Performance Profiling

Swift on Linux

JP Simard — @simjp — jp@lyft.com

You've ported your Swift code to Linux. 🎉

But you find out it's **slooooooooooooww**. 🐢


But I thought Linux was webscale?!
@garit

Why might your Swift code be slower on Linux?

Reasons why Swift code might be slower on Linux

- Platform differences
- Low level library differences (mach ports, libpthread, allocator)
- Different implementations
- Darwin Foundation vs `swift-corelibs-foundation`
- Lack of Objective-C runtime & optimizations

On macOS, you'd use `Instruments.app` 🤔

No such thing as

```
$ apt-get install instruments.app
```

CPU Profilers

Tool	OS	Notes
Instruments.app	macOS	Super powerful, accurate, great GUI
DTrace	macOS, Solaris, FreeBSD	Powers part of Instruments.app, very scriptable, not on Linux
Callgrind	macOS, Linux, Solaris	Fickle, slow, single-threaded, can run in VMs & containers
Perf Events	Linux on bare metal	Fast, accurate, CLI only, requires support for hardware counters

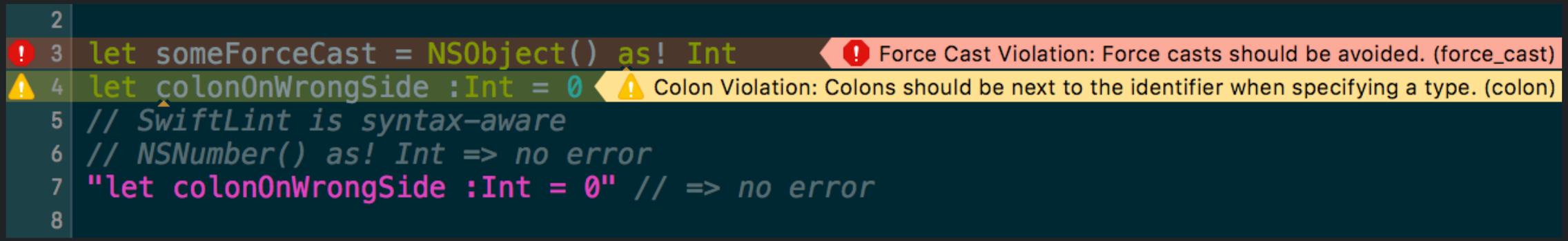
VMs vs Containers vs Bare Metal

VMs/Containers	Bare Metal
Cheap	Expensive
Local on macOS	Dedicated Machine or Dual Boot
Convenient	Not So Much
Limited Perf Tools	No Limitations

Case **Study**

SwiftLint

A tool to enforce Swift style and conventions



— Integrates into Xcode

— Plugins for AppCode, Vim, Sublime Text, Atom, Emacs

— 109 rules and counting, covering lint, idiomatic, style, metrics & performance

```
➜  SwiftLint git:(f074ff7f)
```

SwiftLint Rule

Recently, a rule was introduced that was particularly slow

Variable Declaration Whitespace

Identifier	Enabled by default	Supports autocorrection	Kind
`let_var_whitespace`	Disabled	No	style

