

VML "Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

Photo: El Caracol by Tinker & Rove

Goal: understand; explain risk, debt, complexity; empathy; non-technical sponsors;
[El Caracol, Tinker & Rove](#)

TITANIUM SPONSORS

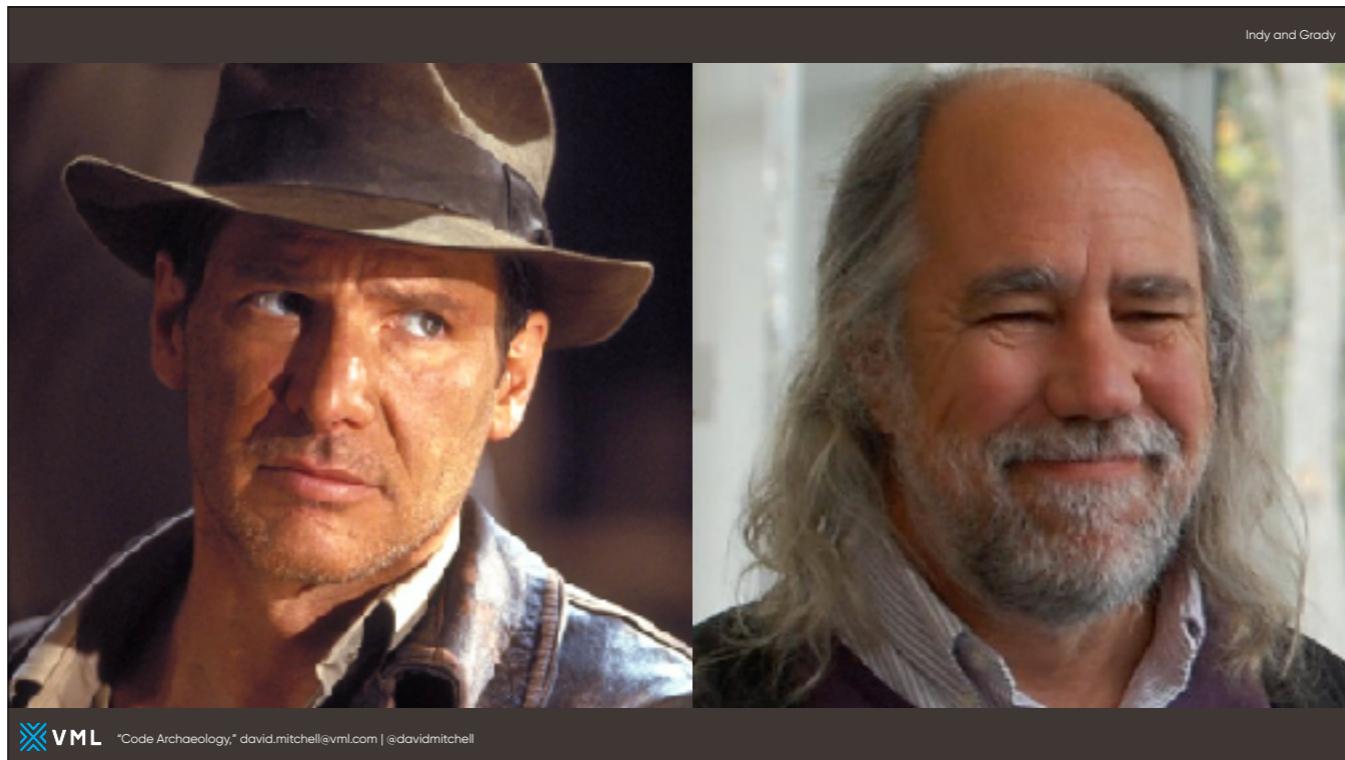


Platinum Sponsors



Gold Sponsors

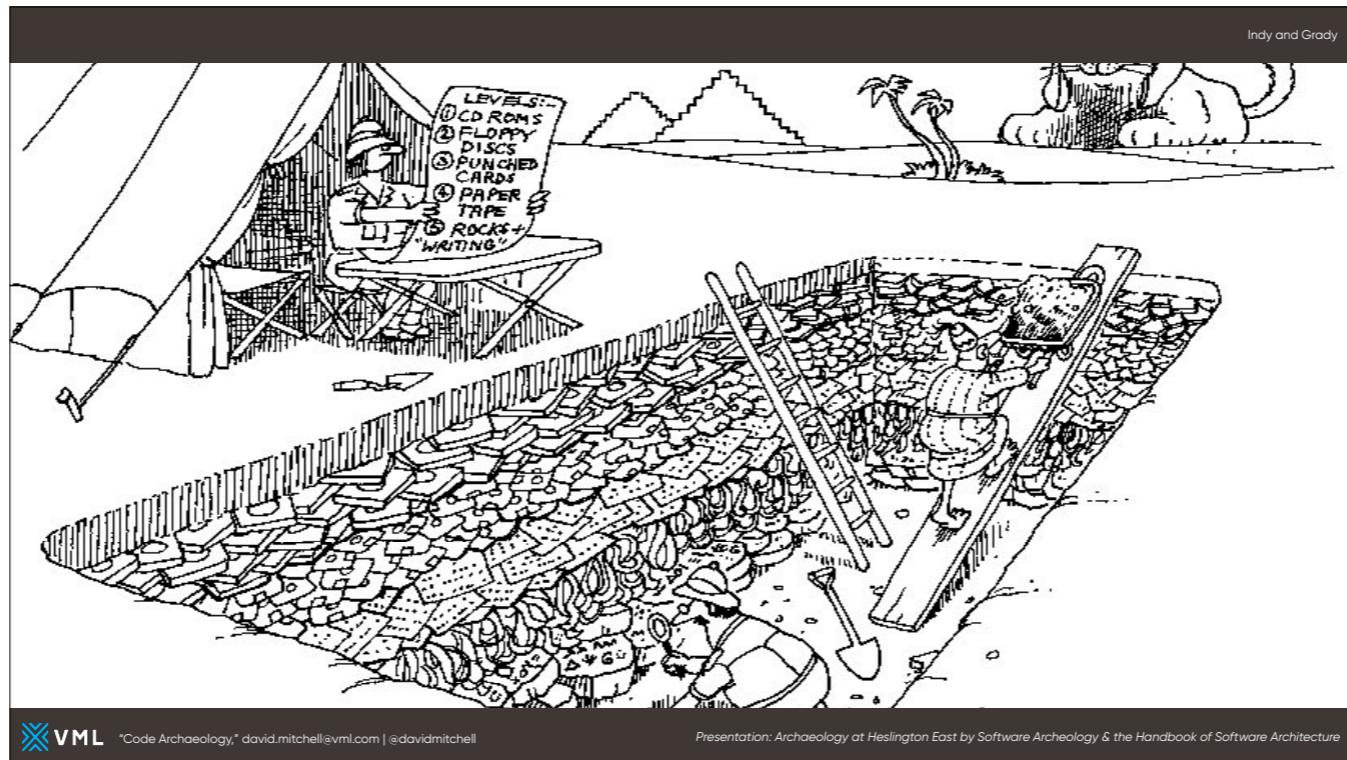




 VML

"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

Booch OMT one of 3 Rational founders, leads to UML, popularized archaeology for code



Grady Booch, IBM Fellow
IBM Research
Software Archeology & the Handbook of Software Architecture
© 2008, Grady Botch

<https://fg-sre.gi.de/fileadmin/gliederungen/fg-sre/wsr2008/booch.pdf>



Inherit some legacy code. Inherit legacy, JPMorgan, Why devalued, undervalued, misunderstood? See also: Legacy Code Rocks podcast

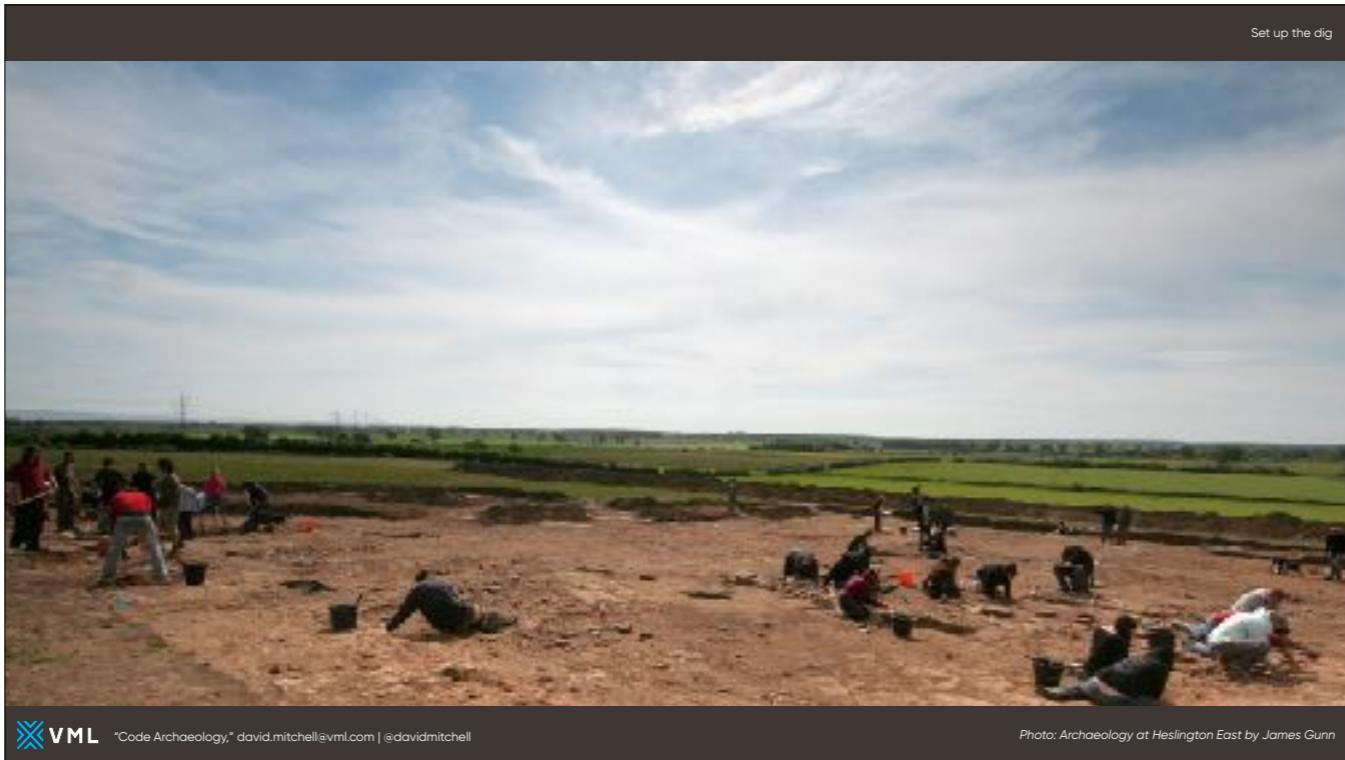
Image: [Trashing old software, jm3](#)

Clone, build, run the tests



Clone, build, run the tests

Book: [Continuous Delivery, Jez Humble & David Farley](#)



Set up the dig:

If you don't have a full devops ci pipeline, you can still have your own version control. Put it in a folder in Dropbox and use a local git repo to track any changes.
SE Radio 148: (Prag) Dave Thomas <http://www.se-radio.net/2009/11/episode-148-software-archaeology-with-dave-thomas/>

Image

[Archaeology at Heslington East, James Gunn](#), used under <https://creativecommons.org/licenses/by-sa/2.0/>,

```
55 \rightskip=0pt % this makes equal spacing
56 {\vtopskip 9pc % this makes equal spacing
57 \vskip-\parskip
58 \tenpoint
59 \noindent\hang\hangafter-2
60 \smash{\lower12pt\hbox to 0pt{\vskip-\hangindent\cmmn G\hfill}}}\vskip-16pt
61 {\sc ENTLE} R{\sc EADER}: \strut This is a handbook about
62 \TeX, a new typesetting system intended for the creation
63 of beautiful books—and especially for books that contain a lot of
64 mathematics. By preparing a manuscript in \TeX\ format, you will be
65 telling a computer exactly how the manuscript is to be transformed into
66 pages whose typographic quality is comparable to that of the world's
67 finest printers; yet you won't need to do much more work than would be
68 involved if you were simply typing the manuscript on an ordinary
69 typewriter. In fact, your total work will probably be significantly less,
70 if you consider the time it ordinarily takes to revise a typewritten manuscript,
71 since computer text files are so easy to change and to reprocess. \
72 (If such claims sound too good to be true, keep in mind that they were made
73 by \TeX's designer, on a day when \TeX\ happened to
74 be working, so the statements may be biased; but read on anyway.)
75 -- David Mitchell, who have never used \TeX\ before, as
```



"Code Archaeology," david.mitchell@vml.com | @davidmitchell

Read the fine manual

**Tudor Girba**

@girba

"Developers spend some half of their time reading code." Do you agree?



"Code Archaeology," david.mitchell@vml.com | @davidmitchell

Slideshare: *The Glamorous Toolkit: Towards a novel live IDE*

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)

Reading

Tudor Girba
@girba

"Developers spend some half of their time reading code." Do you agree?

