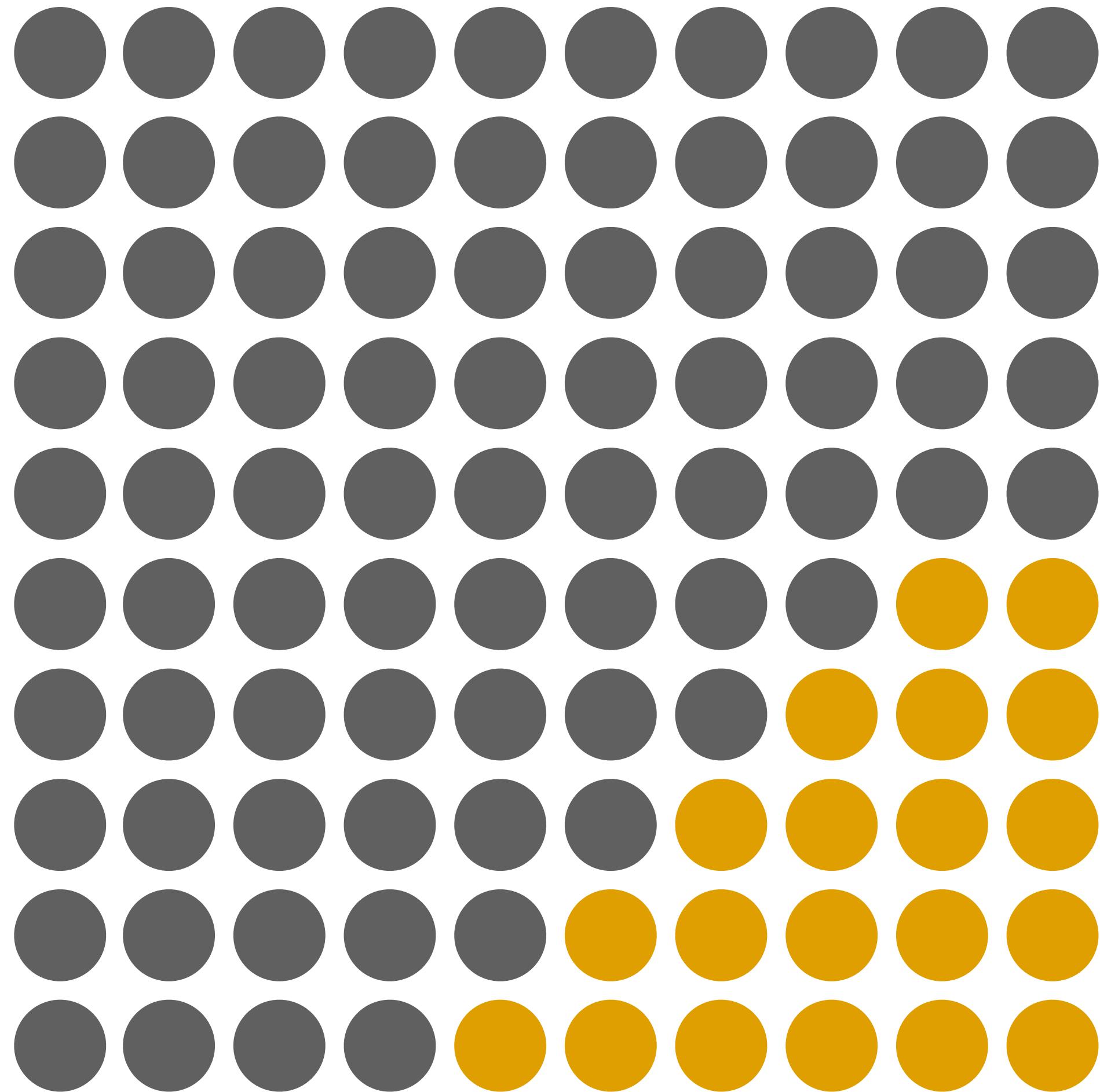
Who can you still talk to?

Most authors of your product quit way back when.

Proportion of code
written by current
colleagues.