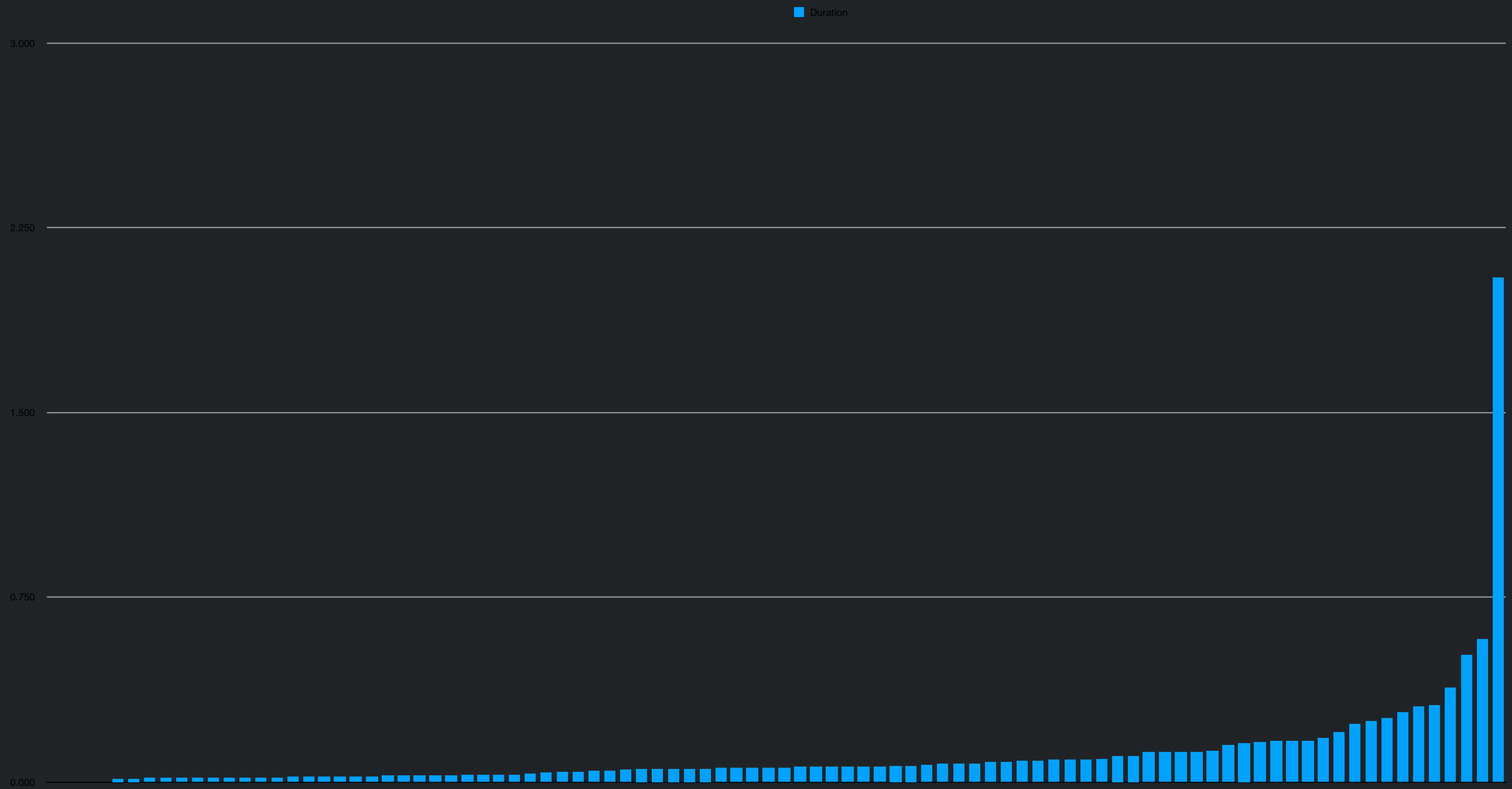
Let and var should be separated from other statements by a blank line.