84% Yes

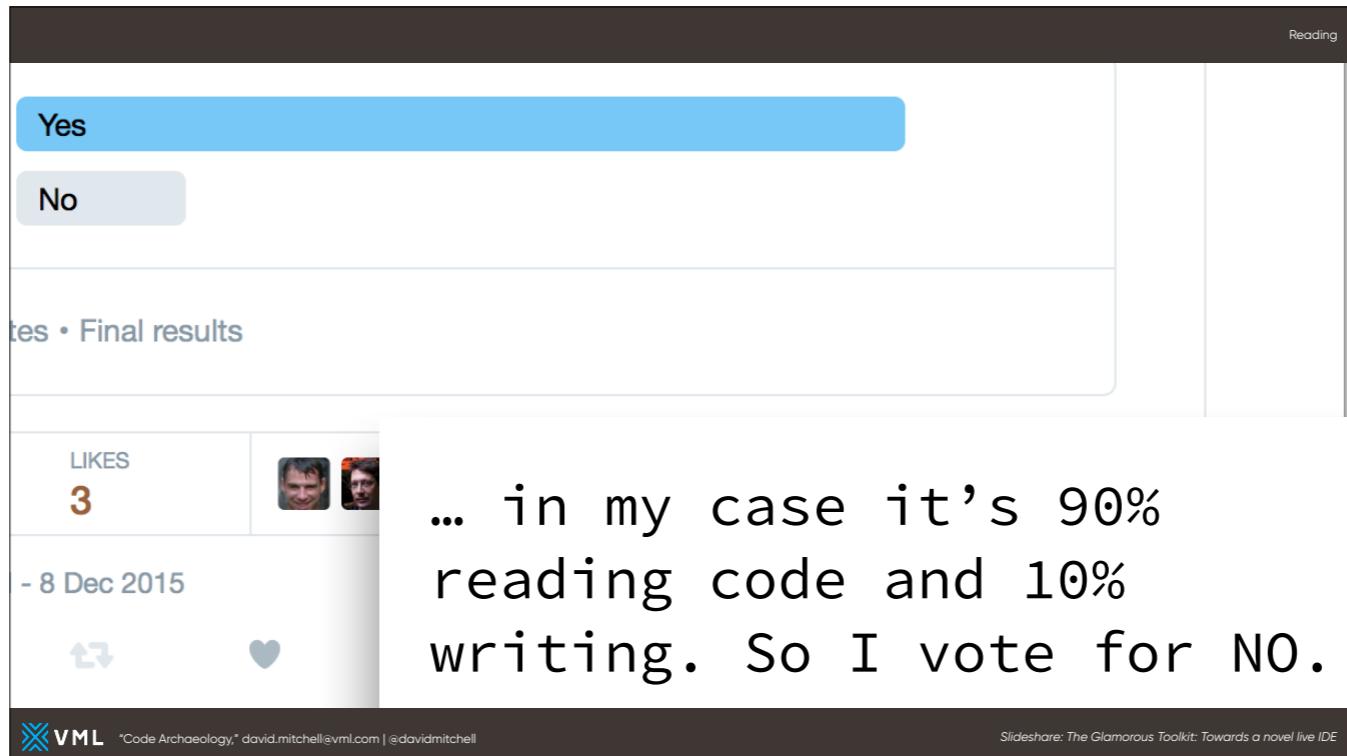
16% No

32 votes • Final results

VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

Slideshare: The Glamorous Toolkit: Towards a novel live IDE

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



Code Reading

https://books.google.com/books?id=pqw2U-II5lwC&pg=PA345&lpg=PA345&dq=ldaptransformer+microsoft+word&source=bl&ots=a0U9mo_jiQ&sig=mJ0GcCWKqFv3DWdHcZuOurV2_3U&hl=en&sa=X&ved=0ahUKEwiTxNil6ZrUAhUmhIQKHXdFAMkQ6AEIKjAA#v=onepage&q=ldaptransformer%20microsoft%20word&f=false

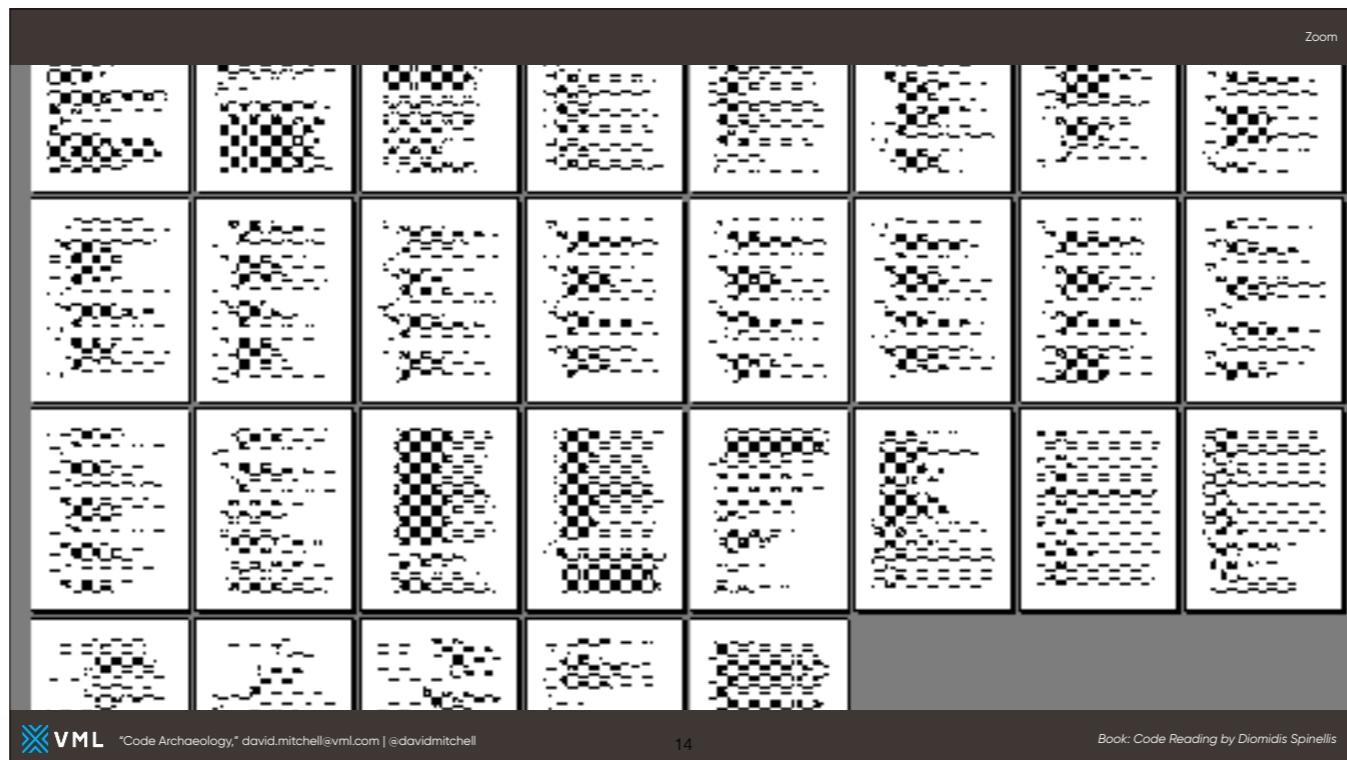
Book: Code Reading by Diomidis Spinellis

library inclusion

Main

Open source vintage: CVS (source forge), SVN (was any dominant open source repo using SVN), and today git (GitHub).

Code reading chapter 1 is focused on very fine grain source reading (line at a time) sort of like a 'be the compiler' exercise in a Head First book.



VML "Code Archaeology," david.mitchell@vml.com | [@davidmitchell](#)

14

Book: *Code Reading* by Diomidis Spinellis

From *Code Reading, the Open Source Perspective*: <https://www.amazon.com/Code-Reading-Open-Source-Perspective/dp/0201799405>
Diomidis Spinellis

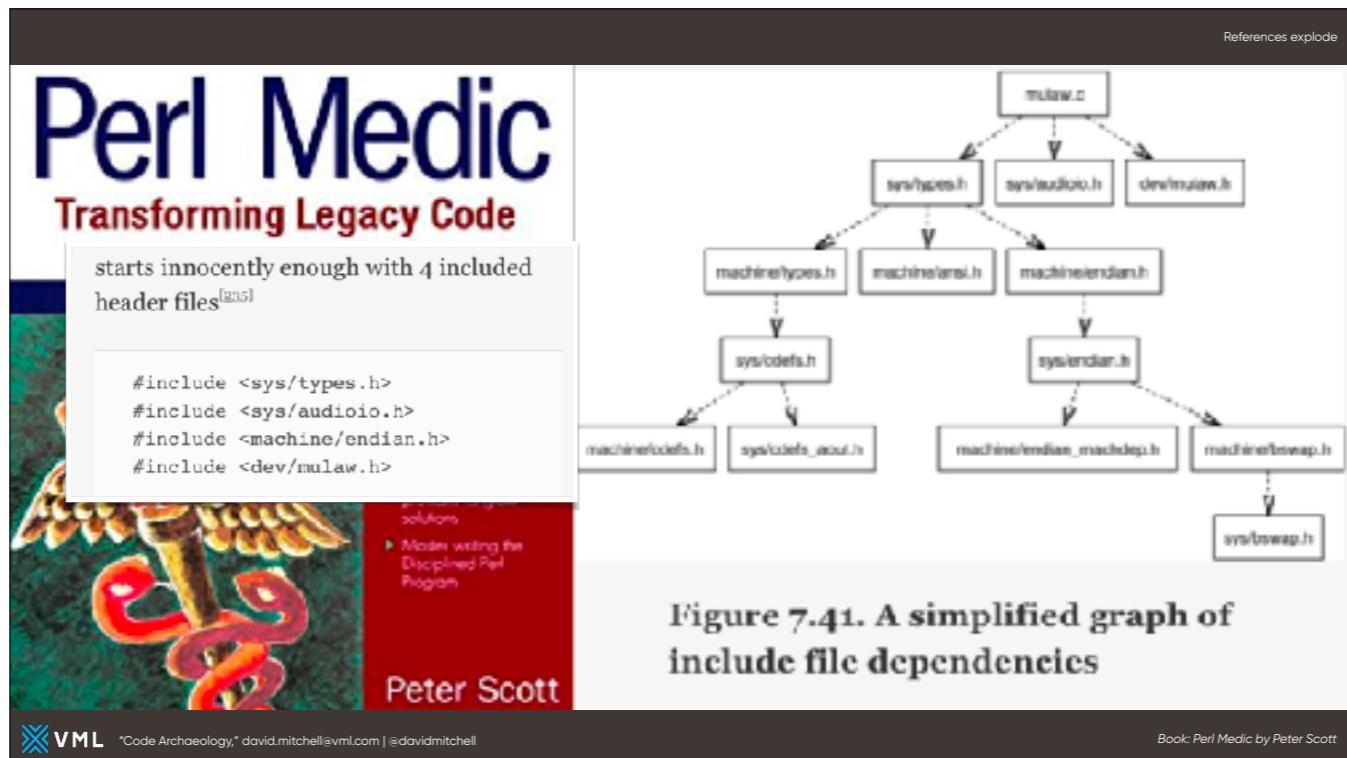
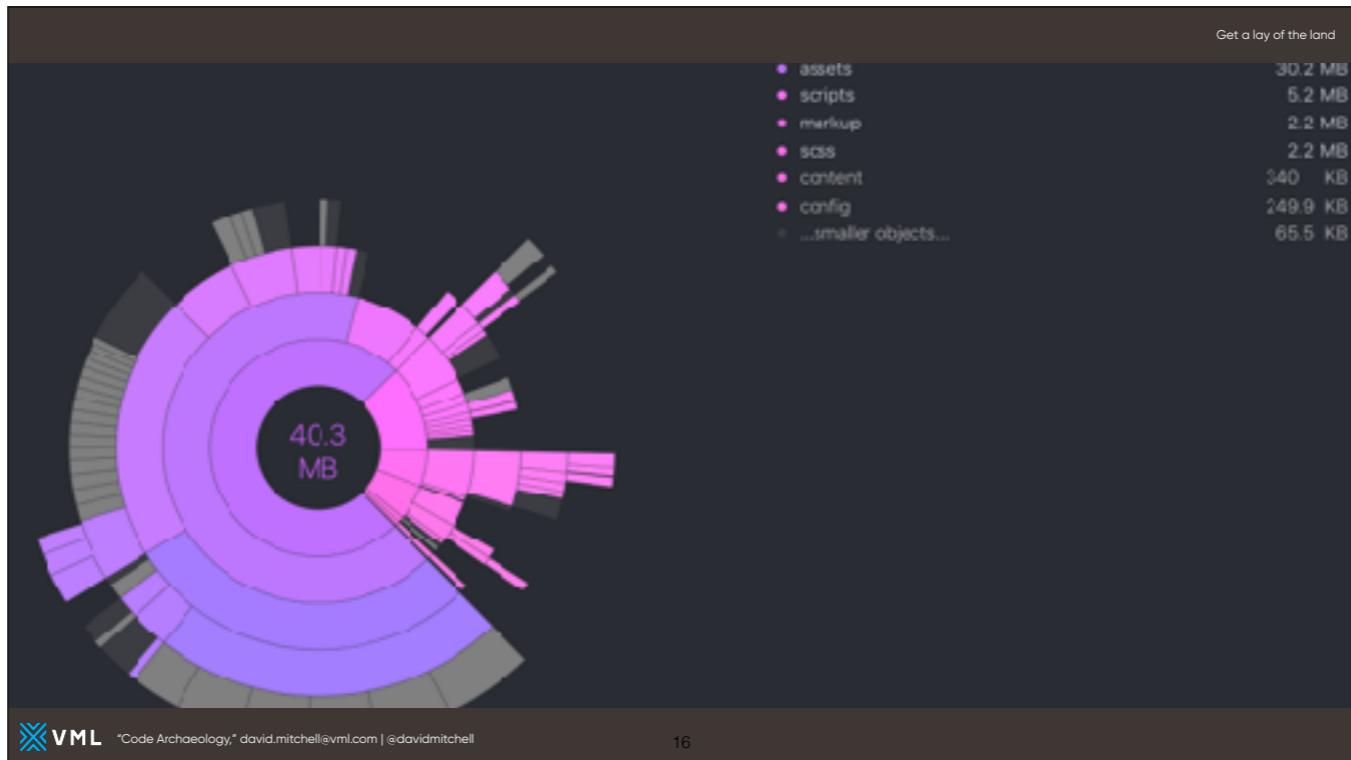
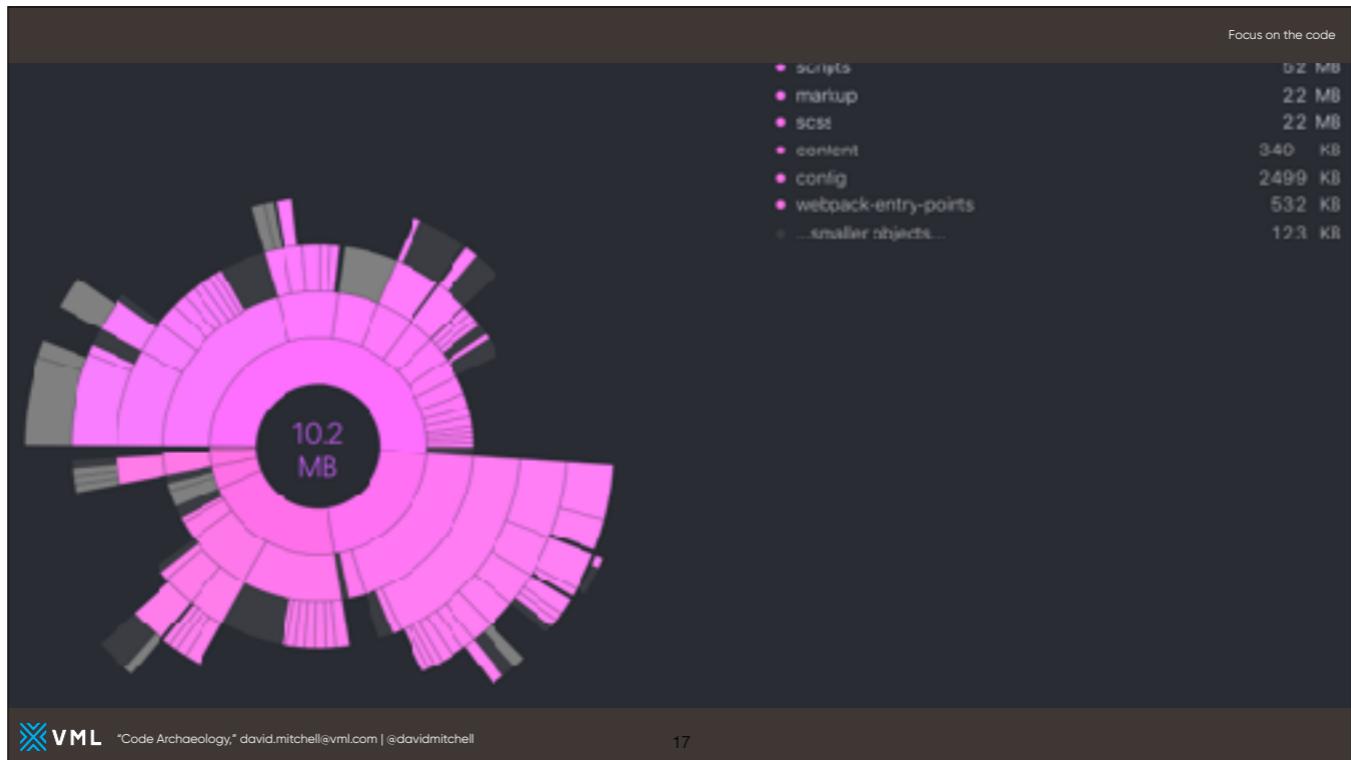


