COM401 Software Engineering

Lab-9

Res.Asst Arzum KARATAŞ

December 23,2014

Agenda

- Metrics
- An Example from the Lecture Notes
- Wet your feet!
- Project Evaluation Session
- Helpful Links
- Q&A

Metrics

In software development, a metric is the measurement of a particular characteristic of a program's performance or efficiency. Similarly in network routing, a metric is a measure used in calculating the next host to route a packet to. A metric is sometimes used directly and sometimes as an element in an algorithm . In programming, a benchmark includes metrics.[1]

Benefits of Metrics

- You can measure the complexity of rails code
- Creates a “hit list” of most complex methods
- Reveals hidden issues like bugs and other complexities
- Examine worst offenders
- Refactoring complex methods
- Creates smaller, easier to understand methods

[1] http://whatis.techtarget.com/definition/metric

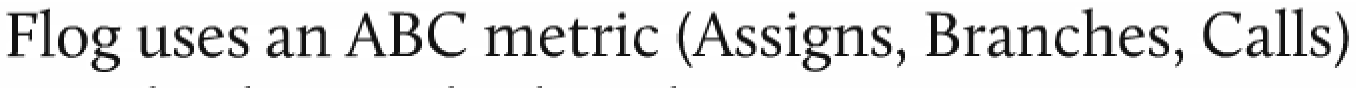
A metric tool :Metric_fu

- It is created by **Jake Scruggs**
- **Metric_fu** is a compilation of several different tools that provides reports. This show which parts of your rails code might need refactoring.

Metric_Fu Sub Components

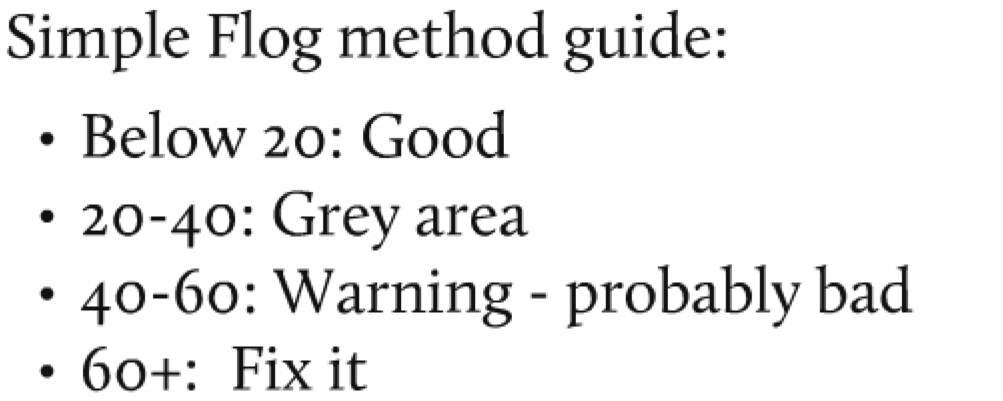
- Flog – measures code complexity
- Saikuro – measures cyclomatic complexity
- Flay – measures structural similarities
- Reek – spots code smells
- Roodi – finds lots of general problems(like design issues)
- Churn – identifies files that changes too often
- Rails best practices – code metric tool for rails
- Cane – spots code quality threshold violations
- HotSpot– Meta analysis of your metrics to find hotspots in the code

Flog



Flog uses an ABC metric (Assigns, Branches, Calls)

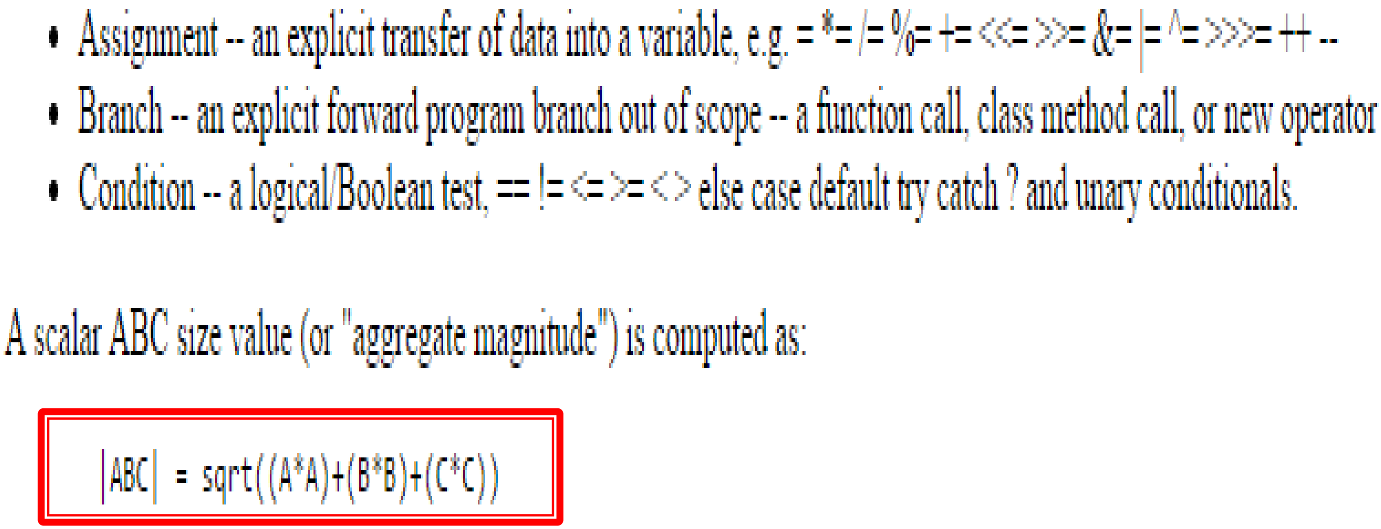
The report Flog generates lists each file in your application in order of its complexity. It’s worth scrolling though the report and looking at the files that score highly in the Highest Score column.



Simple Flog method guide:

- Below 20: Good
- 20-40: Grey area
- 40-60: Warning - probably bad
- 60+: Fix it

Flog: ABC Complexity Calc.

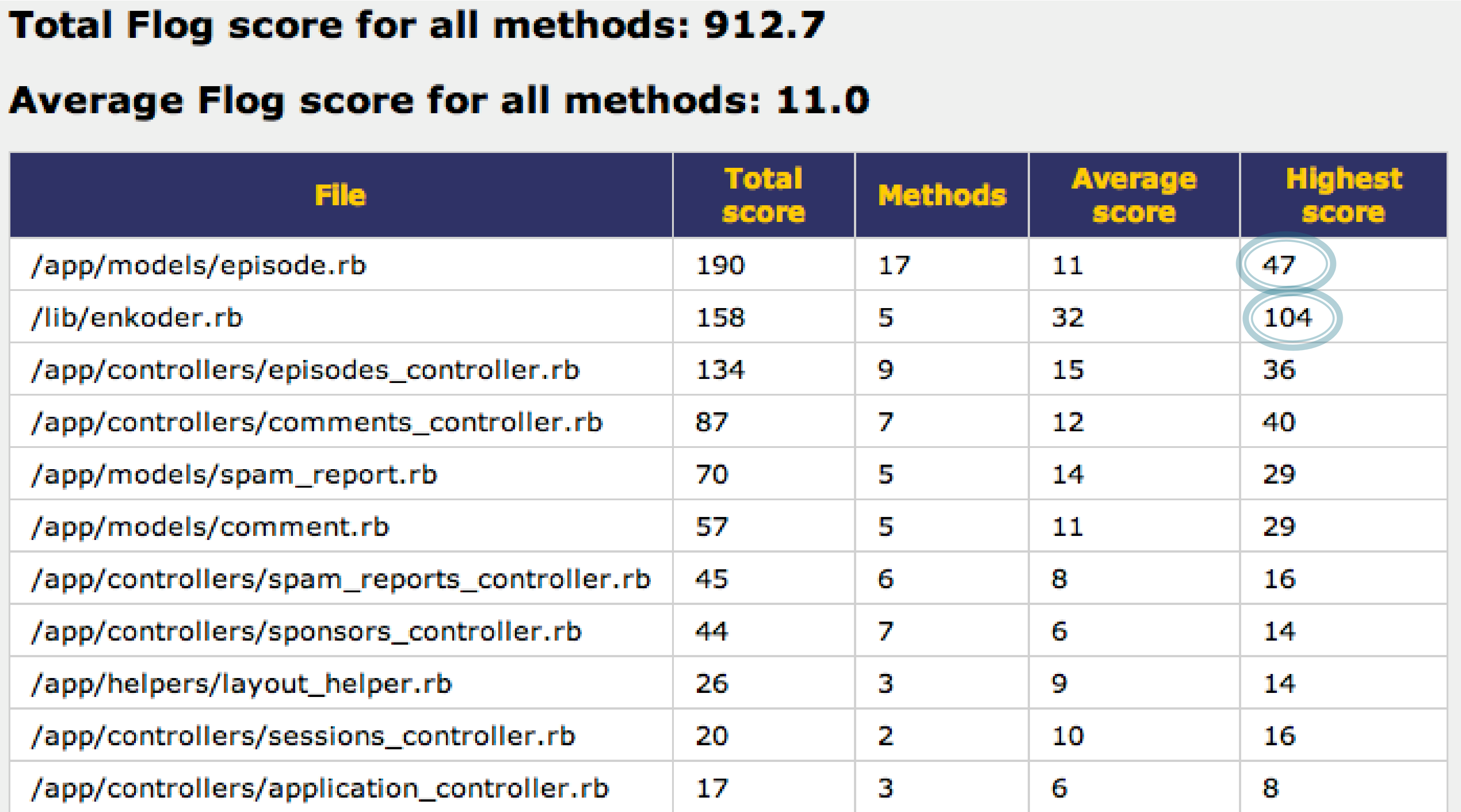


- Assignment -- an explicit transfer of data into a variable, e.g. = *= /= %= += <<= >>= &= |= ^= >>>= ++ --
- Branch -- an explicit forward program branch out of scope -- a function call, class method call, or new operator
- Condition -- a logical/Boolean test, == != <= >= < > else case default try catch ? and unary conditionals.

A scalar ABC size value (or "aggregate magnitude") is computed as:

```
|ABC| = sqrt((A*A)+(B*B)+(C*C))
```

Flog(cont.)