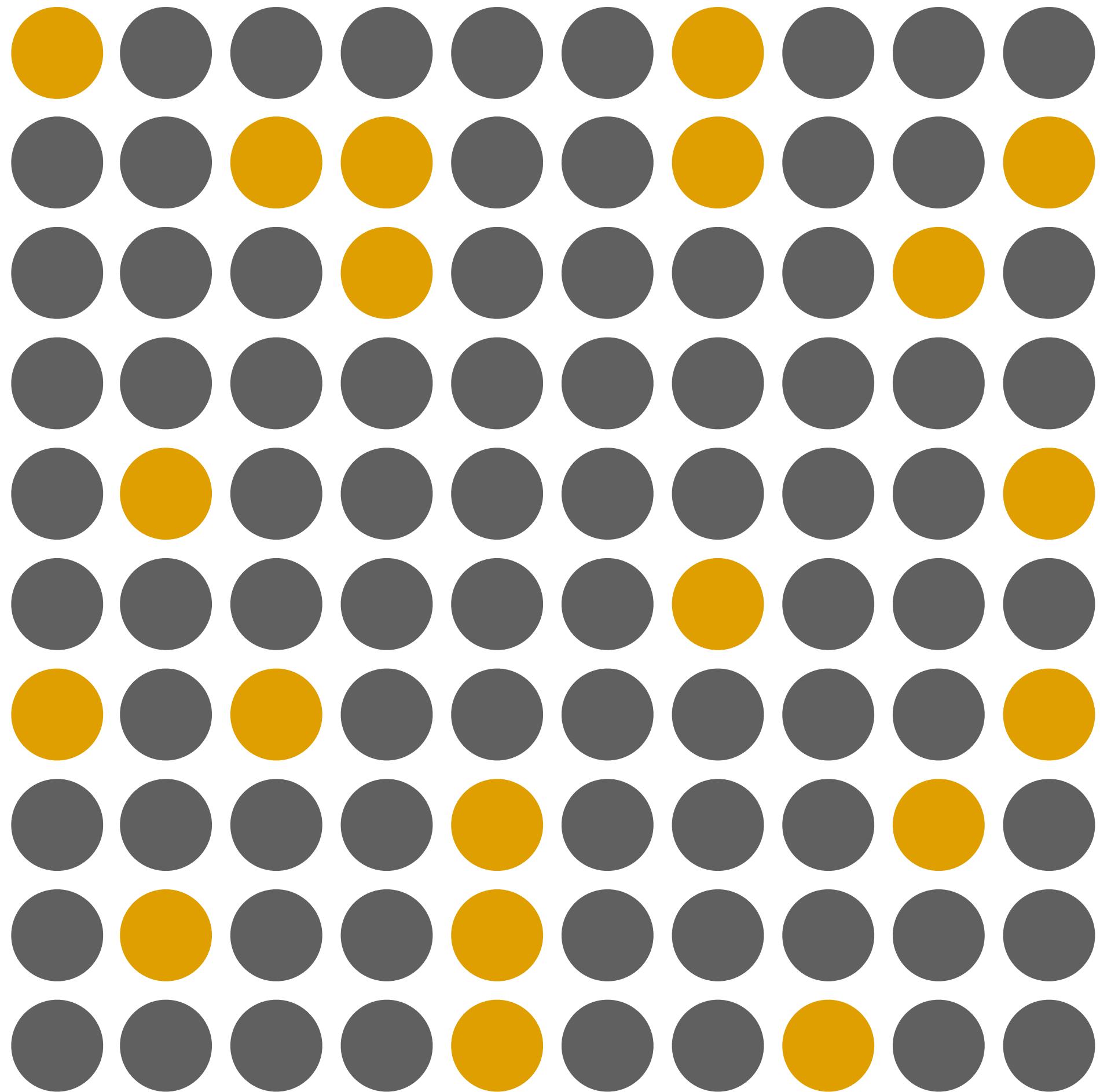


**20% after
20 years**



20%

of your code was written by people on
your team today after 20 years of
development.



20%

Working as a relatively small team in a large codebase can be lonely, disorienting and confusing.

Architecture provides continuity.

Thank you!

Questions?