Figure 7.41. A simplified graph of include file dependencies

Book: Perl Medic by Peter Scott



War and peace is 3MB



scripts: javascript

markup: pure components (later slides show extension and inclusion)

scss: sass/css

content: external application labels (easy to translate)

config: ?

webpack-entry-points: ?

GREP (v4 page 24)

Grep, generally cited as the prototypical software tool, was born when Ken Thompson was asked how to search for patterns in a file but was too big for the office. Thompson promptly liberated his regular expression engine and obfuscated it after the *of* command *g/v/v/p*. It was an instant hit and soon entered our language as a verb.

The success of grep suggested other utilities. One of these, *grep* for global substitution, evolved into the stream editor *ed* [McMahon, v7]. Grep also inspired Al Aho to apply his encyclopedic knowledge of language theory to make the very general *egrep* and the highly specialized *fgrep* (v7). Over the years Aho honed grep into an awesomely performing program, simple on the outside but the highest of tech on the inside.

-8-

The v4 manual page for *grep* is typical of earlier editions. Options were described in running text along with the basic usage; the more readable convention of always displaying options separately, no matter how few a command might have, gained ground only slowly. In typical unix style the synopsis permits an output argument as well as an input argument—poor human engineering, because of the disastrous consequences of invoking command *input output* as if *view command input1 input2*. This disastrous syntax was purged from most commands by v4, but at least one fossilized instance survives in BSD 4.3.

The image shows two book covers side-by-side. The left book is titled 'A Research UNIX Reader' by Douglas McIlroy. The right book is titled 'Stranger in a Strange Land' by Robert Heinlein. Both books have dark, textured covers.

VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

Books: *A Research UNIX Reader* by Douglas McIlroy and *Stranger in a Strange Land* by Robert Heinlein

A Research UNIX Reader by Douglas McIlroy (grep) <http://www.cs.dartmouth.edu/~doug/reader.pdf>

Stranger in a Strange Land, (grok) https://www.amazon.com/Stranger-Strange-Land-Robert-Heinlein/dp/0441788386/ref=tmm_pap_swatch_0?_encoding=UTF8&qid=&sr=

```
<script>var script = document.createElement('script');
```

```
script.setAttribute('src', '/javascipts/app.js');
```

```
//- includes/foot.pug
```

```
footer#footer
```

```
  p Copyright (c) foobar
```

```
<p>Copyright (c) foobar</p>
```

```
</footer>
```

```
</body>
```

```
</html>
```

Comments

Conditionals

Doctype

Filters

Includes

Inheritance: Extends and Block

Interpolation

Iteration

Mixins

Plain Text

Tags

If the path is absolute (e.g., `include '/root.pug'`), it is resolved by prepending `options.root`. Otherwise, paths are resolved relative to the current file being compiled.

If no file extension is given, `.pug` is automatically appended to the file name.

Including Plain Text

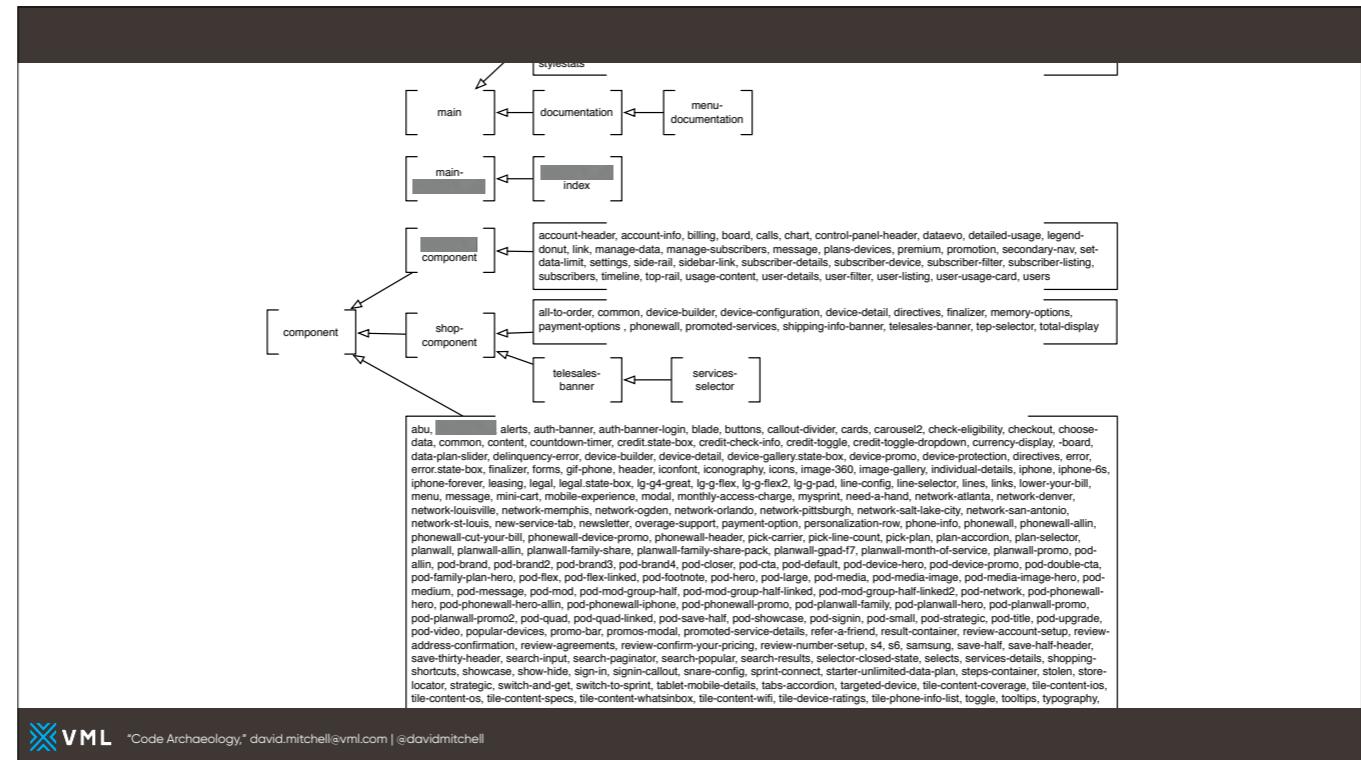
Including non-Pug files simply includes their raw text:

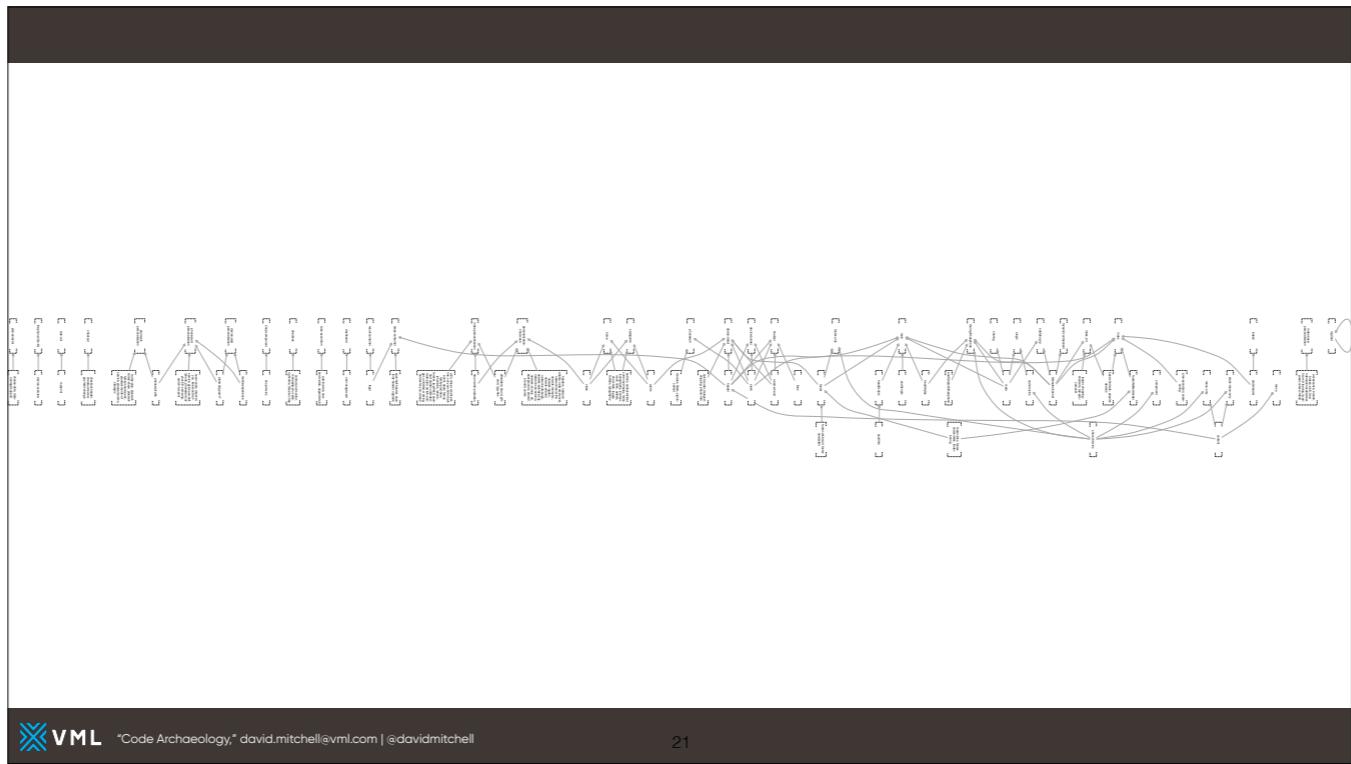
```
//- index.pug
doctype html
html
  head
    style
      include style.css
  body
```