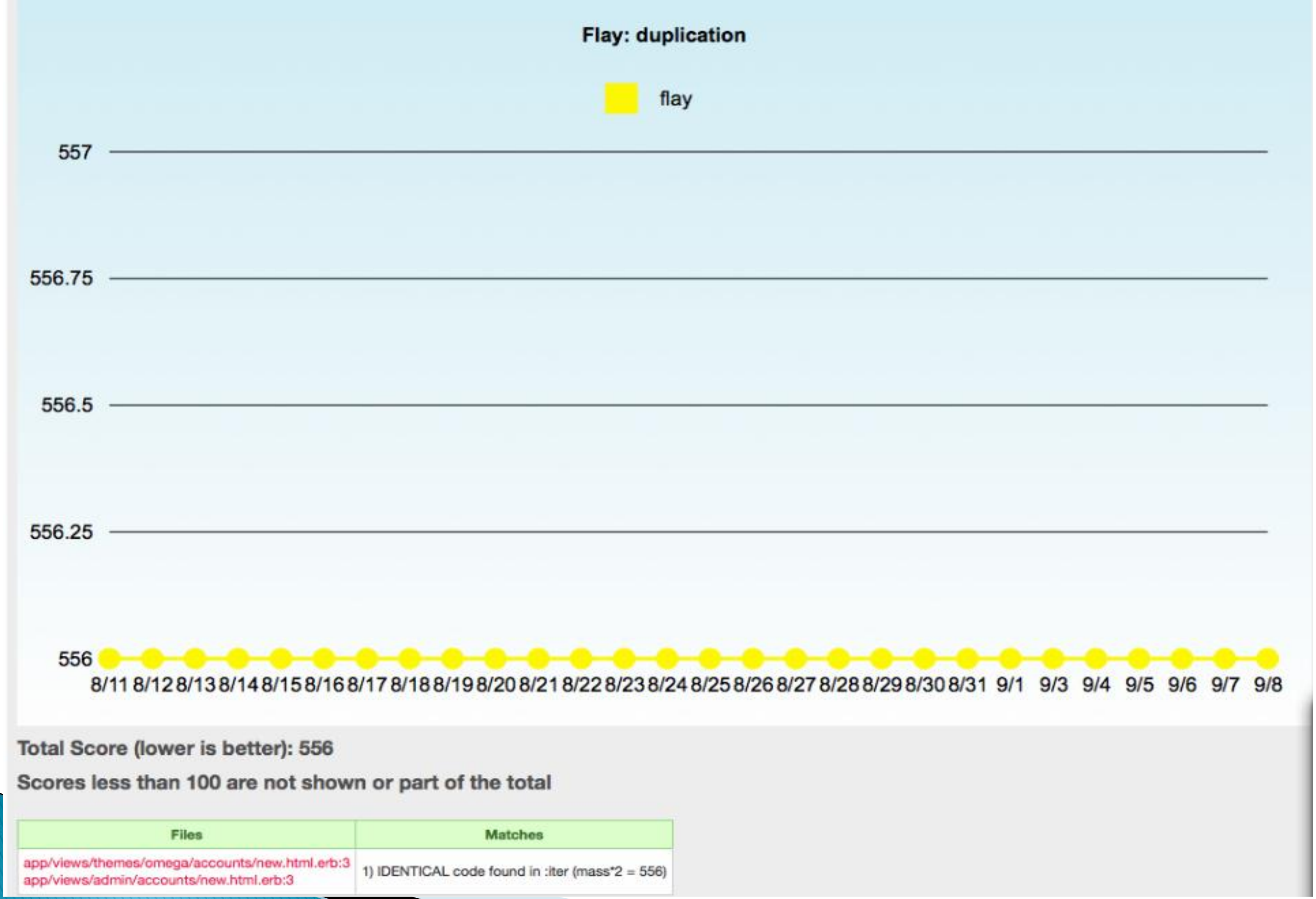
Total Flog score for all methods: 912.7

Average Flog score for all methods: 11.0

File	Total score	Methods	Average score	Highest score
/app/models/episode.rb	190	17	11	47
/lib/enkoder.rb	158	5	32	104
/app/controllers/episodes_controller.rb	134	9	15	36
/app/controllers/comments_controller.rb	87	7	12	40
/app/models/spam_report.rb	70	5	14	29
/app/models/comment.rb	57	5	11	29
/app/controllers/spam_reports_controller.rb	45	6	8	16
/app/controllers/sponsors_controller.rb	44	7	6	14
/app/helpers/layout_helper.rb	26	3	9	14
/app/controllers/sessions_controller.rb	20	2	10	16
/app/controllers/application_controller.rb	17	3	6	8

You have to refactor the files whose highest score is > 40

Flay



Total Score (lower is better): 556

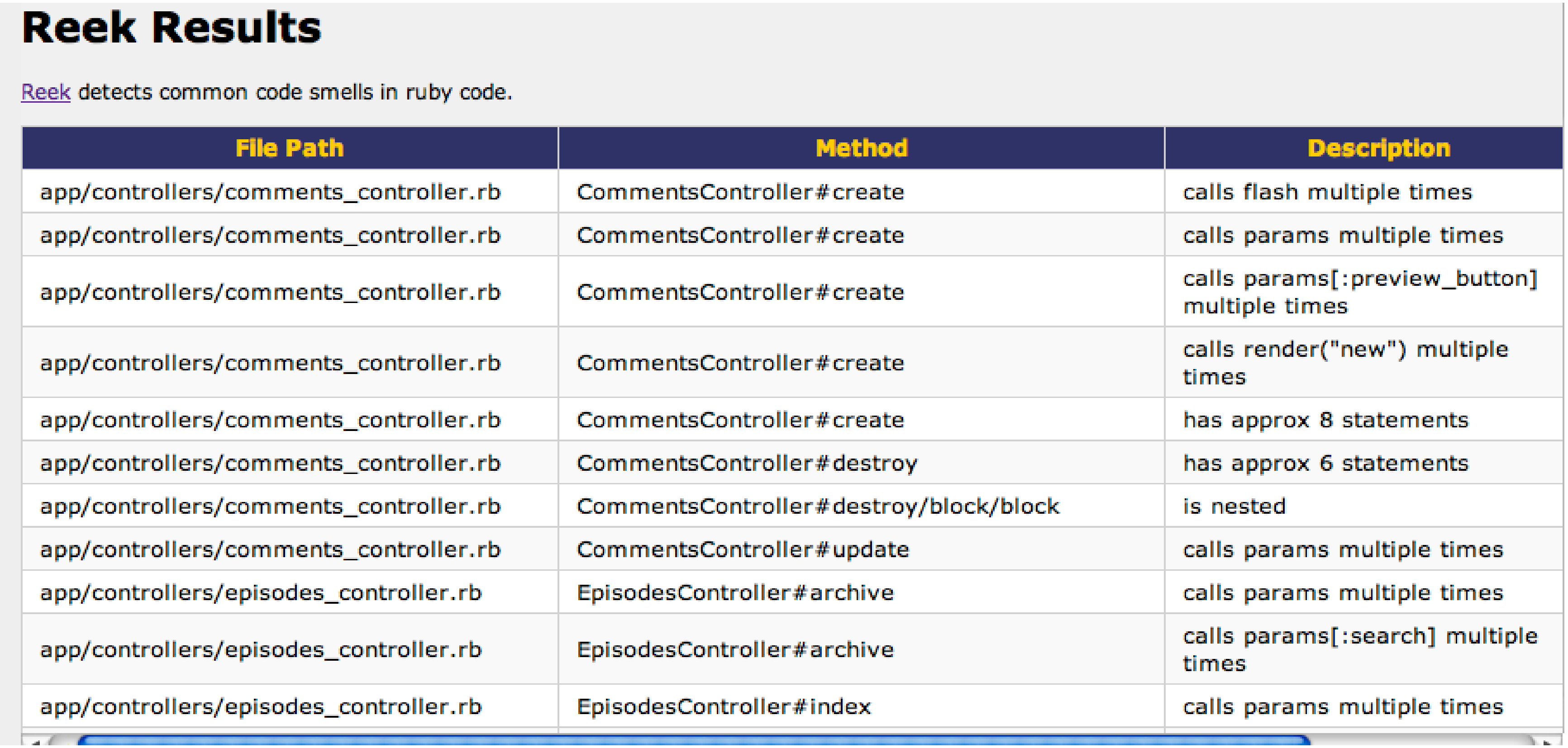
Scores less than 100 are not shown or part of the total

Files	Matches
app/views/themes/omega/accounts/new.html.erb:3 app/views/admin/accounts/new.html.erb:3	1) IDENTICAL code found in :iter (mass*2 = 556)

Reek

- **Reek** is a similar tool to Flay in that it looks for parts of your code that need work.
- It detects “code smells” in your application and reports on them.
- Common smells include code duplication, long methods and repeated calls.
- Reek provides a useful description of each problem which should make it easy for you to find them and refactor the offending code.

Reek(cont.)

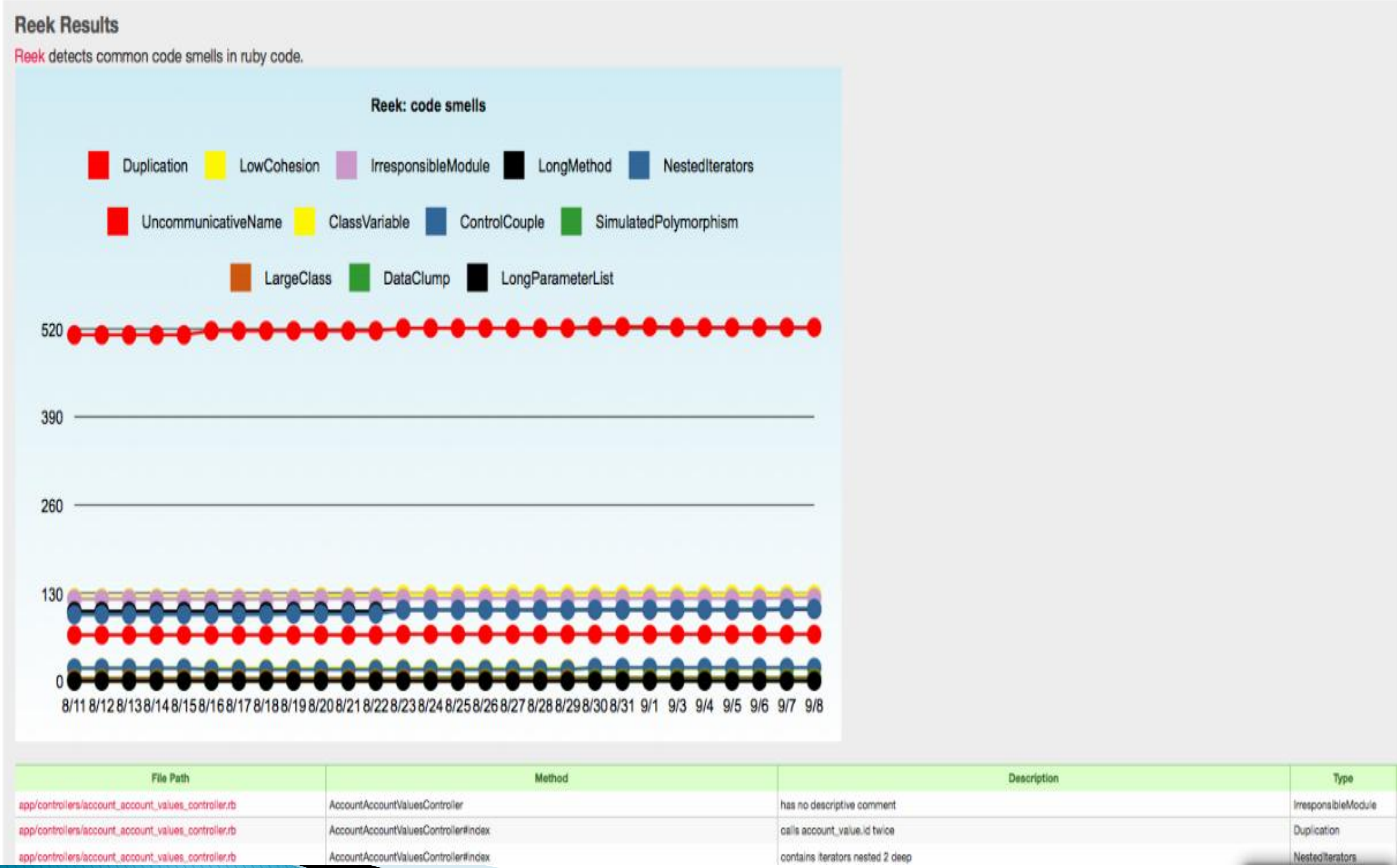


Reek Results

Reek detects common code smells in ruby code.