Examples

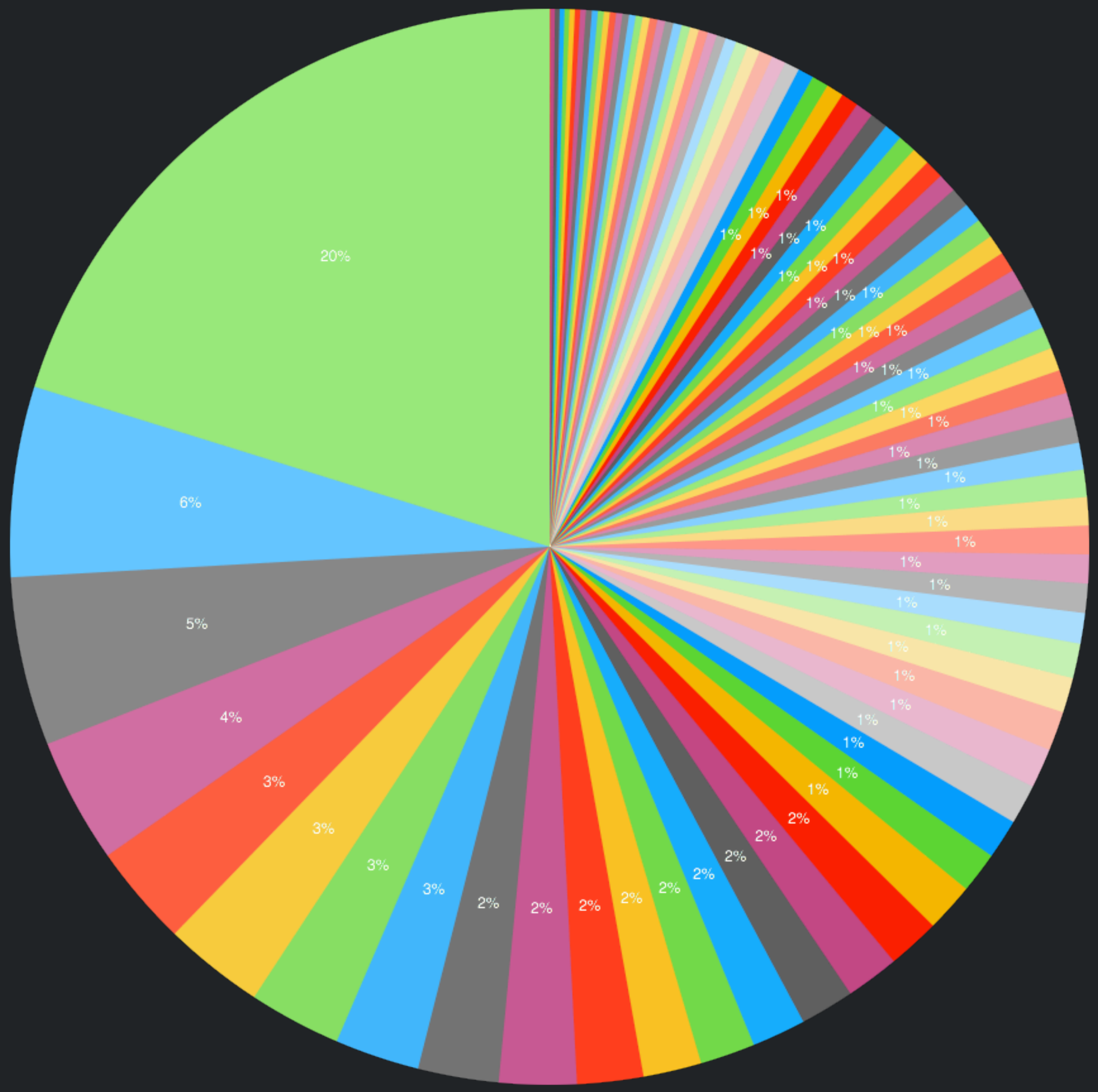
▶ Non Triggering Examples
▶ Triggering Examples