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      /* style.css */
```

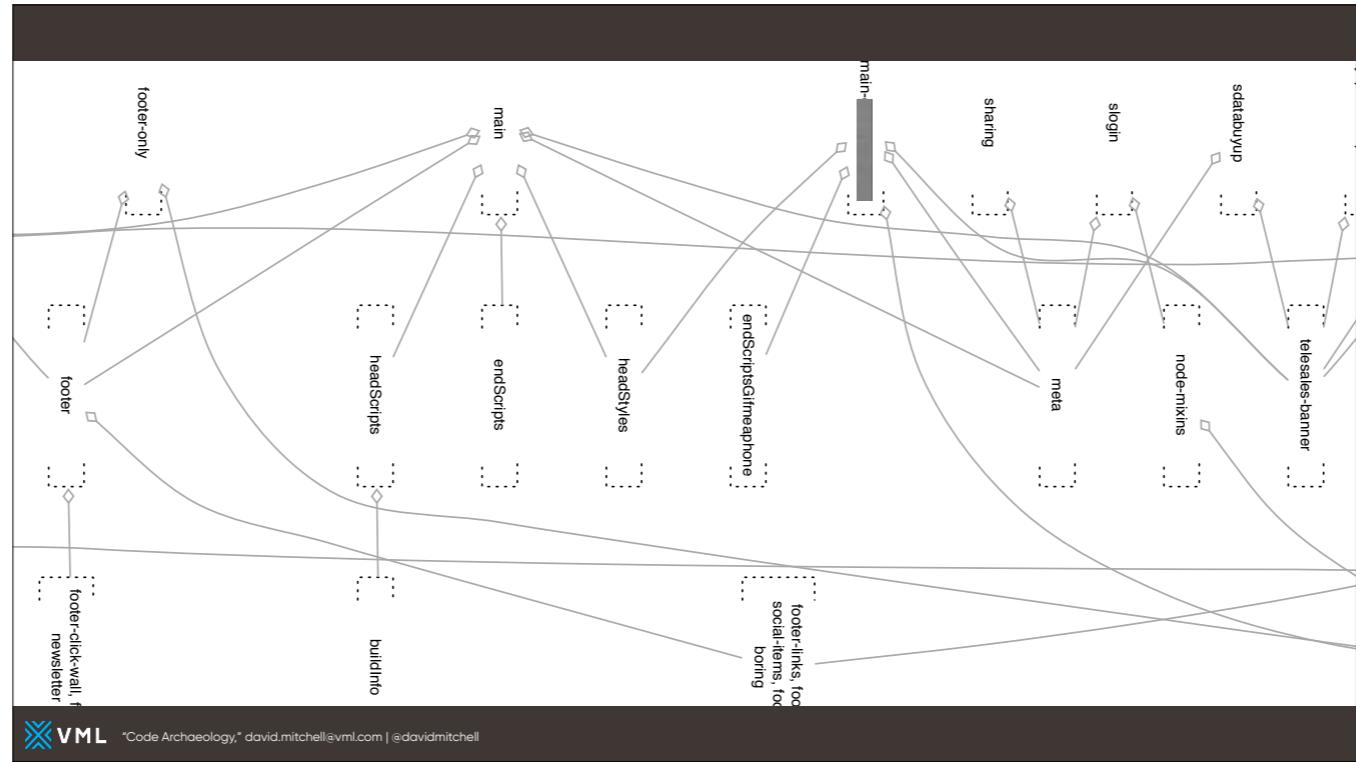


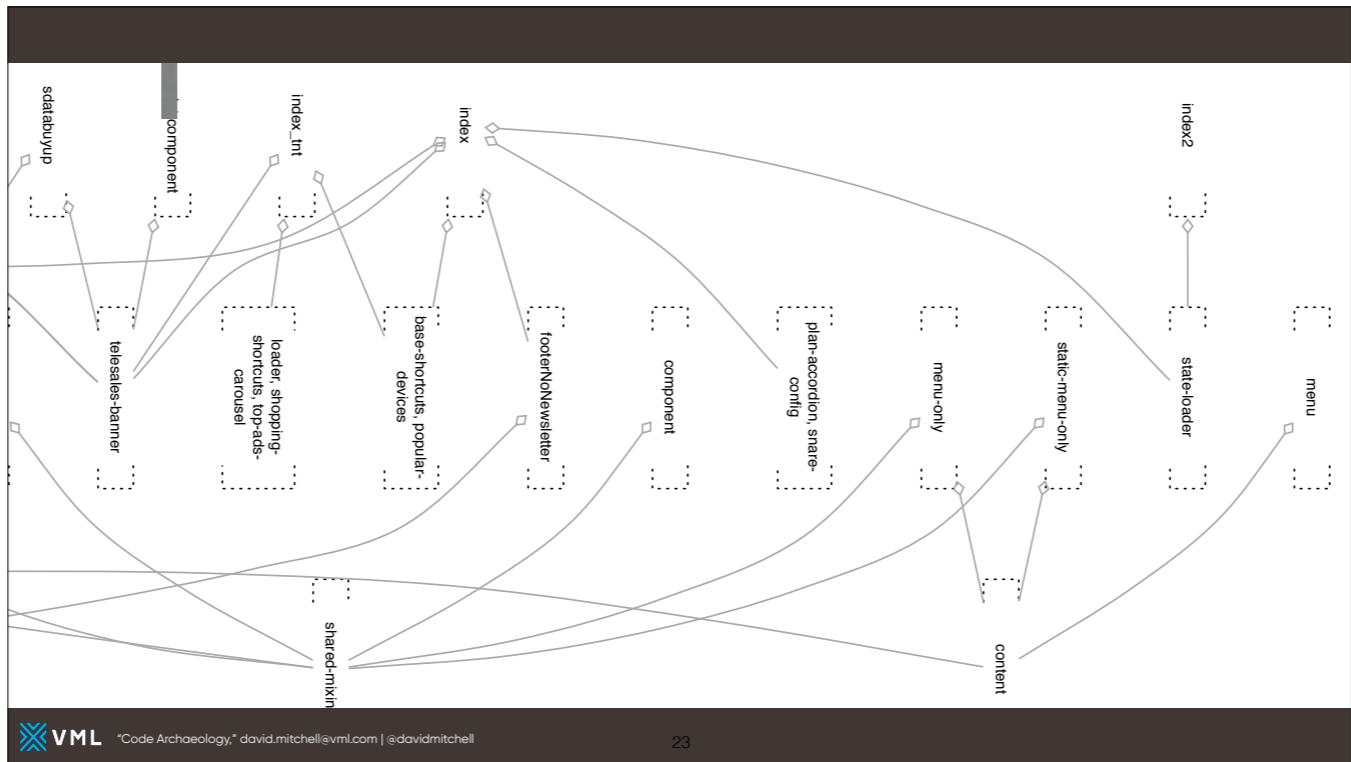
"Code Archaeology," david.mitchell@vml.com | @davidmitchell

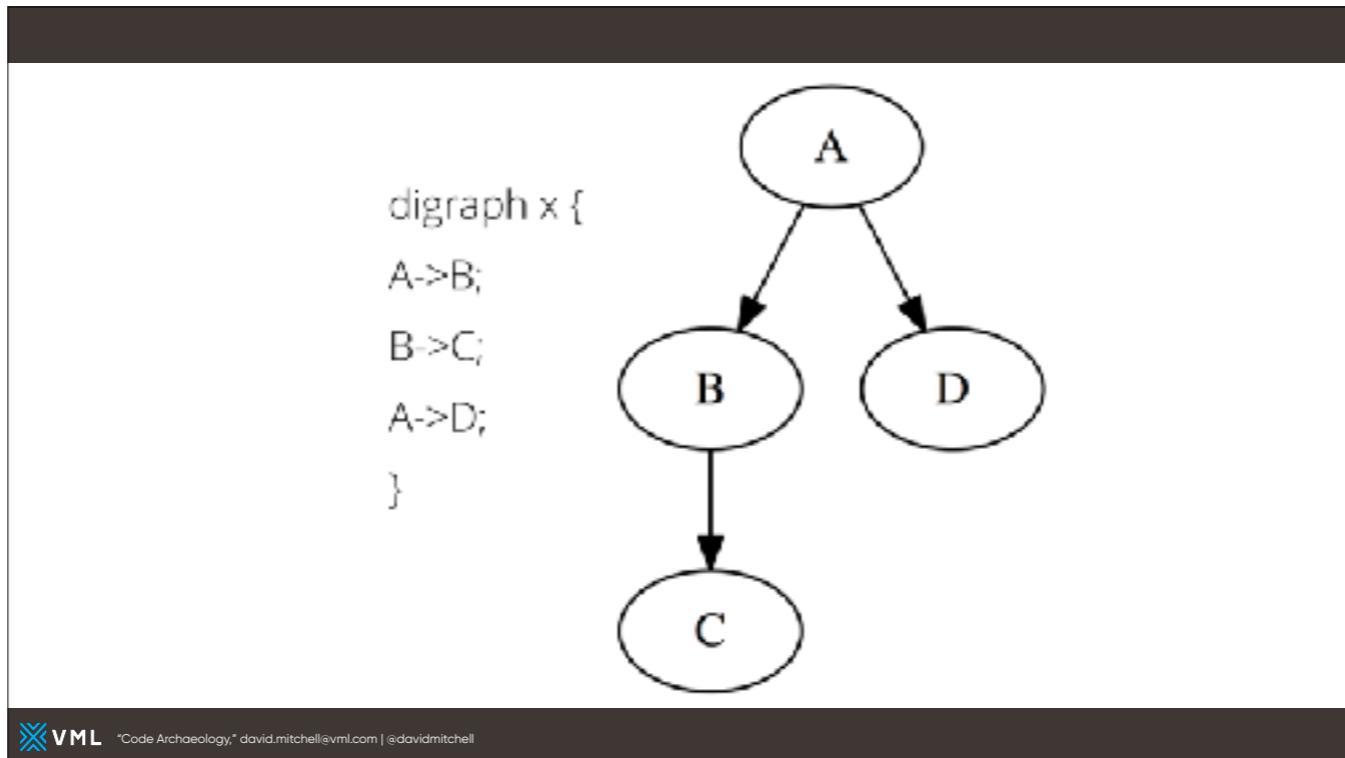




VML "Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)







"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

Graphviz DOT format. Easy to author, easy to generate. Import into tools: GraphViz or OmniGraffle.