File Path	Method	Description
app/controllers/comments_controller.rb	CommentsController#create	calls flash multiple times
app/controllers/comments_controller.rb	CommentsController#create	calls params multiple times
app/controllers/comments_controller.rb	CommentsController#create	calls params[:preview_button] multiple times
app/controllers/comments_controller.rb	CommentsController#create	calls render("new") multiple times
app/controllers/comments_controller.rb	CommentsController#create	has approx 8 statements
app/controllers/comments_controller.rb	CommentsController#destroy	has approx 6 statements
app/controllers/comments_controller.rb	CommentsController#destroy/block/block	is nested
app/controllers/comments_controller.rb	CommentsController#update	calls params multiple times
app/controllers/episodes_controller.rb	EpisodesController#archive	calls params multiple times
app/controllers/episodes_controller.rb	EpisodesController#archive	calls params[:search] multiple times
app/controllers/episodes_controller.rb	EpisodesController#index	calls params multiple times

Reek (another ex.)

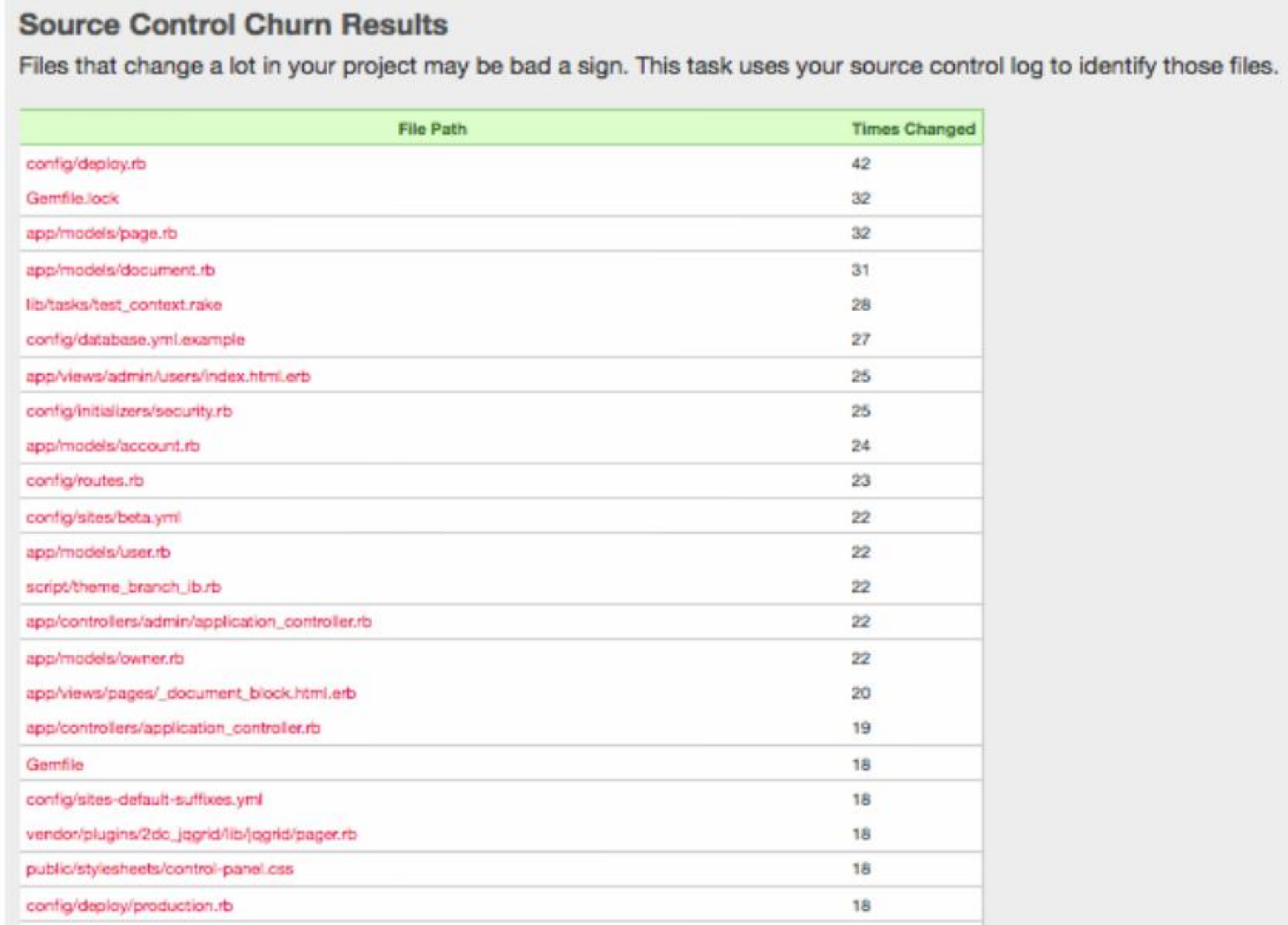


Reek Results

Reek detects common code smells in ruby code.

File Path	Method	Description	Type
app/controllers/account_account_values_controller.rb	AccountAccountValuesController	has no descriptive comment	IrresponsibleModule
app/controllers/account_account_values_controller.rb	AccountAccountValuesController#index	calls account_value.id twice	Duplication
app/controllers/account_account_values_controller.rb	AccountAccountValuesController#index	contains iterators nested 2 deep	NestedIterators

Churn

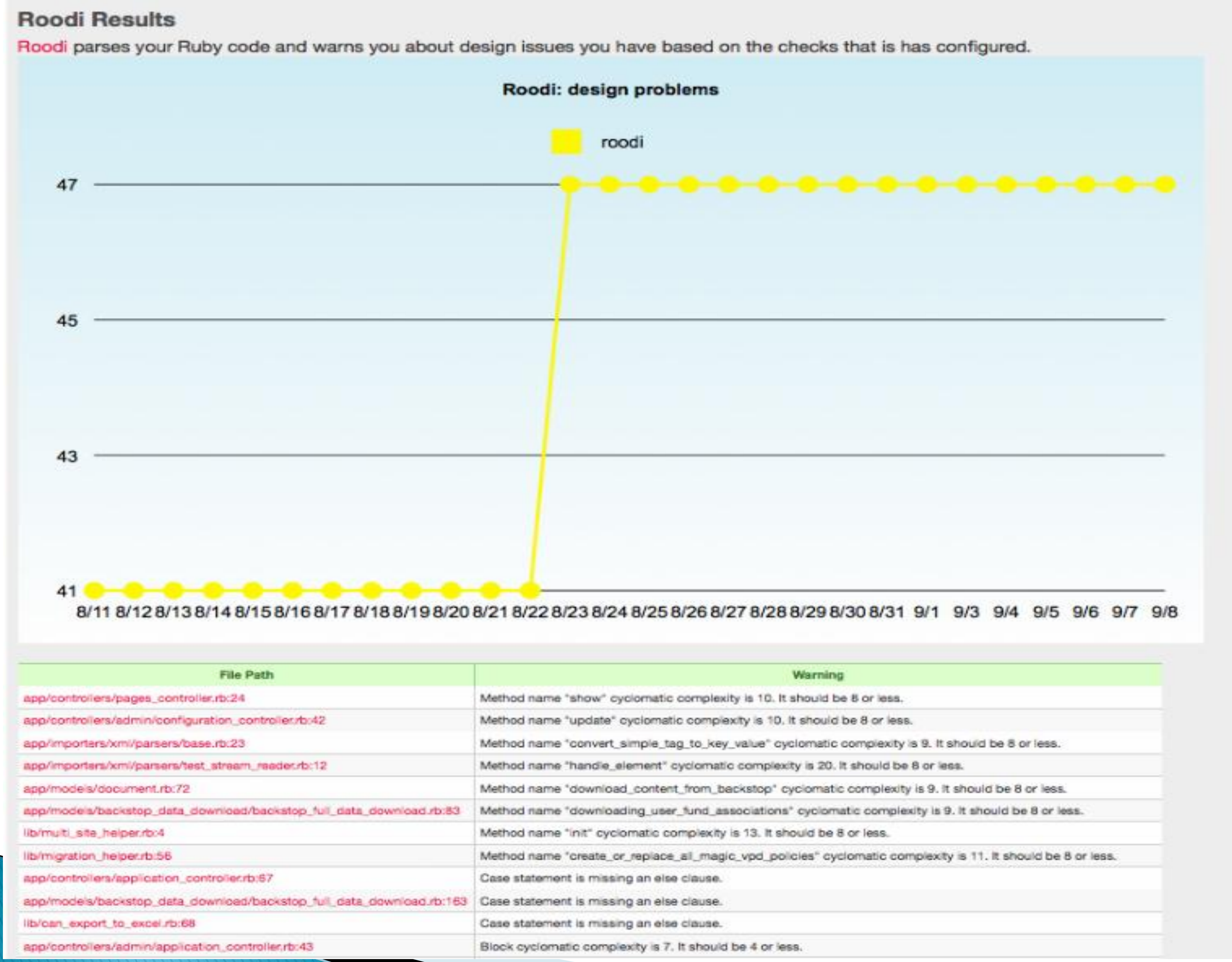


Source Control Churn Results

Files that change a lot in your project may be bad a sign. This task uses your source control log to identify those files.