Robert Smallshire

 @robsmallshire

SixtyNORTH



 @sixty_north

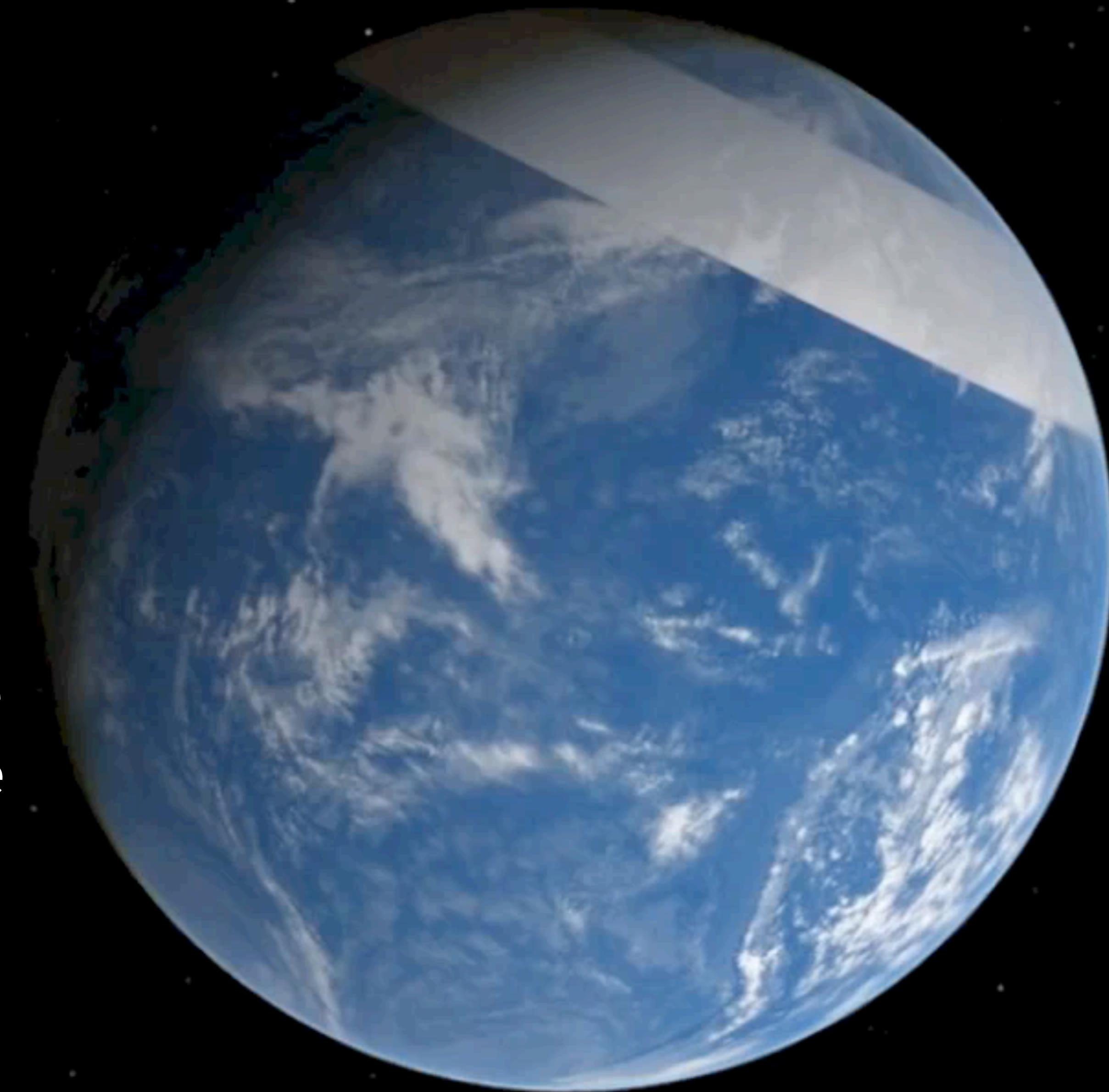
Thank you!

Questions?

Robert Smallshire

 @robsmallshire

SixtyNORTH



 @sixty_north



SixtyNORTH