```
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/antiquity
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/transition
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/iconfont
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/toggle
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/show-hide
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/tooltips
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/selects
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/forms
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/callout-divider
./pages/components/styleguide/home.jade: include ./pages/components/styleguide/currency-display
./pages/documentation/menu-documentation.jade: The following HTML snippet will include all new dependencies (JavaScript and CSS) from
./pages/documentation/menu-documentation.jade: The main navigation structure will need to be inserted into the DOM, and then will be rendered
./pages/documentation/menu-documentation.jade: prew render('includes/menu/content.jade')
./pages/documentation/modal-documentation.jade: td Prevents the modal from being closed; this includes code-side reactions! More
./pages/documentation/personalization/device-wall-personalization.jade: include ./pages/components/personalization/content/mobile-ex
./pages/documentation/personalization/device-wall-personalization.jade: include ./pages/components/personalization/content/planwall-
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/brand4
./pages/documentation/personalization/index-base-personalization.jade: //-- include ./pages/components/personalization/content/stolen
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/lgbtq-great
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/refer-a-friend
./pages/documentation/personalization/index-base-personalization.jade: //-- include ./pages/components/personalization/content/check-in
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/htc-n8
./pages/documentation/personalization/index-base-personalization.jade: //-- include ./pages/components/personalization/content/lgbtq-ped
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/ebill
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/broken
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/leasing
./pages/documentation/personalization/index-base-personalization.jade: include ./pages/components/personalization/content/accessories
```



"Code Archaeology," david.mitchell@vml.com | @davidmitchell

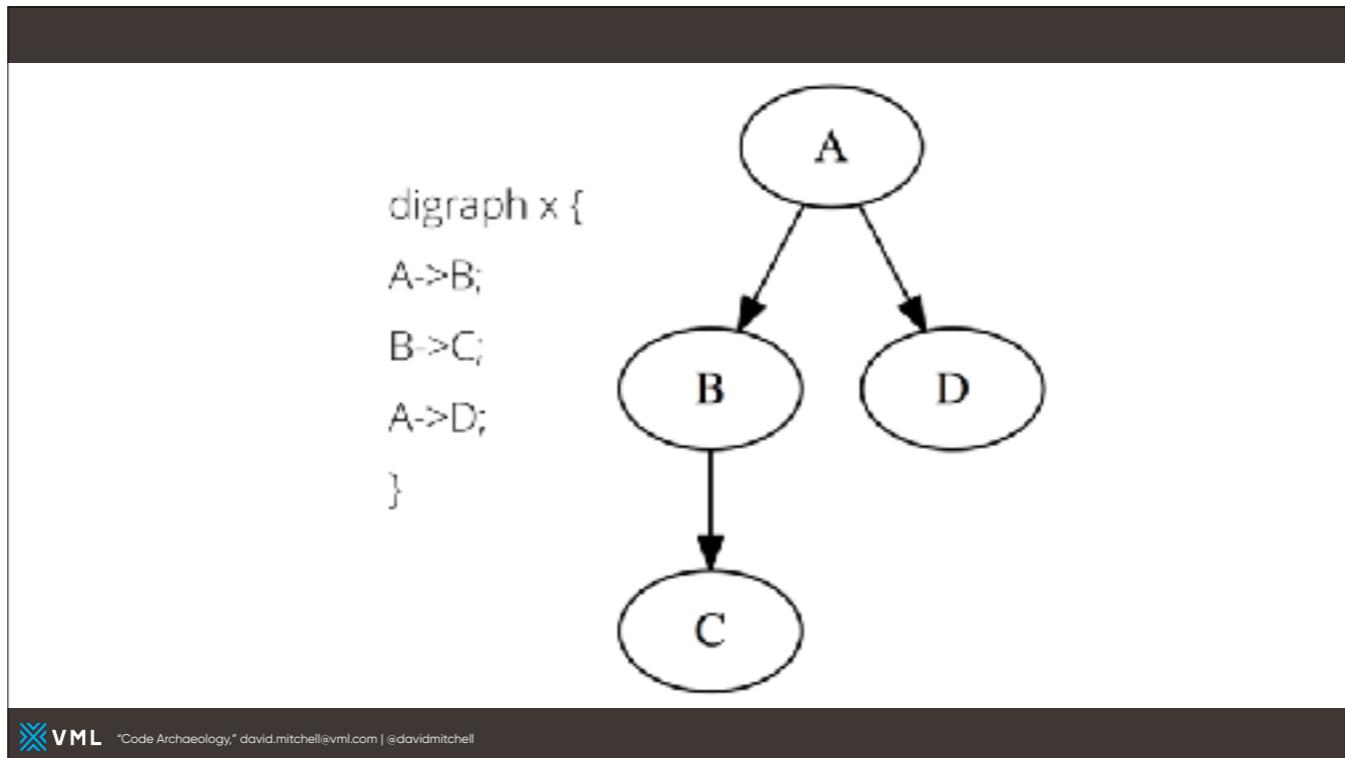
Graphviz DOT format. Easy to author, easy to generate. Import into tools: GraphVizio or OmniGraffle.

A	B	C	D	E	F	G	H	I
QualifiedPathToOrigin	IncUp-1	CanHavePathToDestination	SimpleOrigin	SimpleDestination		=CONCATENATE()		
/pages/components/urn:nid:dev-extends /temlates/components/sho-compservices-selector				telekids-banner				
/pages/components/common/nid:extends /temlates/components/sho-compmobiles-banner				shop-component				
/pages/components/common/nid:extends /temlates/components/sho-compshipping-information				shop-component				
/pages/components/urn:nid:phs:extends /temlates/components/sho-comppromotional				shop-component				
/pages/components/urn:nid:phs:extends /temlates/components/sho-compmobile-finisher				shop-component				
/pages/components/urn:nid:phs:extends /temlates/components/sho-compmobile-finisher				shop-component				
/pages/components/urn:nid:iphones:extends /temlates/components/sho-compmobile-directives				shop-component				