File Path	Times Changed
config/deploy.rb	42
Gemfile.lock	32
app/models/page.rb	32
app/models/document.rb	31
lib/tasks/test_context.rake	28
config/database.yml.example	27
app/views/admin/users/index.html.erb	25
config/initializers/security.rb	25
app/models/account.rb	24
config/routes.rb	23
config/sites/beta.yml	22
app/models/user.rb	22
script/theme_branch_ib.rb	22
app/controllers/admin/application_controller.rb	22
app/models/owner.rb	22
app/views/pages/_document_block.html.erb	20
app/controllers/application_controller.rb	19
Gemfile	18
config/sites-default-suffixes.yml	18
vendor/plugins/2dc_jqgrid/lib/jqgrid/pager.rb	18
public/stylesheets/control-panel.css	18
config/deploy/production.rb	18

Roodi

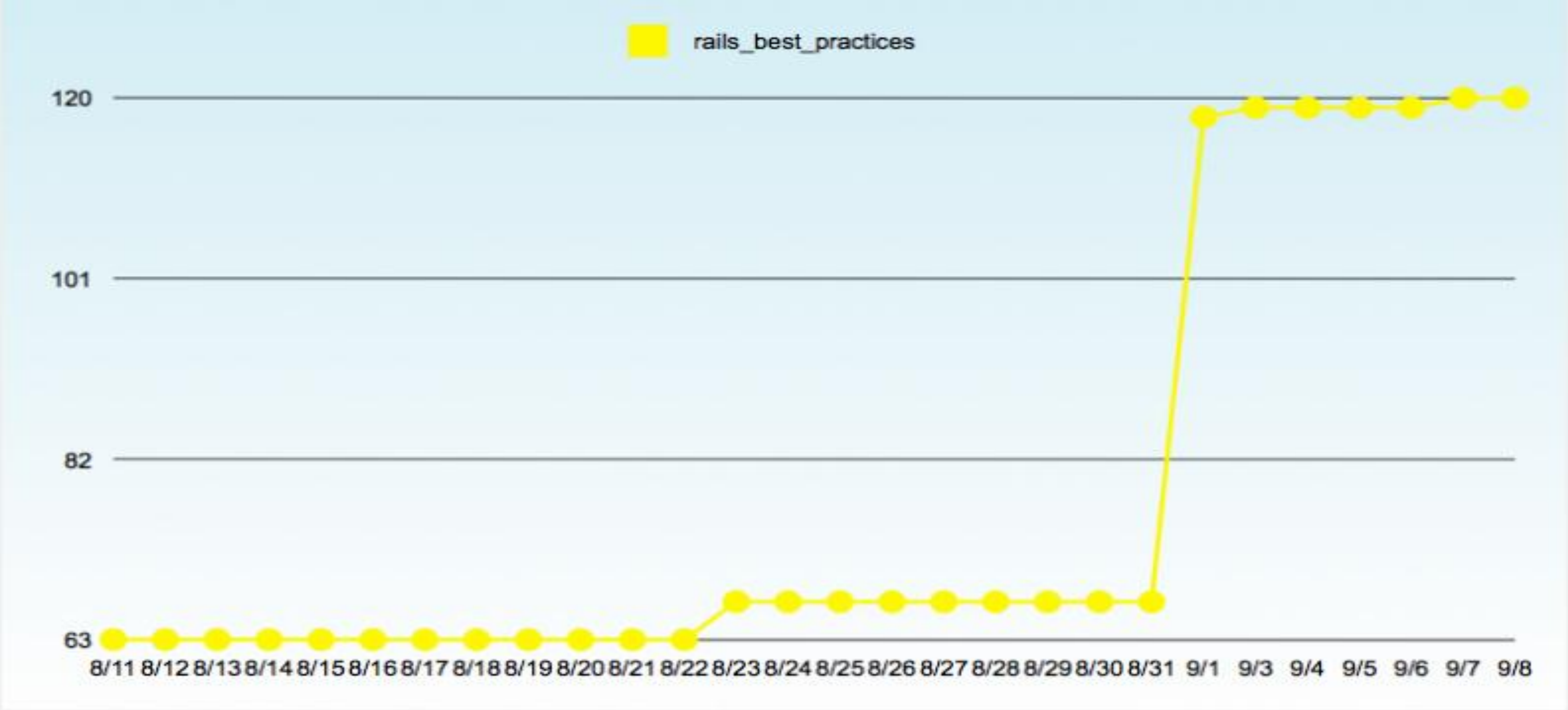


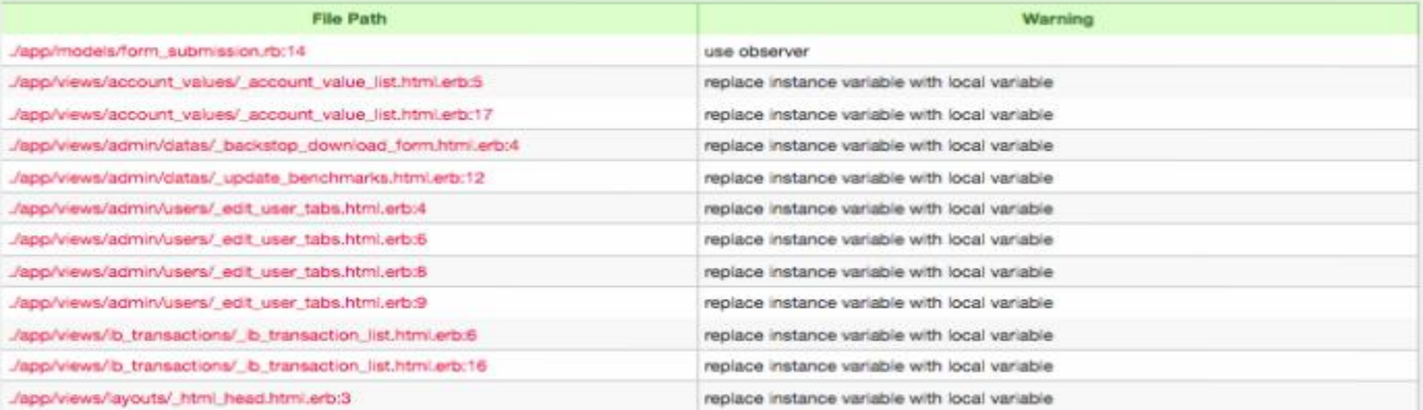
Roodi Results

Roodi parses your Ruby code and warns you about design issues you have based on the checks that is has configured.

File Path	Warning
app/controllers/pages_controller.rb:24	Method name "show" cyclomatic complexity is 10. It should be 8 or less.
app/controllers/admin/configuration_controller.rb:42	Method name "update" cyclomatic complexity is 10. It should be 8 or less.
app/importers/xml/parsers/base.rb:23	Method name "convert_simple_tag_to_key_value" cyclomatic complexity is 9. It should be 8 or less.
app/importers/xml/parsers/test_stream_reader.rb:12	Method name "handle_element" cyclomatic complexity is 20. It should be 8 or less.
app/models/document.rb:72	Method name "download_content_from_backstop" cyclomatic complexity is 9. It should be 8 or less.
app/models/backstop_data_download/backstop_full_data_download.rb:83	Method name "downloading_user_fund_associations" cyclomatic complexity is 9. It should be 8 or less.
lib/multi_site_helper.rb:4	Method name "init" cyclomatic complexity is 13. It should be 8 or less.
lib/migration_helper.rb:56	Method name "create_or_replace_all_magic_vpd_policies" cyclomatic complexity is 11. It should be 8 or less.
app/controllers/application_controller.rb:67	Case statement is missing an else clause.
app/models/backstop_data_download/backstop_full_data_download.rb:163	Case statement is missing an else clause.
lib/can_export_to_excel.rb:68	Case statement is missing an else clause.
app/controllers/admin/application_controller.rb:43	Block cyclomatic complexity is 7. It should be 4 or less.

Rails Best Practices

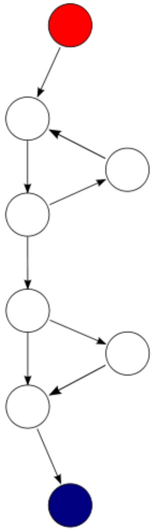




File Path	Warning
./app/models/form_submission.rb:14	use observer
./app/views/account_values/_account_value_list.html.erb:5	replace instance variable with local variable
./app/views/account_values/_account_value_list.html.erb:17	replace instance variable with local variable
./app/views/admin/datas/_backstop_download_form.html.erb:4	replace instance variable with local variable
./app/views/admin/datas/_update_benchmarks.html.erb:12	replace instance variable with local variable
./app/views/admin/users/_edit_user_tabs.html.erb:4	replace instance variable with local variable
./app/views/admin/users/_edit_user_tabs.html.erb:6	replace instance variable with local variable
./app/views/admin/users/_edit_user_tabs.html.erb:8	replace instance variable with local variable
./app/views/admin/users/_edit_user_tabs.html.erb:9	replace instance variable with local variable
./app/views/lb_transactions/_lb_transaction_list.html.erb:6	replace instance variable with local variable
./app/views/lb_transactions/_lb_transaction_list.html.erb:16	replace instance variable with local variable
./app/views/layouts/_html_head.html.erb:3	replace instance variable with local variable

Saikuro – CC calculation

- measures cyclomatic complexity (should be < 10)



E : # of edges on the graph

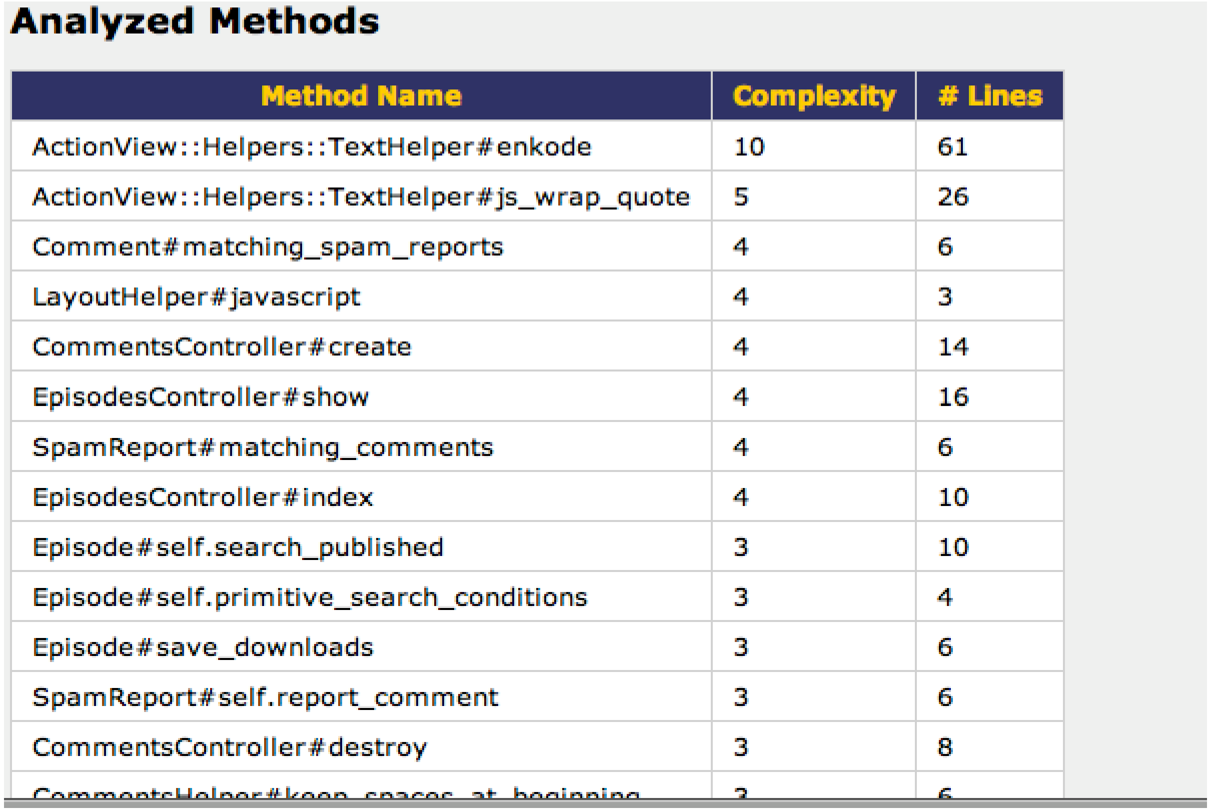
N : # nodes on the graph

P : # of connected parts

The complexity **M** is then defined as $M = E - N + 2P$, [2]

[2] : http://en.wikipedia.org/wiki/Cyclomatic_complexity

Saikuro(cont.)