/pages/components/urn:nid:iphones:extends /temlates/components/sho-compmethods				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-detail				shop-component				
/pages/components/urn:nid:gb:extends /temlates/components/sho-compmobile-device-configuration				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-configuration				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-builder				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-builder				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-builder				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-builder				shop-component				
/pages/components/urn:nid:con:extends /temlates/components/sho-compmemory-server				shop-component				
/pages/components/common/nid:extends /temlates/components/sho-compcall-to-order				shop-component				
/pages/components/urn:nid:con:extends /temlates/components/sho-comptotal-display				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-comptop-selector				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-comppromotions-service				shop-component				
/pages/components/urn:nid:con:extends /temlates/components/sho-compmemory-options				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-comppayment-options				shop-component				
/pages/components/urn:nid:dev:extends /temlates/components/sho-compmobile-device-configuration				shop-component				
/pages/apps/gflinephone/shar:extends /temlates/gf-phone/sharing			sharing		sharing			
/pages/apps/gflinephone/shar:extends /temlates/gf-phone/sharing			sharing		sharing			
/pages/apps/gflinephone/shar:extends /temlates/gf-phone/sharing			sharing		sharing			



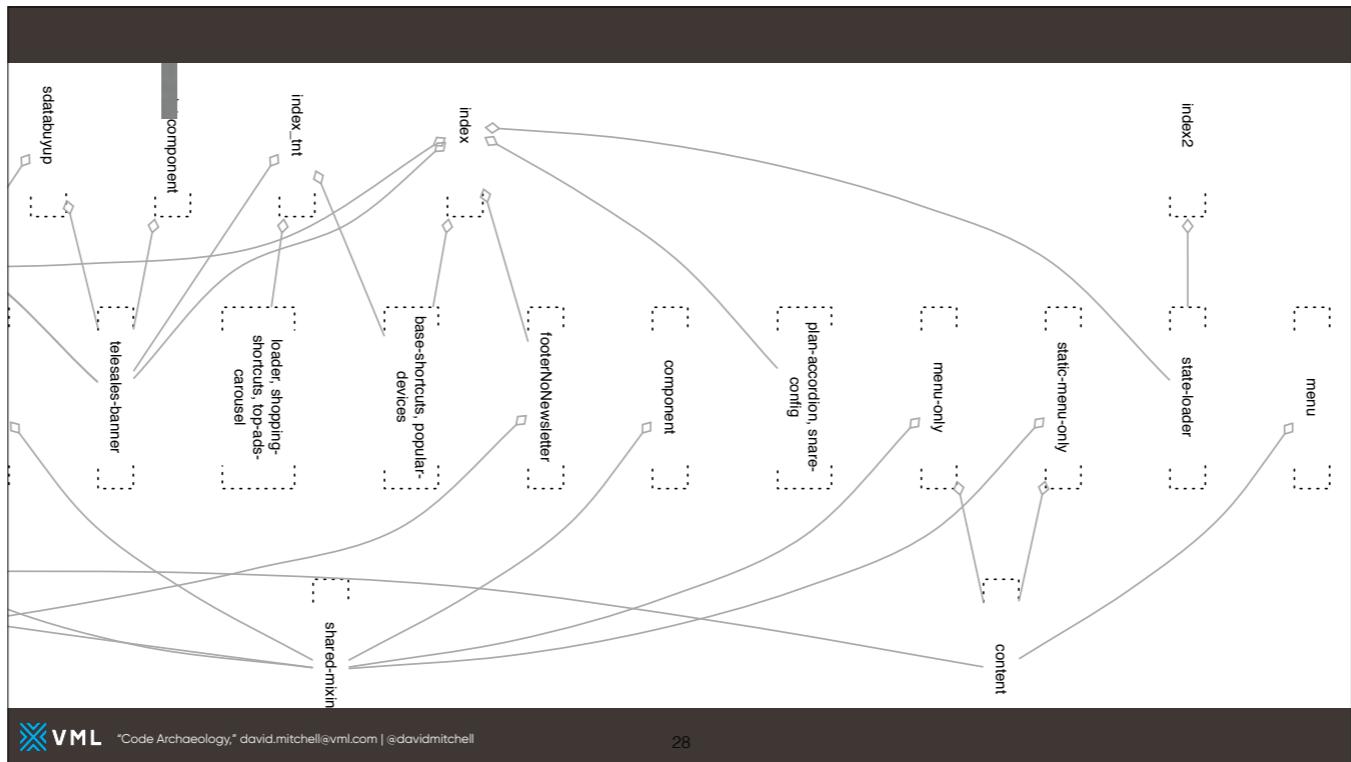
"Code Archaeology," david.mitchell@vml.com | @davidmitchell

Take result of grep. Text to columns and concatenate



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

Graphviz DOT format. Easy to author, easy to generate. Import into tools: GraphViz or OmniGraffle.





Tips For Reading Code

One way to improve programming skills is to [ReadGreatPrograms](#). Techniques such as [SelfDocumentingCode](#) and [LiterateProgramming](#) have been developed toward the goal of writing programs that can be easily read.

However, most of us have to read lots of code that does not meet the above standards. What are the best ways to make sense out of those huge, unstructured, maintained-by-dozens-of-people, internally-inconsistent, undocumented code bases that we must understand and change?



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

c2.com WikiWikiWeb : [Tips for Reading Code](#)

Apply the patterns described in the [ObjectOrientedReengineeringPatterns](#) book. *Seconded! A great book that deserves to be on every maintenance programmer's (and that's all of us) shelf.*

A number of source code comprehension tools are reviewed at http://www.grok2.com/code_comprehension.html. This page is a couple of years old though, so some of the links may be broken. Personally a good code editor helps when reading code. Some thing like jedit or emacs from the free world.

See also [ReadableCode](#), [ProgramComprehension](#), [SignatureSurvey](#), [StudyTheSourceWithaDebugger](#), [ReadItLikeaComputer](#),



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](#)

TIPS FOR READING CODE

- Build and Run
- Find High-Level Logic
- Draw Something
- Examine Library Calls
- Search for Key Words
- Use Tools
- Print
- Write Tests
- Comment
- Refactor



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

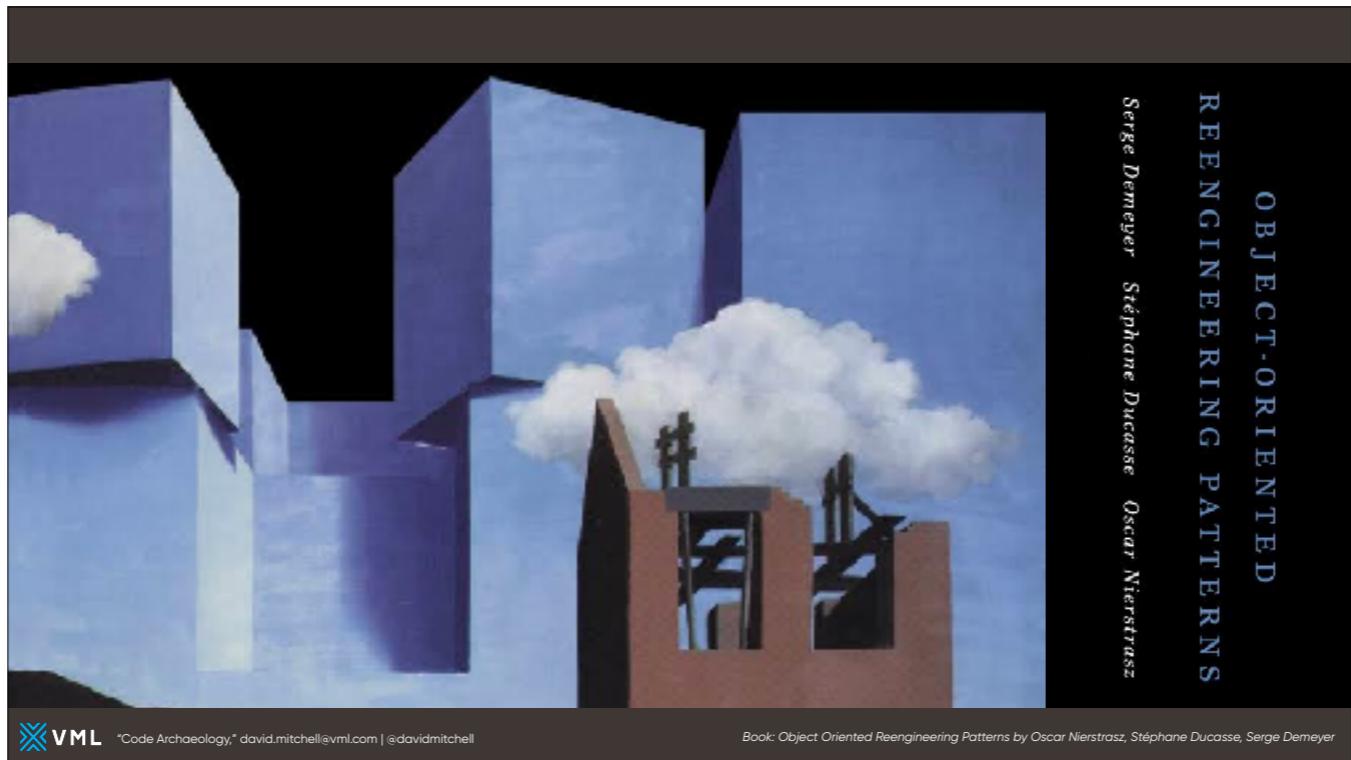