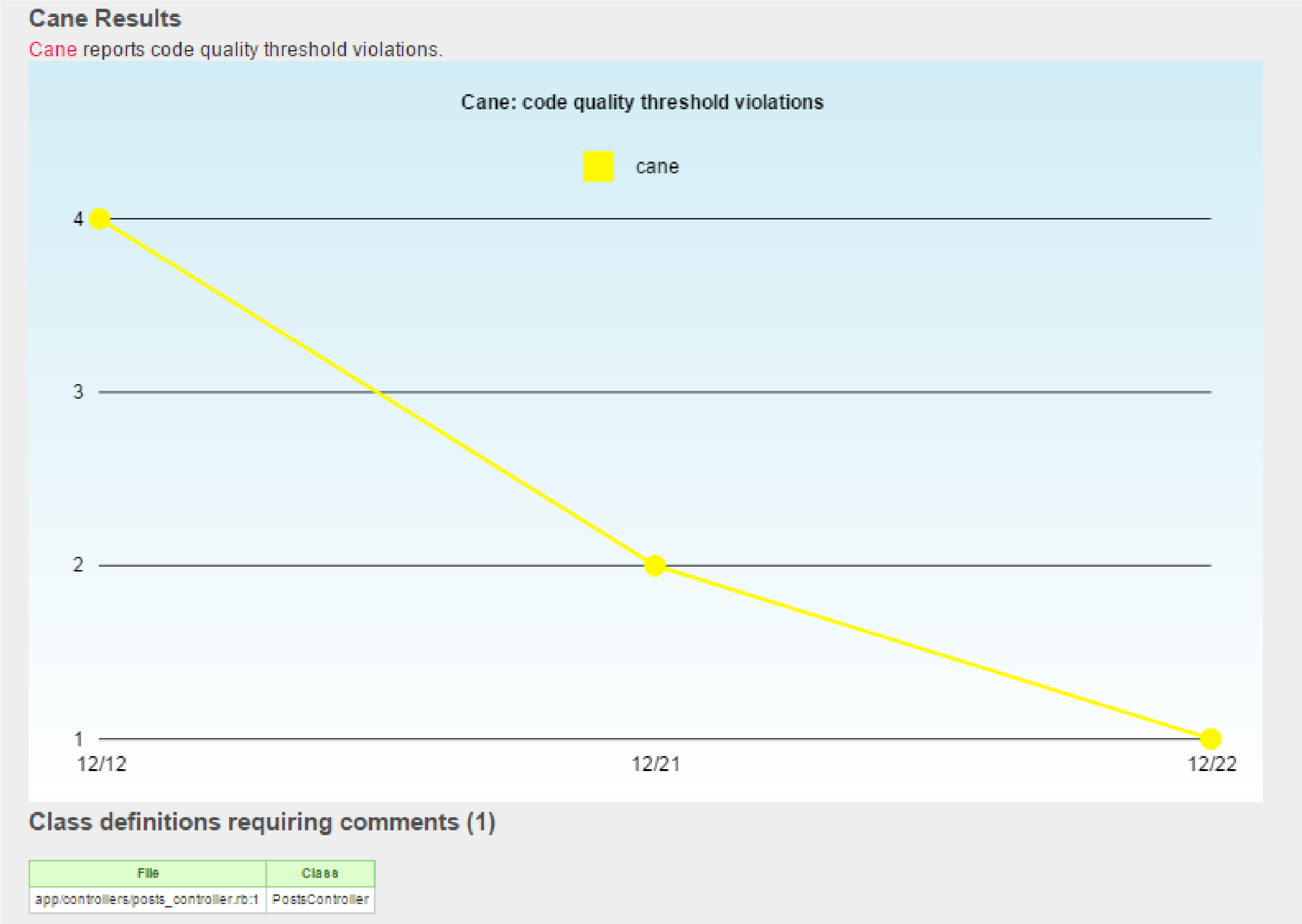


Analyzed Methods

Method Name	Complexity	# Lines
ActionView::Helpers::TextHelper#enkode	10	61
ActionView::Helpers::TextHelper#js_wrap_quote	5	26
Comment#matching_spam_reports	4	6
LayoutHelper#javascript	4	3
CommentsController#create	4	14
EpisodesController#show	4	16
SpamReport#matching_comments	4	6
EpisodesController#index	4	10
Episode#self.search_published	3	10
Episode#self.primitive_search_conditions	3	4
Episode#save_downloads	3	6
SpamReport#self.report_comment	3	6
CommentsController#destroy	3	8
CommentsHelper#keep_spaces_at_beginning	3	6

You do not need to refactor because complexity for each method <= 10

Cane



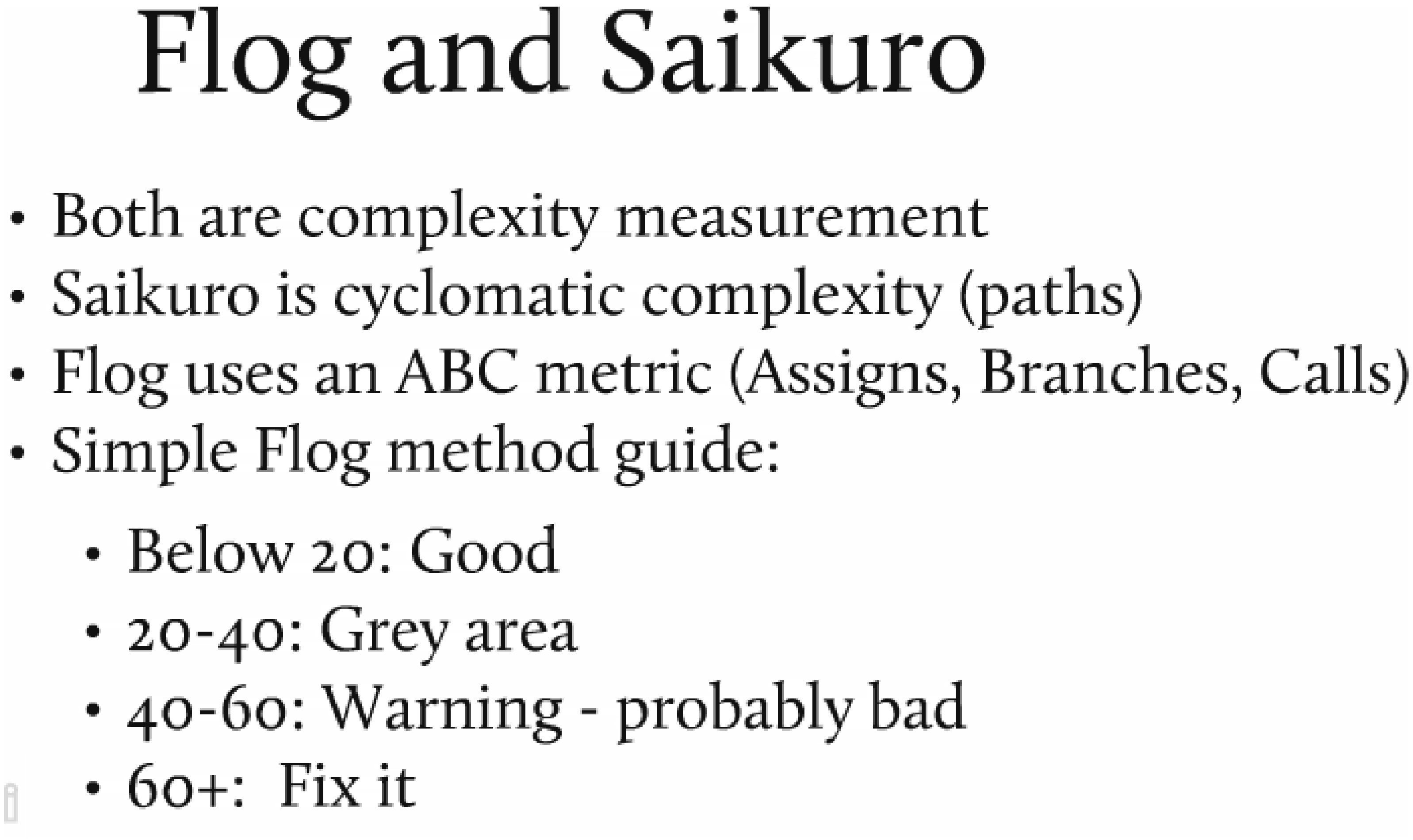
Cane Results

Cane reports code quality threshold violations.

Class definitions requiring comments (1)

File	Class
app/controllers/posts_controller.rb:1	PostsController

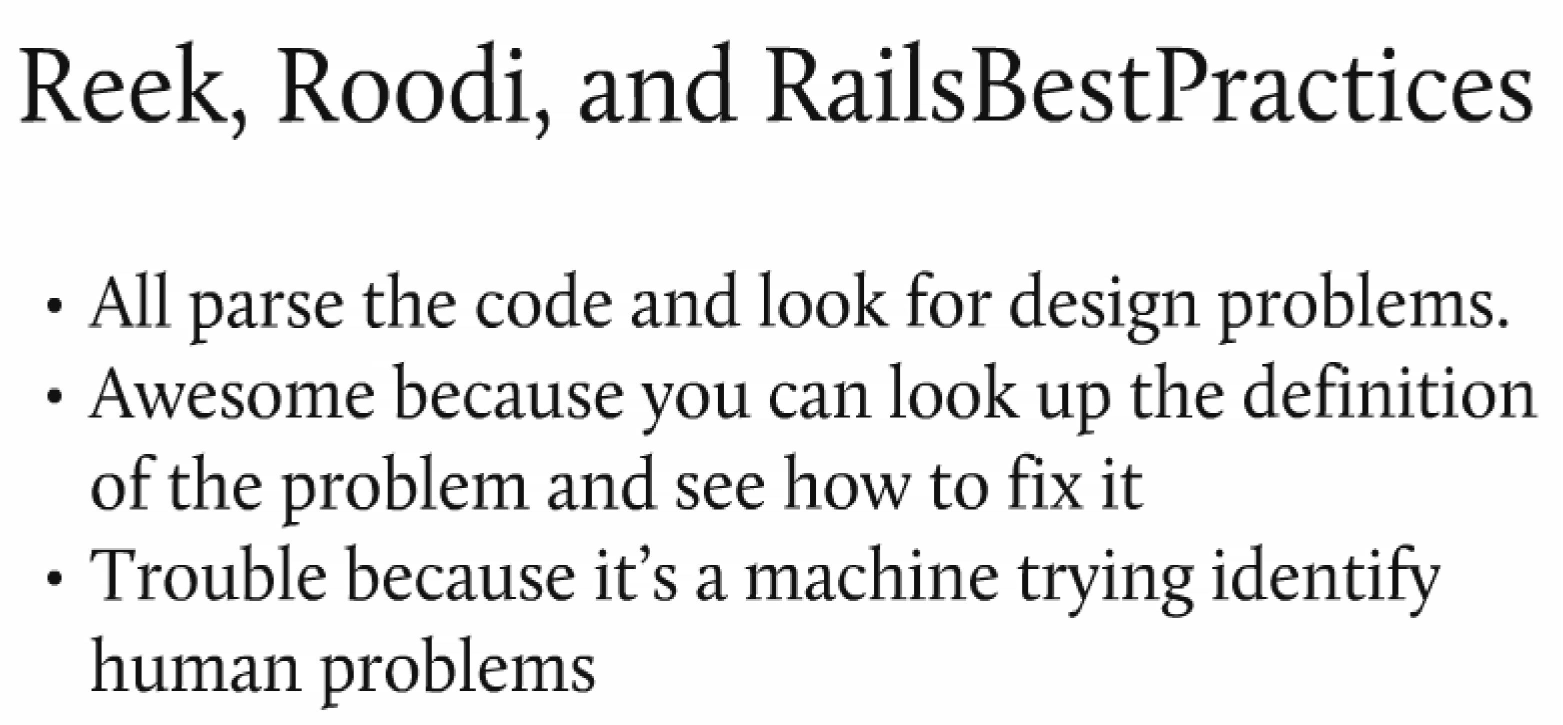
Over all:



Flog and Saikuro

- Both are complexity measurement
- Saikuro is cyclomatic complexity (paths)
- Flog uses an ABC metric (Assigns, Branches, Calls)
- Simple Flog method guide:
  - Below 20: Good
  - 20-40: Grey area
  - 40-60: Warning - probably bad
  - 60+: Fix it

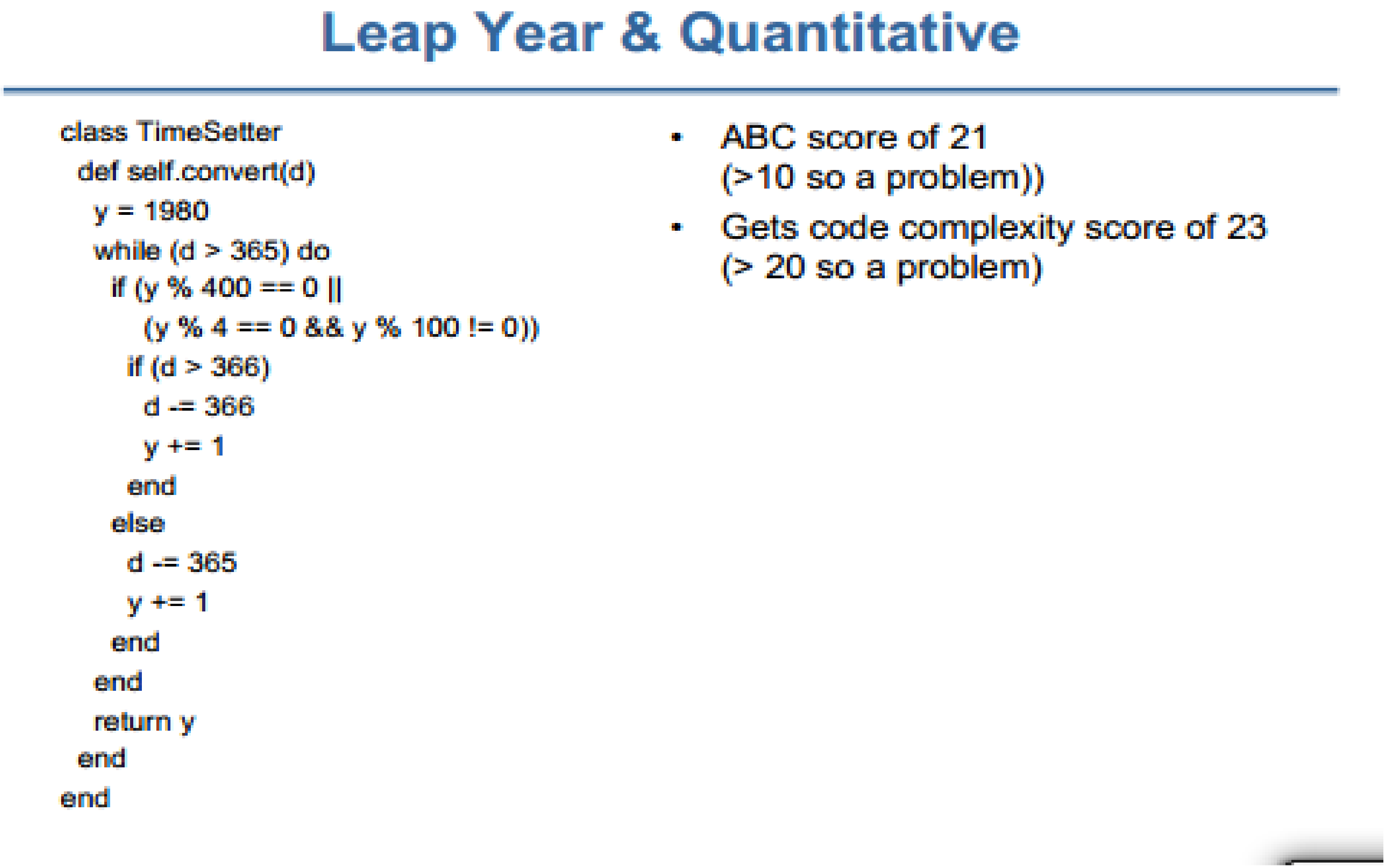
Over all (cont.)



Reek, Roodi, and RailsBestPractices

- All parse the code and look for design problems.
- Awesome because you can look up the definition of the problem and see how to fix it
- Trouble because it’s a machine trying identify human problems

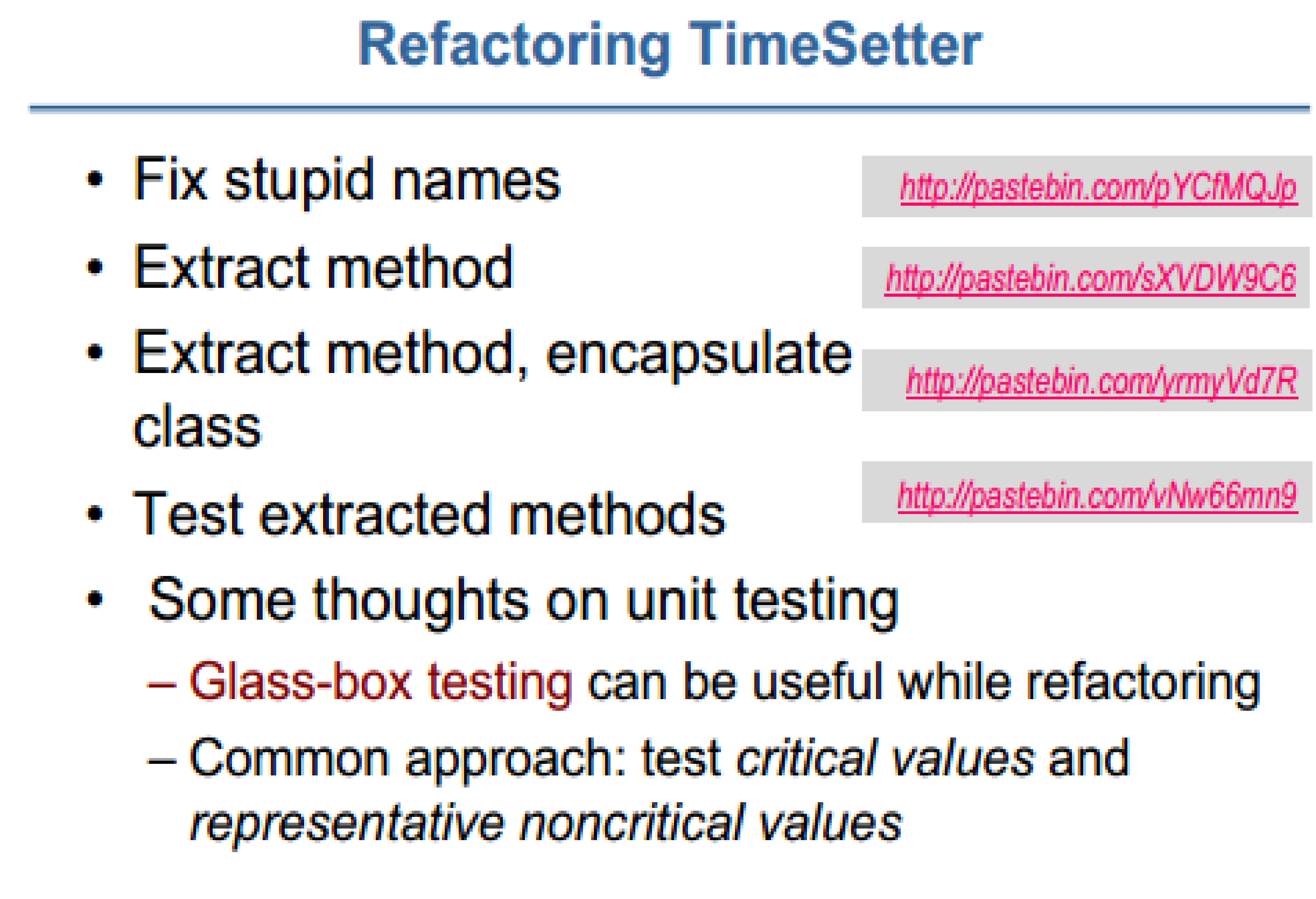
An Example from the Lecture Notes



Leap Year & Quantitative

```
class TimeSetter
  def self.convert(d)
    y = 1980
    while (d > 365) do
      if (y % 400 == 0 ||
          (y % 4 == 0 && y % 100 != 0))
        if (d > 366)
          d -= 366
          y += 1
        end
      else
        d -= 365
        y += 1
      end
    end
    return y
  end
end
```