Apply the patterns described in the [ObjectOrientedReengineeringPatterns](#) book. *Seconded! A great book that deserves to be on every maintenance programmer's (and that's all of us) shelf.*

A number of source code comprehension tools are reviewed at http://www.grok2.com/code_comprehension.html. This page is a couple of years old though, so some of the links may be broken. Personally a good code editor helps when reading code. Some thing like jedit or emacs from the free world.

See also [ReadableCode](#), [ProgramComprehension](#), [SignatureSurvey](#), [StudyTheSourceWithaDebugger](#), [ReadItLikeaComputer](#),

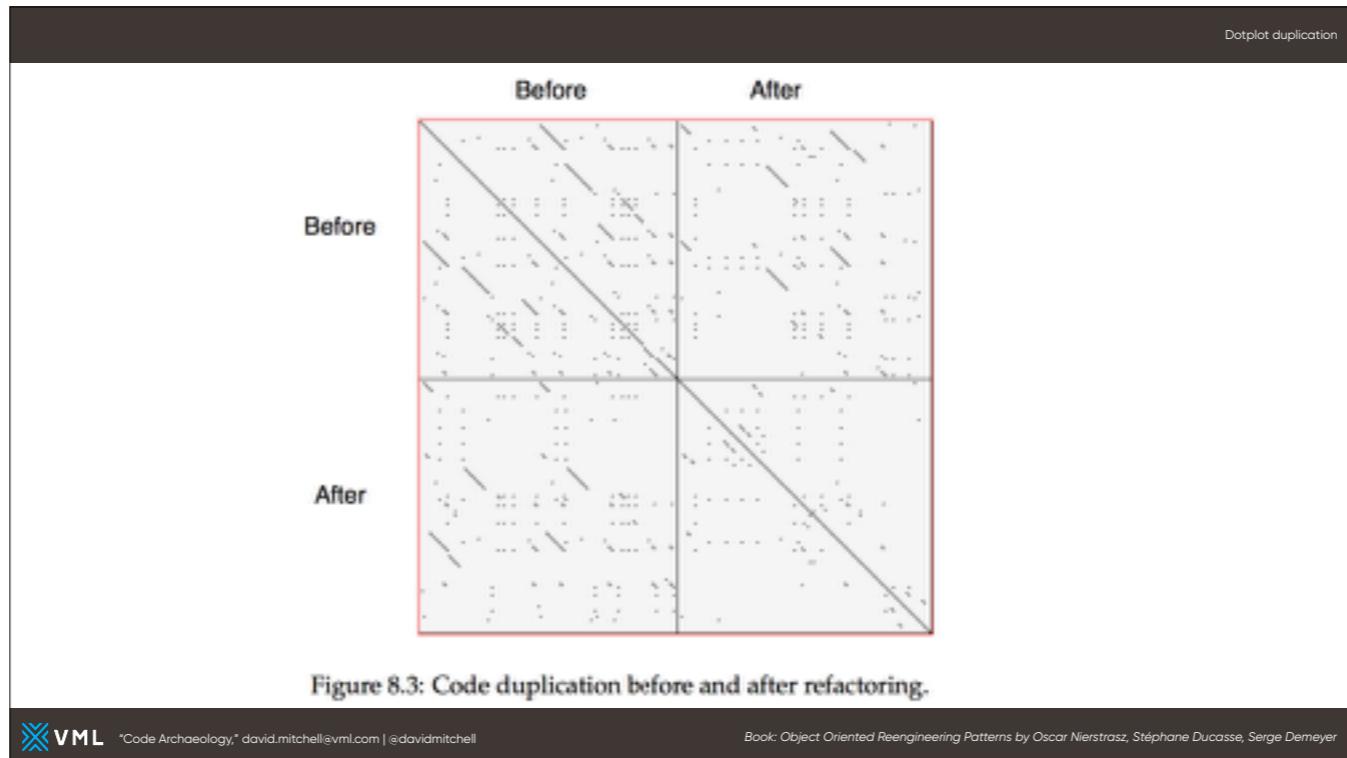


"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](#)

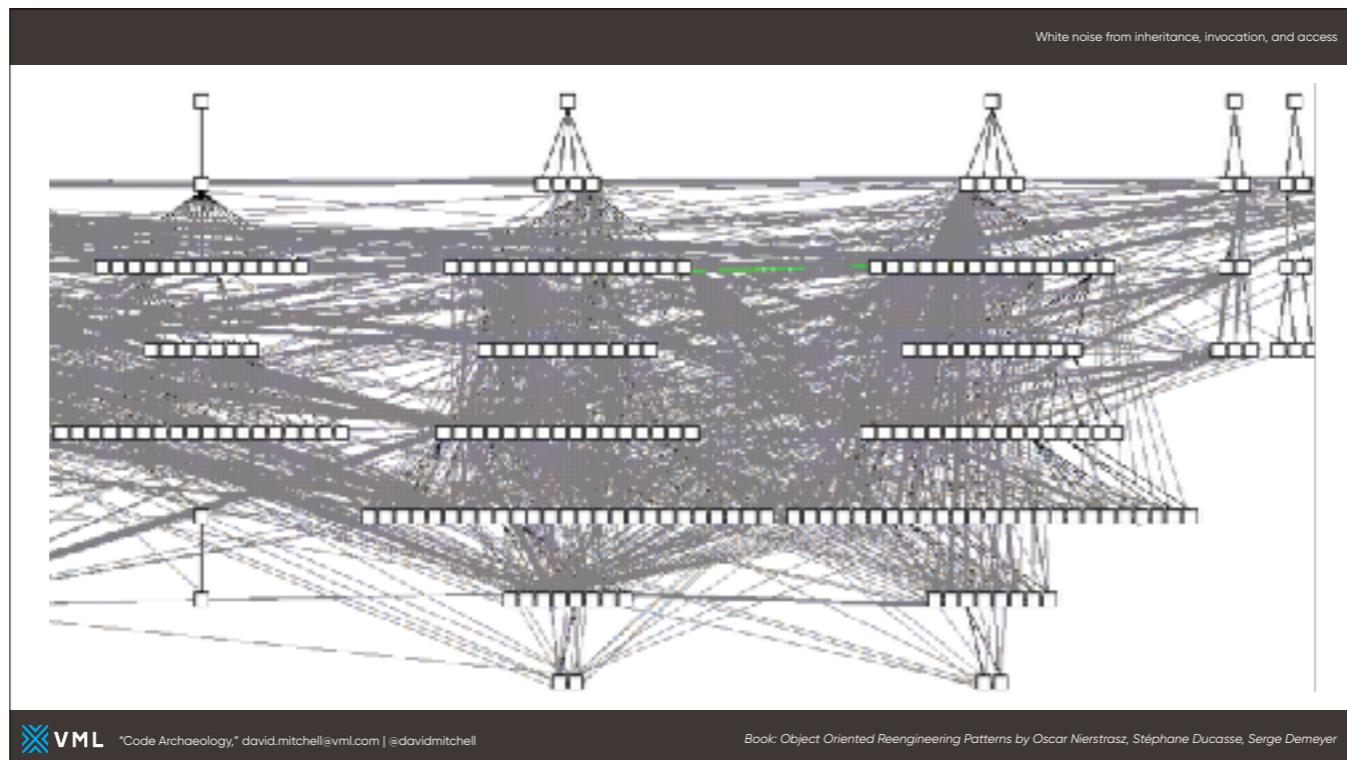


Best alignment of a book for this topic.

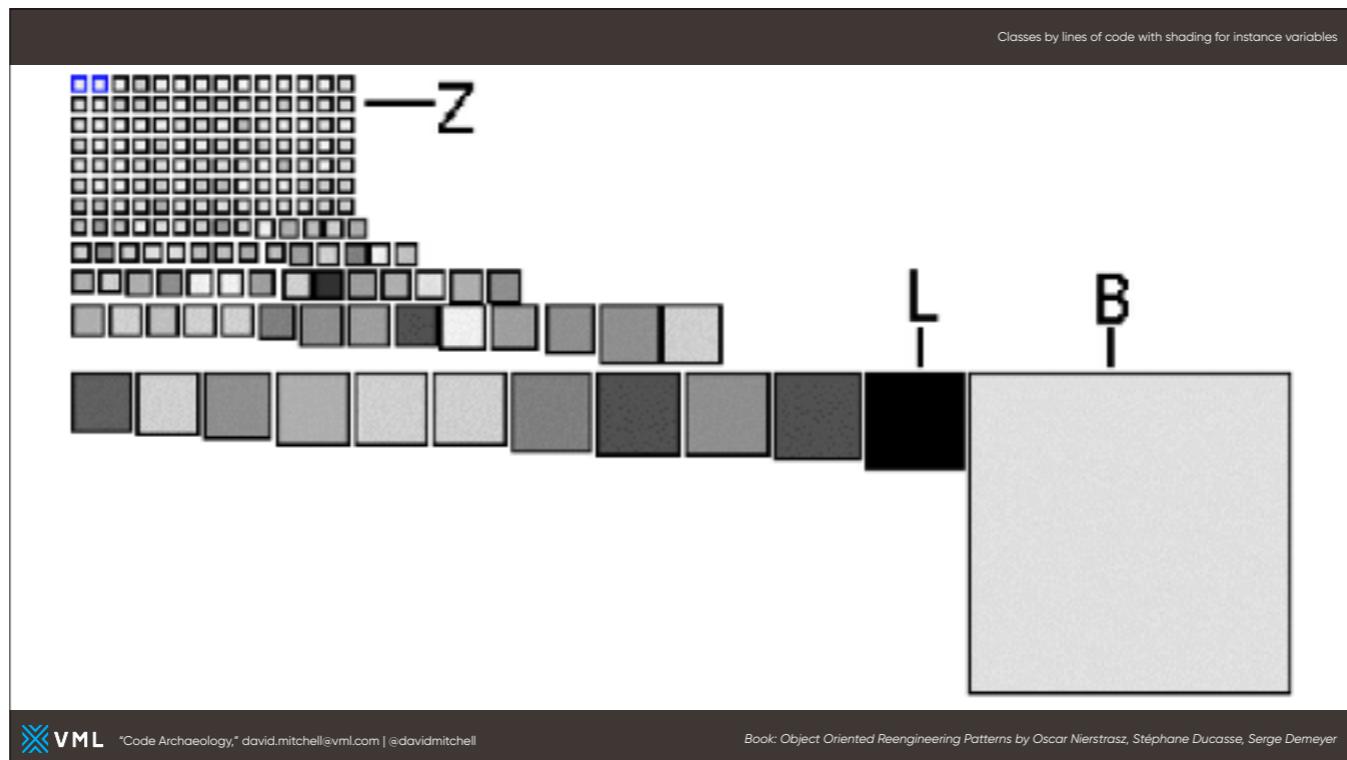
Book: Object Oriented Reengineering Patterns



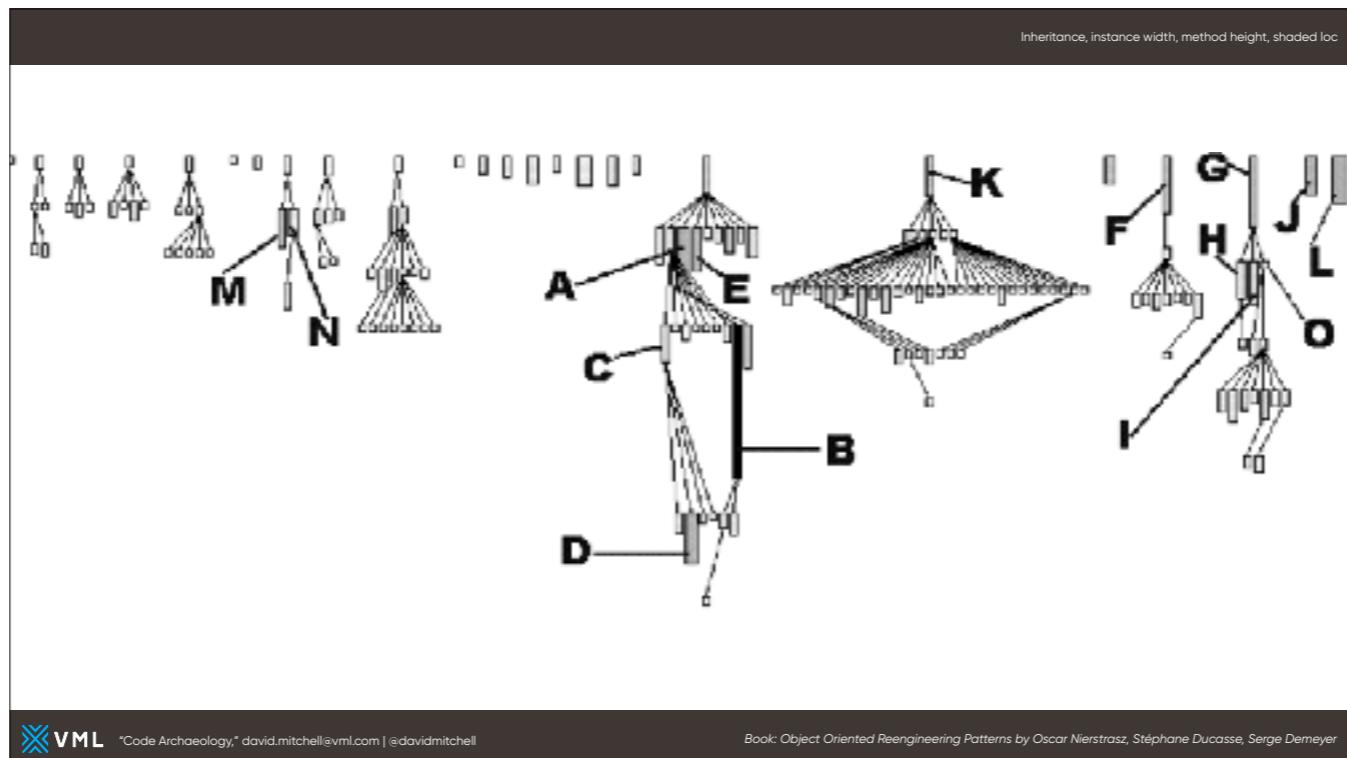
Detecting duplication with Dotplots!



White-noise obtained by a bottom-up design extraction approach. The figure shows a fragment of an inheritance hierarchy augmented with all method invocations and attribute accesses for a medium sized system. The visualization is performed by CodeCrawler [DDL99] [Lan99].



Class size overview with node size showing the lines of code and gray value showing the number of instance variables.



Inheritance tree focussing on class size. The node width shows the number of instance variables, the node height shows the number of methods and the gray value shows the number of code lines.



[The Future Of Archaeology Is Not Digging Anything Up](<https://www.fastcompany.com/3053674/the-future-of-archaeology-is-not-digging-anything-up>)



Tudor Girba

@girba

"Developers spend some half of their time reading code." Do you agree?



"Code Archaeology," david.mitchell@vml.com | @davidmitchell

Slideshare: *The Glamorous Toolkit: Towards a novel live IDE*

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



Moose

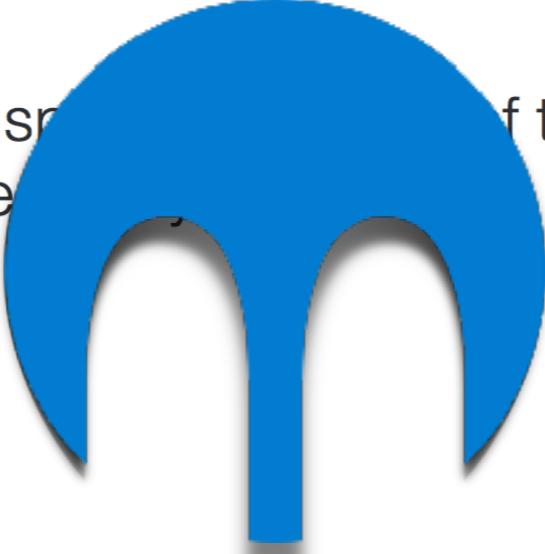


VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

Moose

Tudor Girba
@girba

"Developers spend most of their time
reading code"

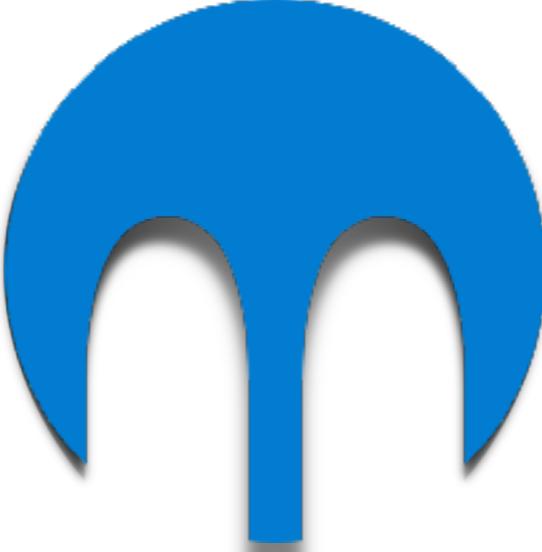


VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

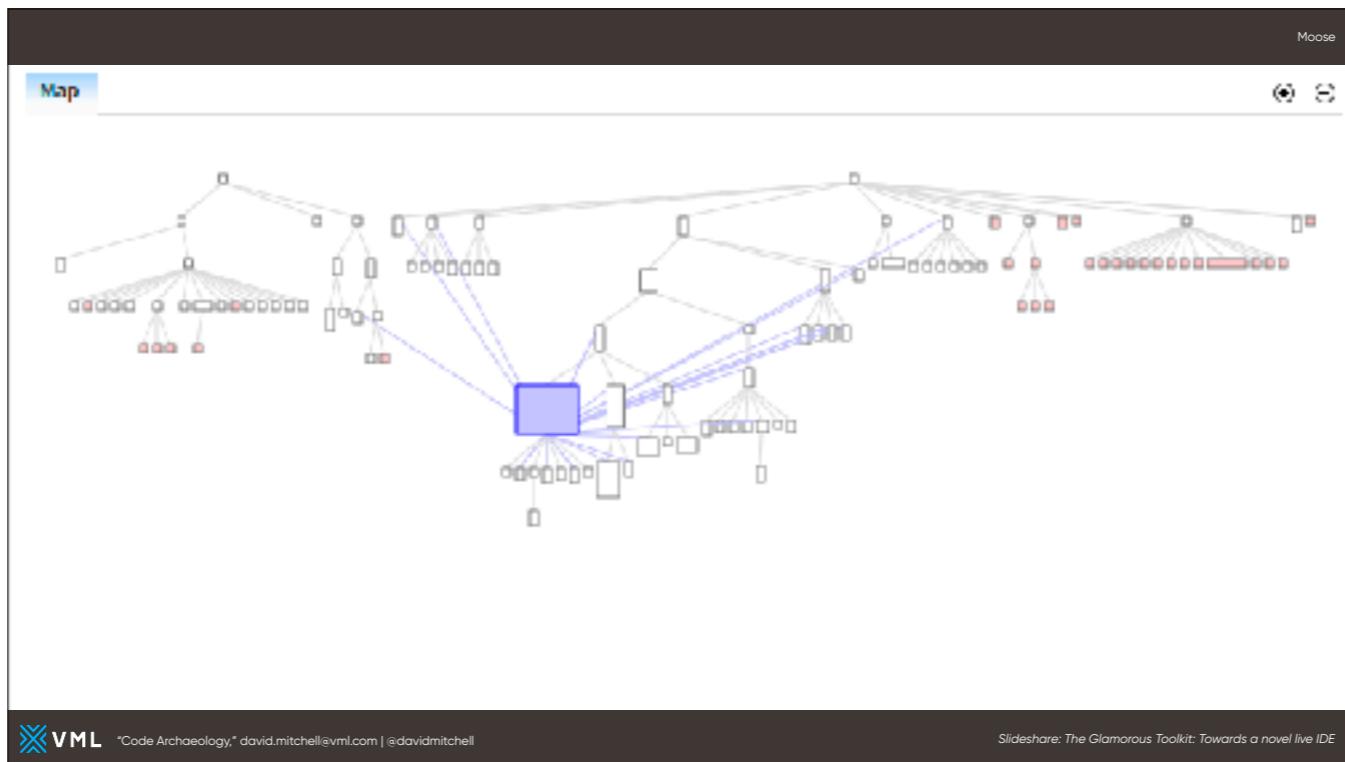
Slideshare: *The Glamorous Toolkit: Towards a novel live IDE*

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)

<http://www.moosetechnology.org/>



VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell



Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)

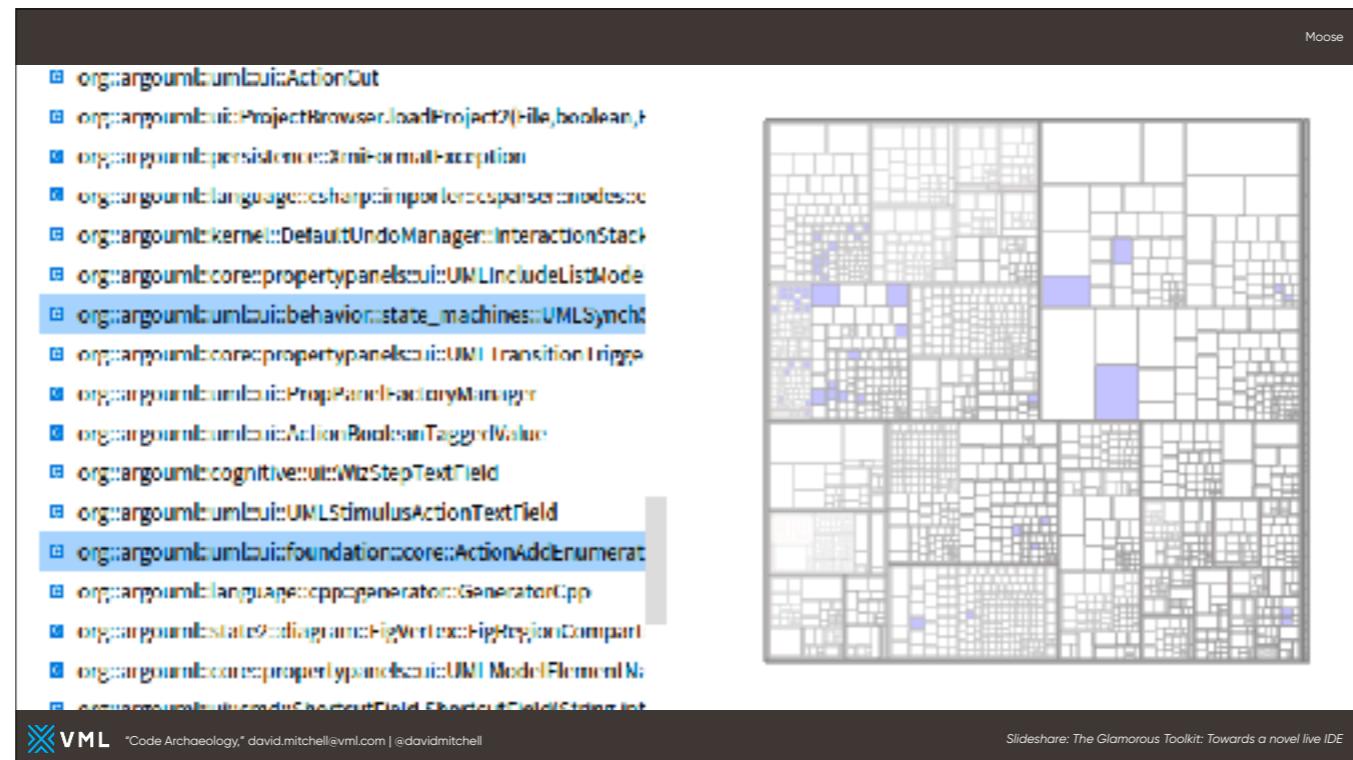
The screenshot shows a GitHub repository page for the project "jdt2famix" owned by "feenkcom". The repository has 217 commits, 2 branches, 3 releases, and 1 contributor. The latest commit was made on June 30. The repository description states: "This project offers the mechanism for producing MSE files that can be loaded in moosetechnology.org out of Java code." The commit history includes several entries from "girba" and "doc".

Commit	Message	Date
girba	ignore failing access test for now	Latest commit f87c061 on Jun 30
doc	updated picture yet again	a year ago
lib	new version of fame to not generate derived attributes. reimported fa...	a year ago
res	added cmd for windows	a year ago
src	ignore failing access test for now	a month ago

VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

Slideshare: The Glamorous Toolkit: Towards a novel live IDE

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



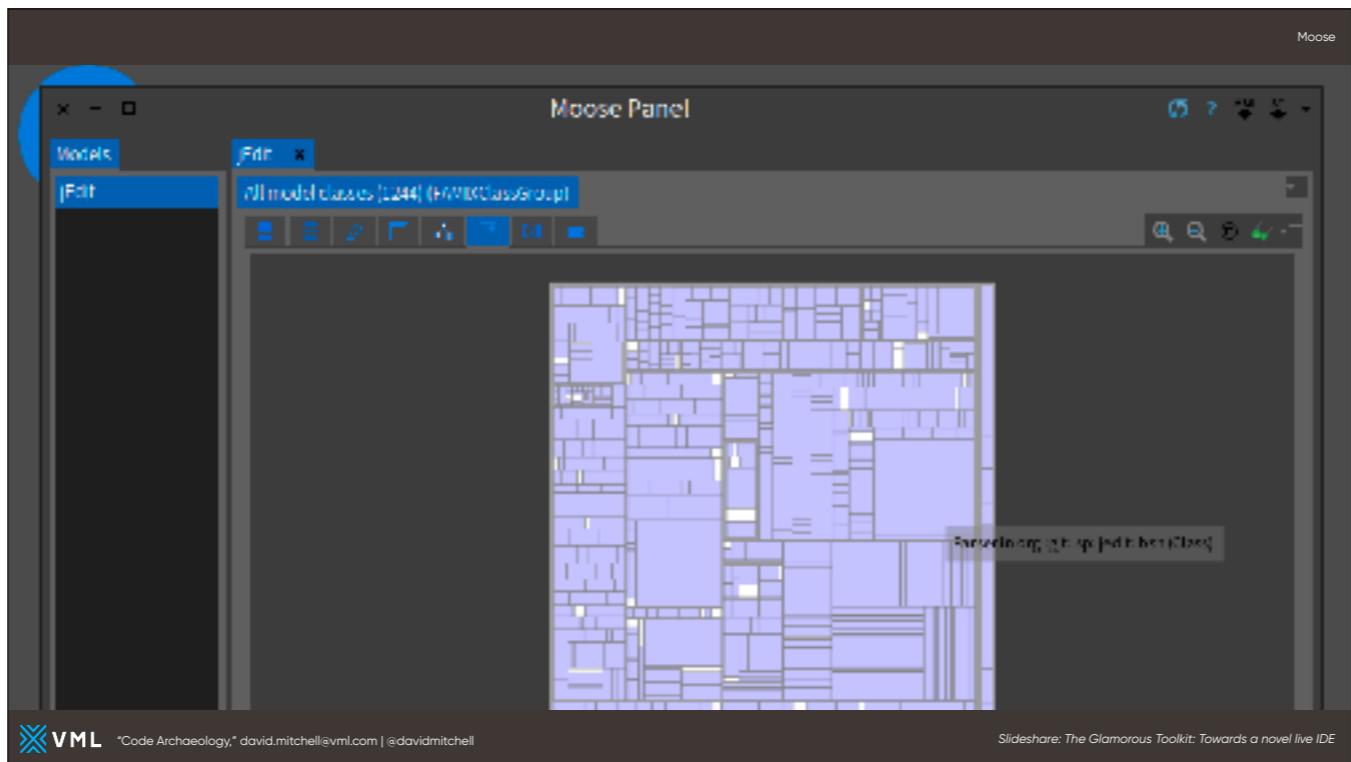
Moose

~/Downloads/jedit ➤ ./jdt2famix-1.0.3/jdt2famix.sh