- ABC score of 21
  (>10 so a problem))
- Gets code complexity score of 23
  (> 20 so a problem)

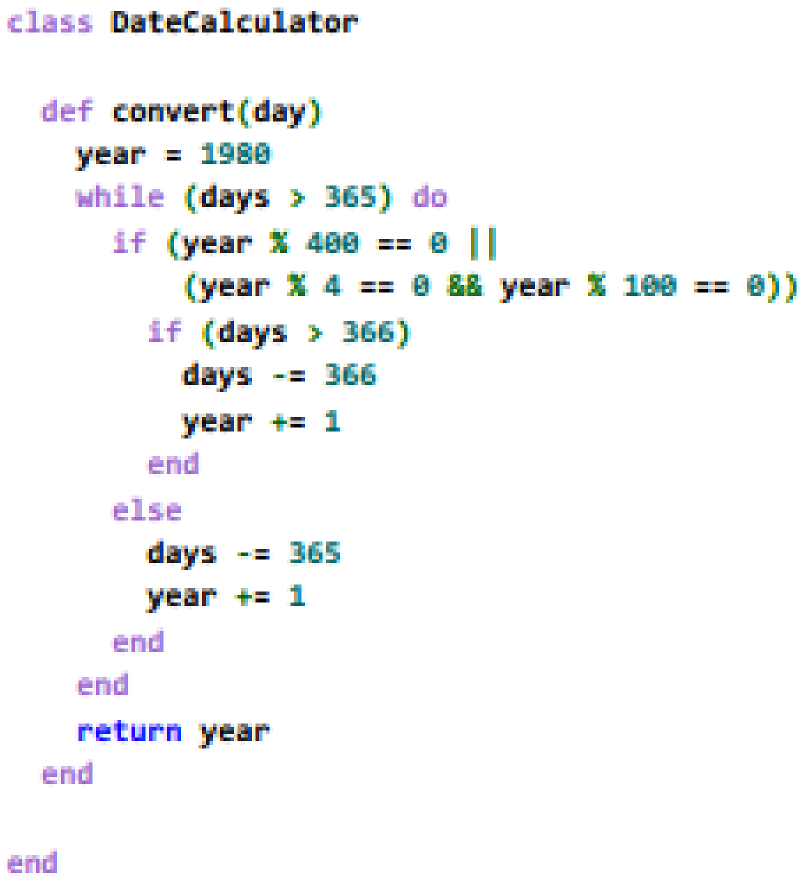


Refactoring TimeSetter

- Fix stupid names http://pastebin.com/pYCfMQJp
- Extract method http://pastebin.com/sXVDW9C6
- Extract method, encapsulate class http://pastebin.com/yrmyVd7R
- Test extracted methods http://pastebin.com/vNw66mn9
- Some thoughts on unit testing
  - Glass-box testing can be useful while refactoring
  - Common approach: test *critical values* and *representative noncritical values*

Fix Stupid Names

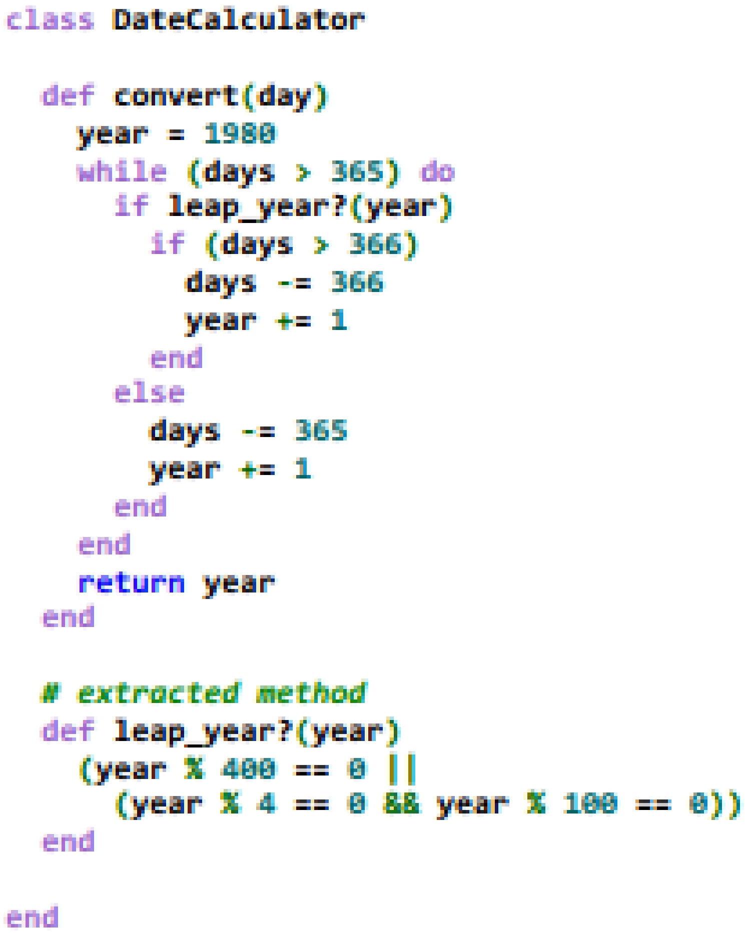
http://pastebin.com/sXVDW9C6



```
class DateCalculator

  def convert(day)
    year = 1980
    while (days > 365) do
      if (year % 400 == 0 ||
          (year % 4 == 0 && year % 100 == 0))
        if (days > 366)
          days -= 366
          year += 1
        end
      else
        days -= 365
        year += 1
      end
    end
    return year
  end

end
```

Extract Method (1/2)

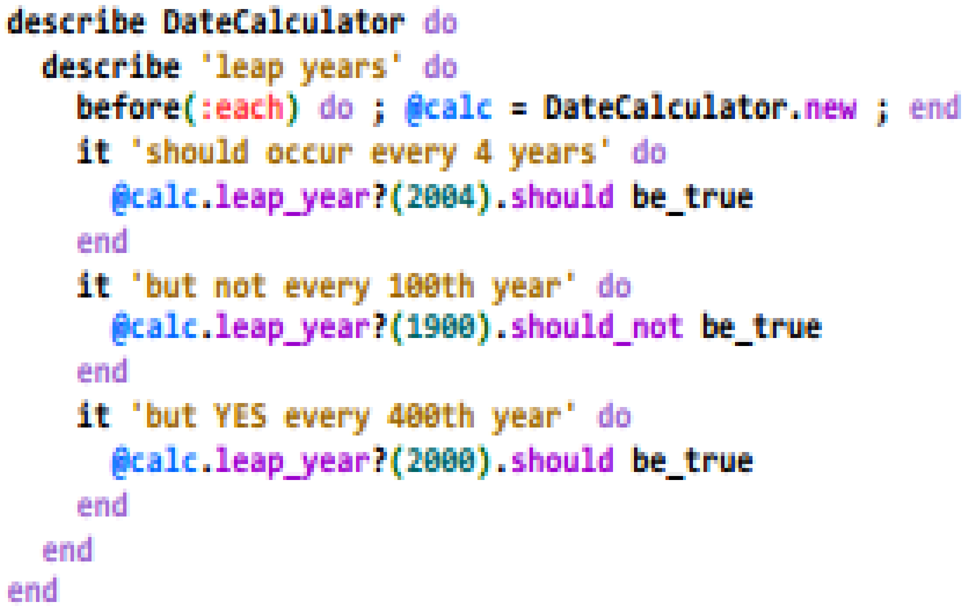
http://pastebin.com/pYCfMQJp



```
class DateCalculator

  def convert(day)
    year = 1980
    while (days > 365) do
      if leap_year?(year)
        if (days > 366)
          days -= 366
          year += 1
        end
      else
        days -= 365
        year += 1
      end
    end
    return year
  end

  # extracted method
  def leap_year?(year)
    (year % 400 == 0 ||
      (year % 4 == 0 && year % 100 == 0))
  end

end
```

Extract Method (2/2)

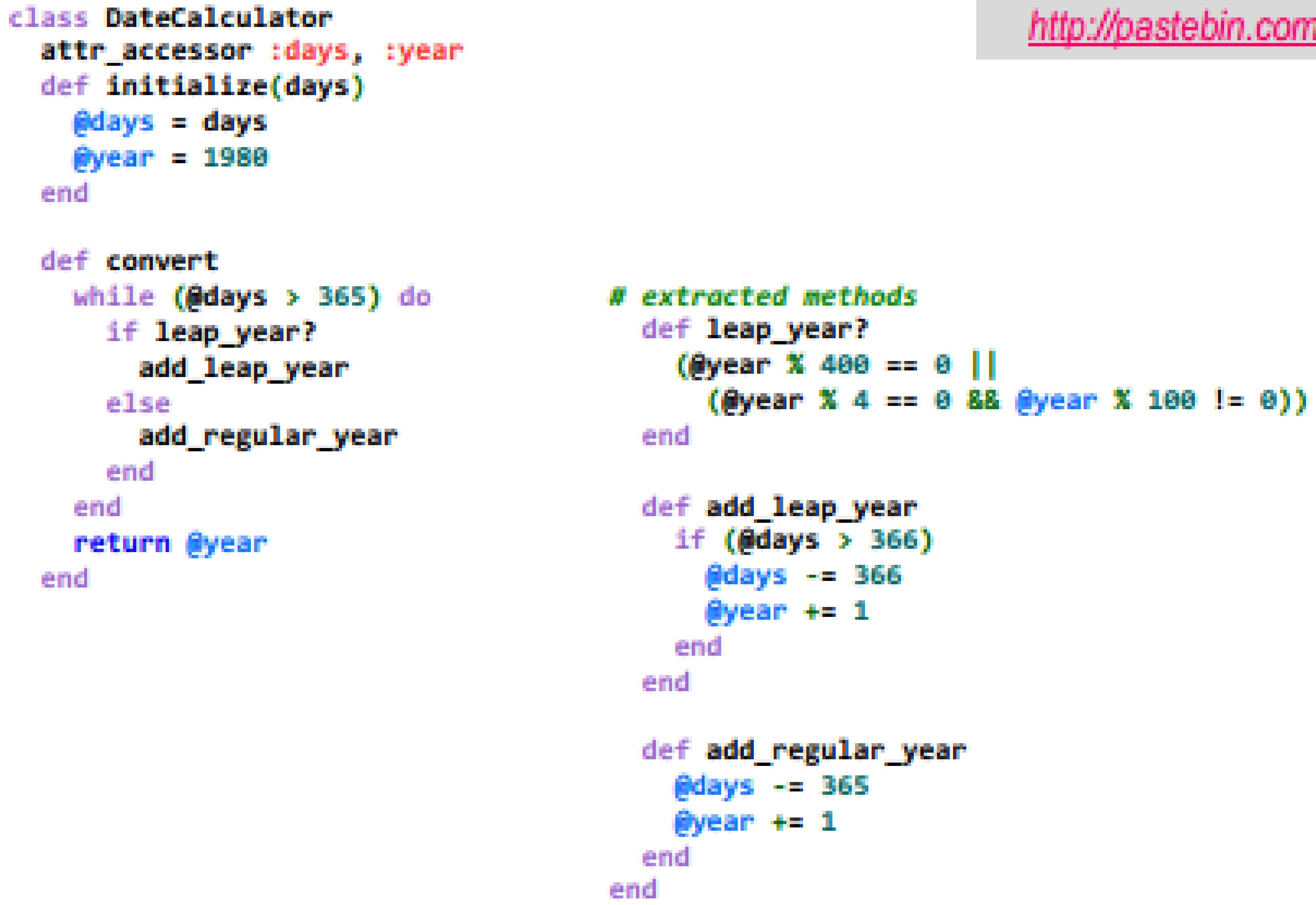
http://pastebin.com/pYCfMQJp



```
describe DateCalculator do
  describe 'leap years' do
    before(:each) do ; @calc = DateCalculator.new ; end
    it 'should occur every 4 years' do
      @calc.leap_year?(2004).should be_true
    end
    it 'but not every 100th year' do
      @calc.leap_year?(1900).should_not be_true
    end
    it 'but YES every 400th year' do
      @calc.leap_year?(2000).should be_true
    end
  end
end
```