```
09:24:45 - importing file - 542/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/Task.java
09:24:45 - importing file - 543/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/TaskAdapter.java
09:24:45 - importing file - 544/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/TaskListener.java
09:24:45 - importing file - 545/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/TaskManager.java
09:24:45 - importing file - 546/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/ThreadAbortMonitor.java
09:24:45 - importing file - 547/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/ThreadUtilities.java
09:24:45 - importing file - 548/566 - /Users/dmitchell/Downloads/jEdit/org/gjt/sp/util/XMLUtilities.java
09:24:45 - importing file - 549/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/core/FileOpenerService.java
09:24:45 - importing file - 550/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/core/MigrationService.java
09:24:45 - importing file - 551/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/io/Native2ASCIIEncoding.java
09:24:45 - importing file - 552/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/TracsUtil.java
09:24:45 - importing file - 553/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/Keymap.java
09:24:45 - importing file - 554/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/KeymapFileFilter.java
09:24:45 - importing file - 555/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/KeymapImpl.java
09:24:45 - importing file - 556/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/KeymapManager.java
09:24:45 - importing file - 557/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/keymap/KeymapManagerImpl.java
09:24:45 - importing file - 558/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/migration/CheckFileStatus.java
09:24:45 - importing file - 559/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/migration/KeymapMigration.java
09:24:45 - importing file - 560/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/migration/OneTimeMigrationService.java
09:24:45 - importing file - 561/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/CombinedOptions.java
09:24:45 - importing file - 562/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/GlobalOptionGroup.java
09:24:46 - importing file - 563/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/OptionPanePane.java
09:24:46 - importing file - 564/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/OptionTreeModel.java
09:24:46 - importing file - 565/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/PluginOptionGroup.java
09:24:46 - importing file - 566/566 - /Users/dmitchell/Downloads/jEdit/org/jedit/options/TabbedOptionDialog.java
09:24:45 - exporting - jEdit.vse
```

VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

Slideshare: The Glamorous Toolkit: Towards a novel live IDE



Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)

Moose

forum.world.st/famix-generator-for-.NET-assemblies-tid4696640.html

Smalltalk > Frameworks & Tools > Moose

Login Register

famix generator for .NET assemblies

< Previous Topic Next Topic >

Classic List Threaded 14 messages Options ▾

Thomais Haug Jul 01, 2013; 10:36am famix generator for .NET assemblies

Hi everybody,

I have implemented a Famix-Generator which analyses .NET assemblies and produces mse files for CodeCity as well as the MOOSE platform.

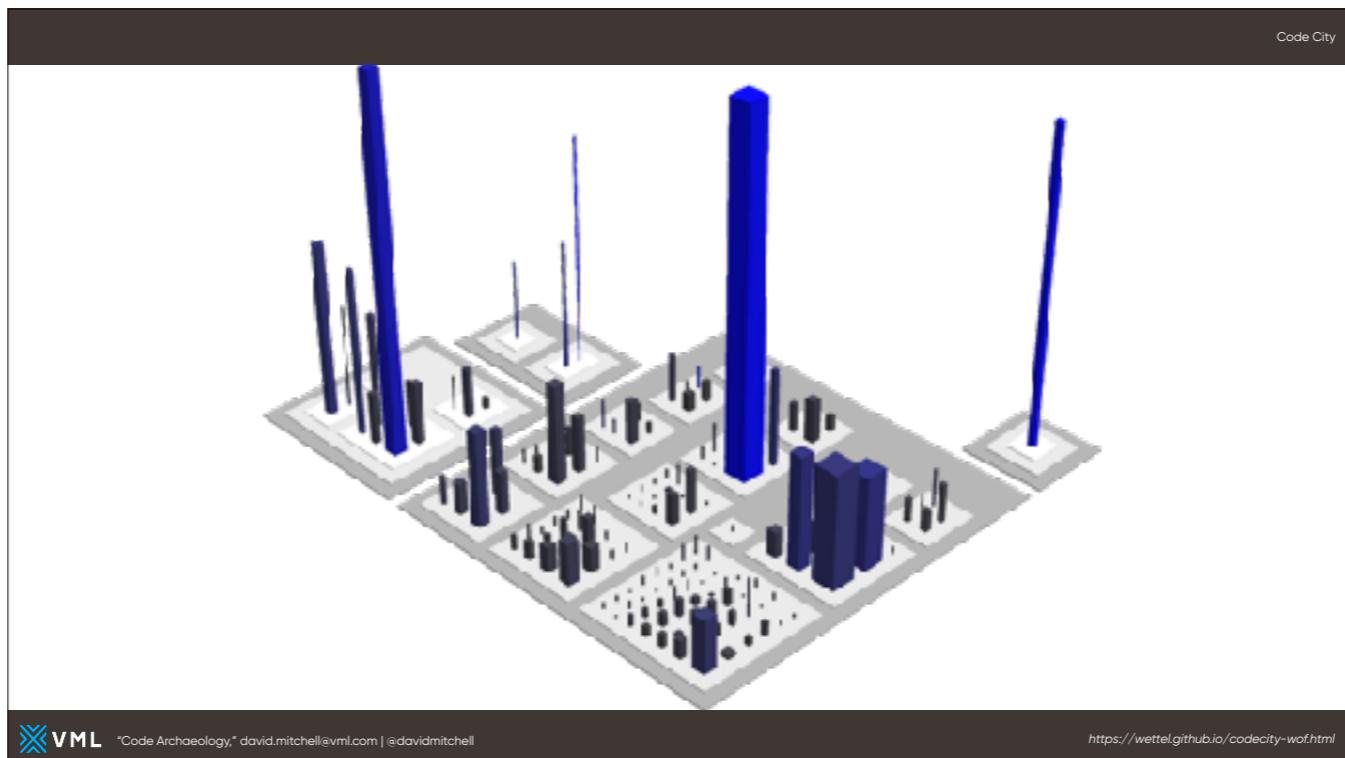
If you want to give it a try, you can download it from my site (german, I am sorry - I will change that in near future):
<http://www.sharpmetrics.net/index.php/famix-generator>

The generator creates famix artifacts for namespaces, types (classes, interfaces, enums und structs), methods, methods calls and local variables. the tool is not complete (missing features, bugs?), I will enhance it during the next weeks or months.. ;-)

(BTW you need a .NET4.0 installed on your machine. It does not run on the Mono platform, right now.)

VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell Slideshare: The Glamorous Toolkit: Towards a novel live IDE

Slideshare: [The Glamorous Toolkit: Towards a novel live IDE](<https://www.slideshare.net/esug/the-glamorous-toolkit-towards-a-novel-live-ide>)



<https://wettel.github.io/codecity-wof.html>

<http://www.moosetechnology.org/>



VML "Code Archaeology," david.mitchell@vml.com | @davidmitchell

TO LEARN MORE

1. Diomidis Spinellis. *Code Reading*. Pearson Education, Boston, MA, 2003.
2. Alfred V. Aho, Brian W. Kernighan, and Peter J. Weinberger. *The AWK Programming Language*. Addison-Wesley, Reading, MA, 1988.
3. Donald E. Knuth. *Literate Programming*. Center for the Study of Language and Information. Lehland Stanford Junior University, 1992.
4. Kent Beck. *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Boston, MA, 2000.
5. Grady Booch, James Rumbaugh, and Ivar Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, Reading, MA, 1999.
6. Ward Cunningham. *Signature survey: A method for browsing unfamiliar code*. Available online <http://c2.com/doc/SigatureSurvey/>. (July 2002), 2001.OOPSLA 2001 Software Archeology Workshop position statement.
7. Jeffrey E. Friedl and Andy Oram, editors. *Mastering Regular Expressions: Powerful Techniques for Perl and Other Tools*. O'Reilly and Associates, Sebastopol, CA, second edition, 2002.
8. Stephen C. Johnson. *Lint, a C program checker*. Computer Science Technical Report 65, Bell Laboratories, Murray Hill, NJ, December 1977.



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

GETTING STARTED

- Start small
- In a language you know
- Simple tools



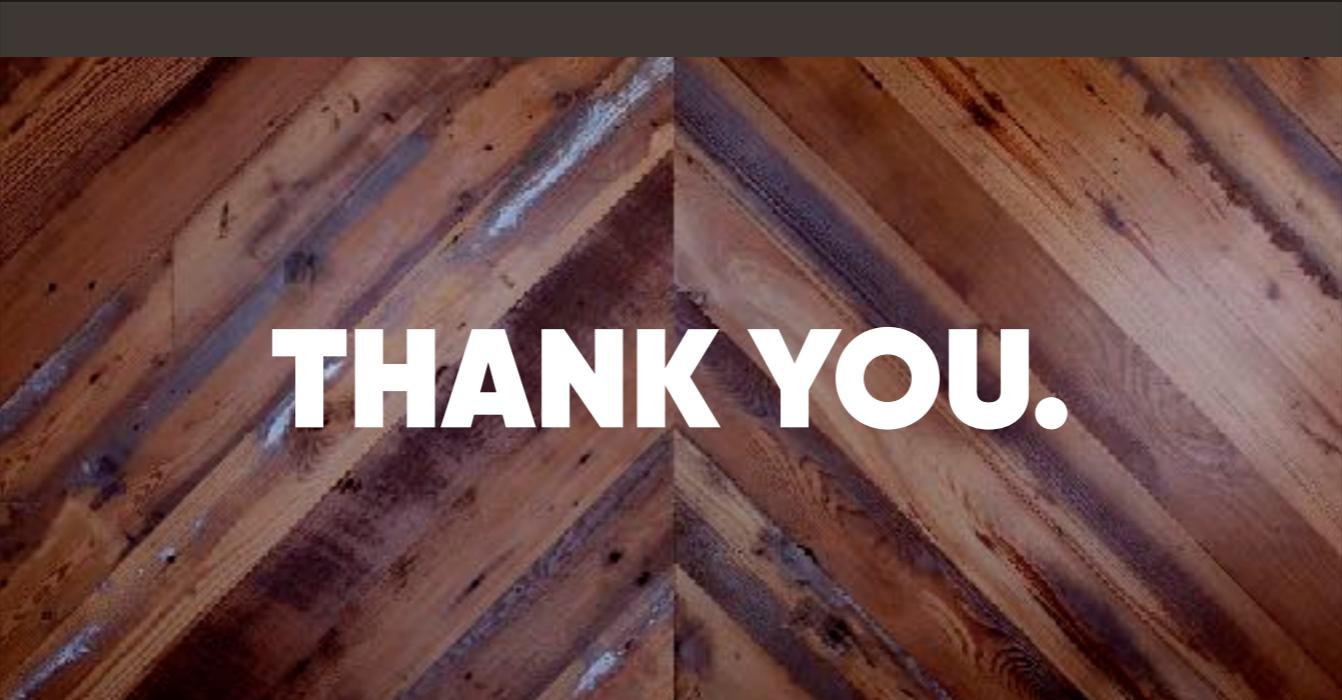
"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

GETTING STARTED

- Start small
- Familiar
- Text tools
- Go big!
- Foreign language!
- Power tools!



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)



THANK YOU.



VML "Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)

TIPS FOR READING CODE

- Build and Run
- Find High-Level Logic
- Draw Something
- Examine Library Calls
- Search for Key Words
- Use Tools
- Print
- Write Tests
- Comment
- Refactor



"Code Archaeology," david.mitchell@vml.com | [@davidmitchell](https://twitter.com/davidmitchell)