Extract Method, Encapsulate Class

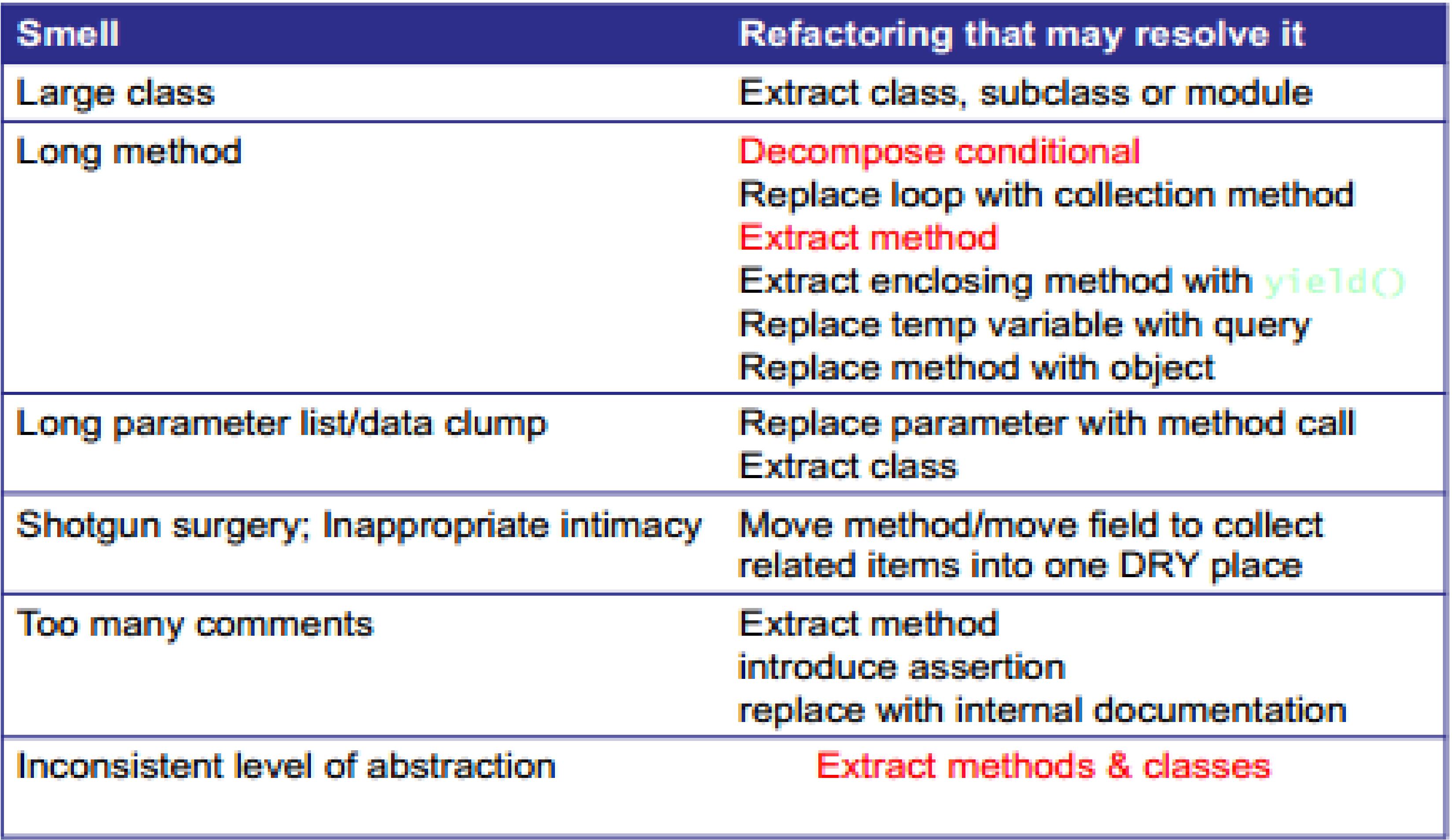
http://pastebin.com/ymyVd7R



```
class DateCalculator
  attr_accessor :days, :year
  def initialize(days)
    @days = days
    @year = 1980
  end

  def convert
    while (@days > 365) do
      if leap_year?
        add_leap_year
      else
        add_regular_year
      end
    end
    return @year
  end
```

```
# extracted methods
  def leap_year?
    (@year % 400 == 0 ||
      (@year % 4 == 0 && @year % 100 != 0))
  end

  def add_leap_year
    if (@days > 366)
      @days -= 366
      @year += 1
    end
  end

  def add_regular_year
    @days -= 365
    @year += 1
  end
end
```

Other Smells & Remedies



Smell	Refactoring that may resolve it
Large class	Extract class, subclass or module
Long method	Decompose conditional Replace loop with collection method Extract method Extract enclosing method with `yield()` Replace temp variable with query Replace method with object
Long parameter list/data clump	Replace parameter with method call Extract class
Shotgun surgery; Inappropriate intimacy	Move method/move field to collect related items into one DRY place
Too many comments	Extract method introduce assertion replace with internal documentation
Inconsistent level of abstraction	Extract methods & classes

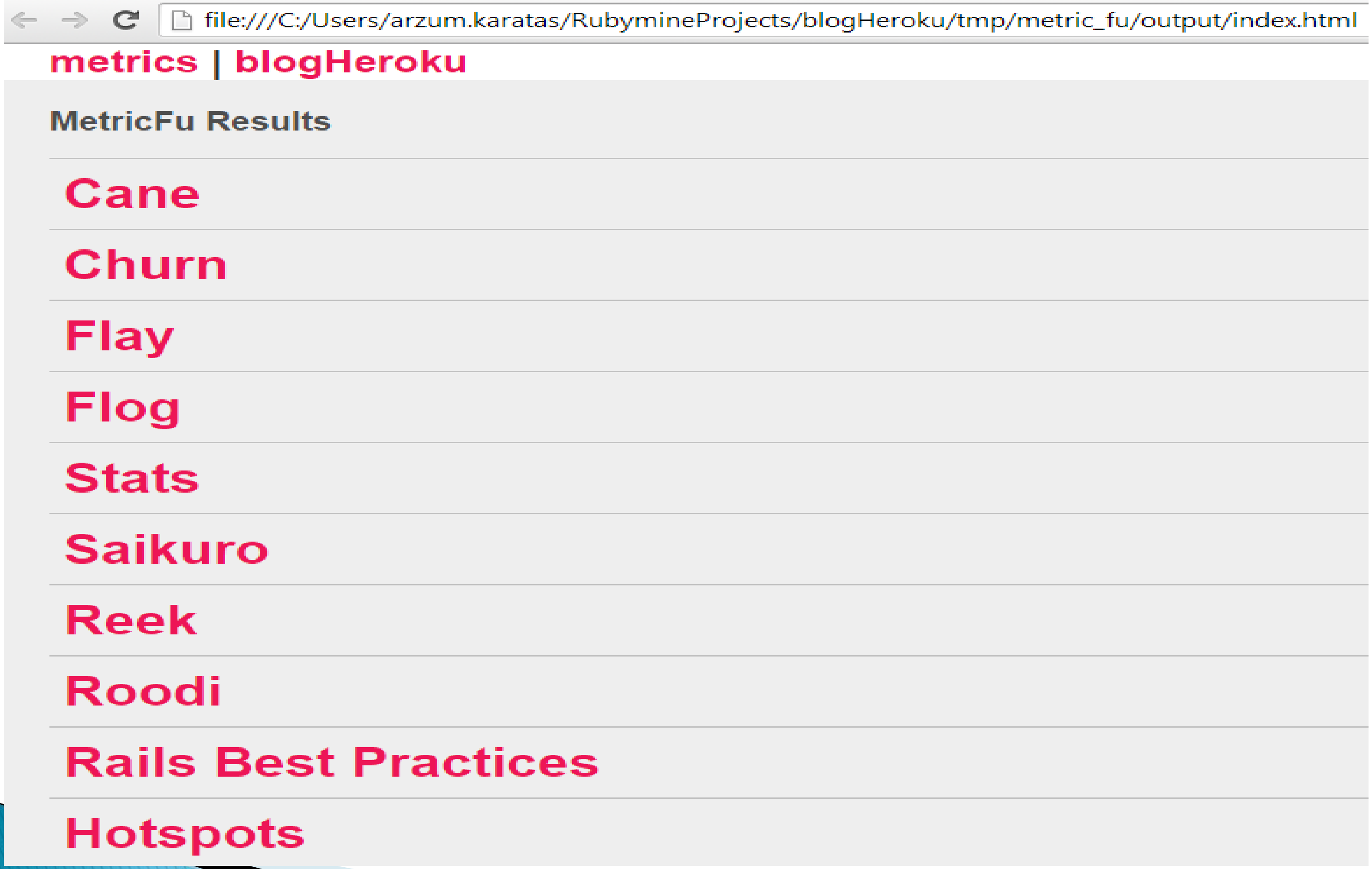
Hint : Refactoring

- You can read the following article to see how to do refactoring in RubyMine. (Optional)

http://pivotallabs.com/automated-refactorings-in-rubymine/

Wet your feet!

- Examine the blogHeroku Project



file:///C:/Users/arzum.karatas/RubymineProjects/blogHeroku/tmp/metric_fu/output/index.html

metrics | blogHeroku

MetricFu Results

- Cane
- Churn
- Flay
- Flog
- Stats
- Saikuro
- Reek
- Roodi
- Rails Best Practices
- Hotspots

Useful Links

- https://prezi.com/iyioiddjyccu/metrics-based-refactoring-what-to-do-with-your-code-metrics/

- http://www.confreaks.com/videos/377-rubyconf2010-hotspots-with-metric-fu

- http://asciicasts.com/episodes/166-metric-fu

QUESTIONS
And
Answers

Project Evaluation Session

Thank you