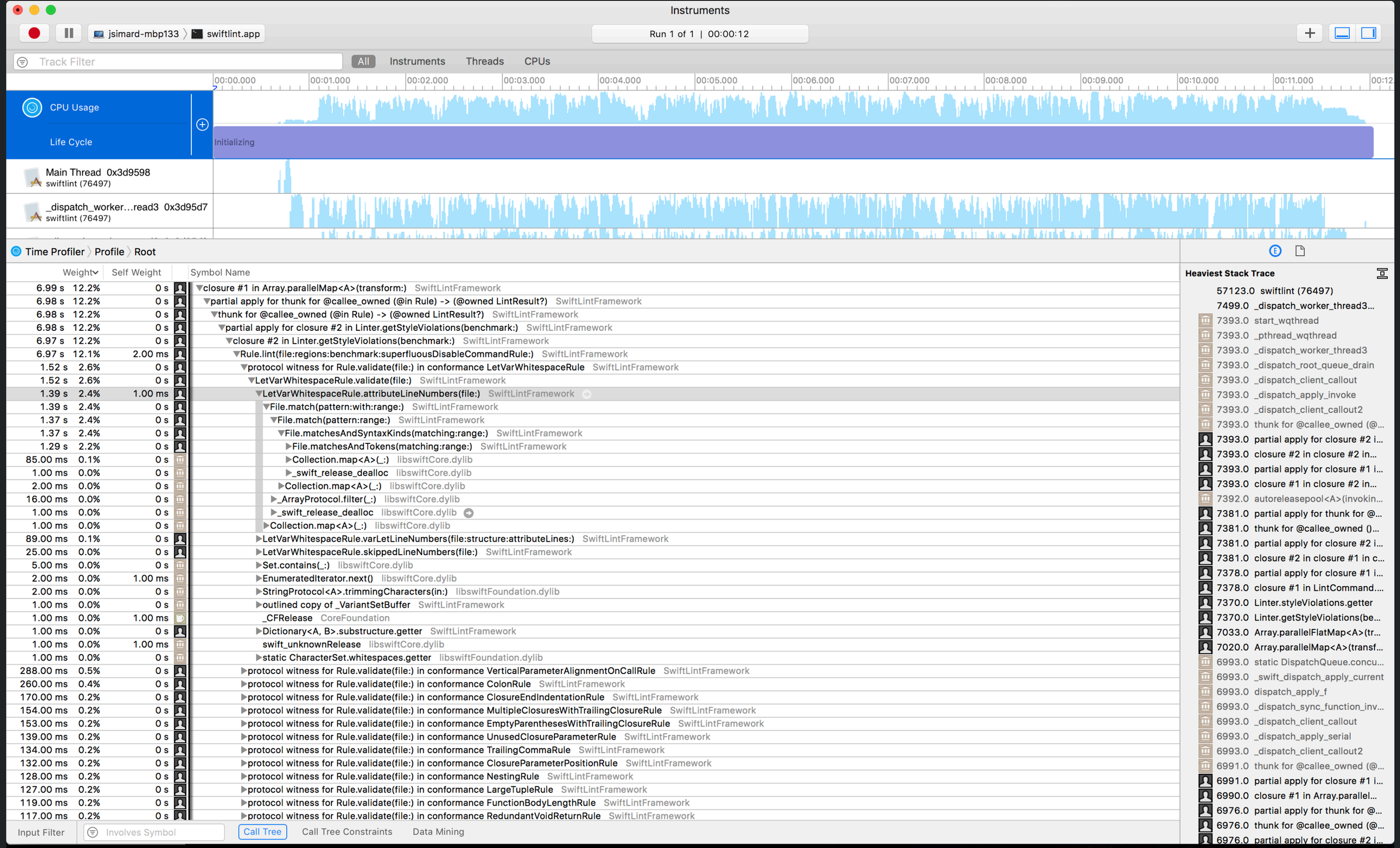
Demo

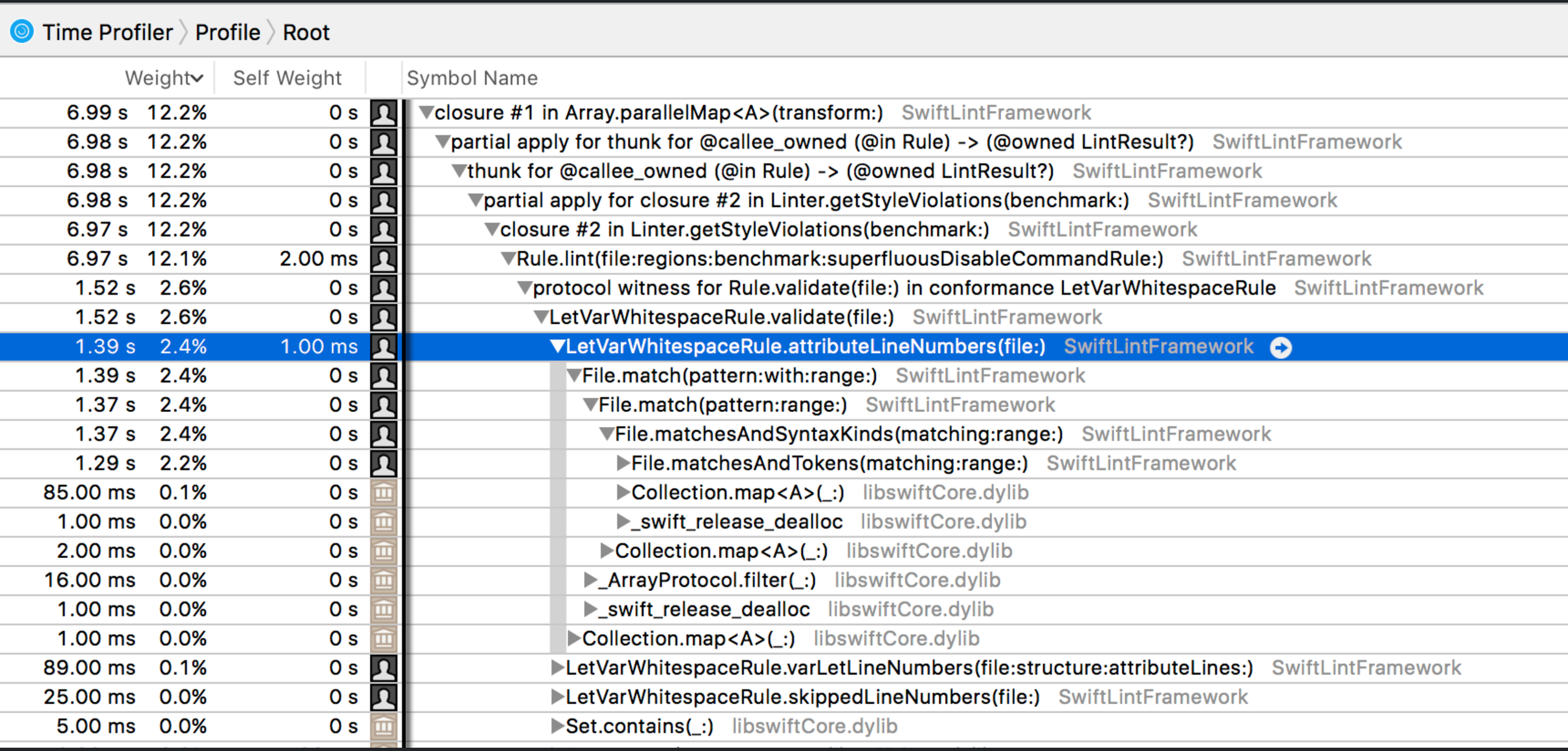
Rule Duration (Line)

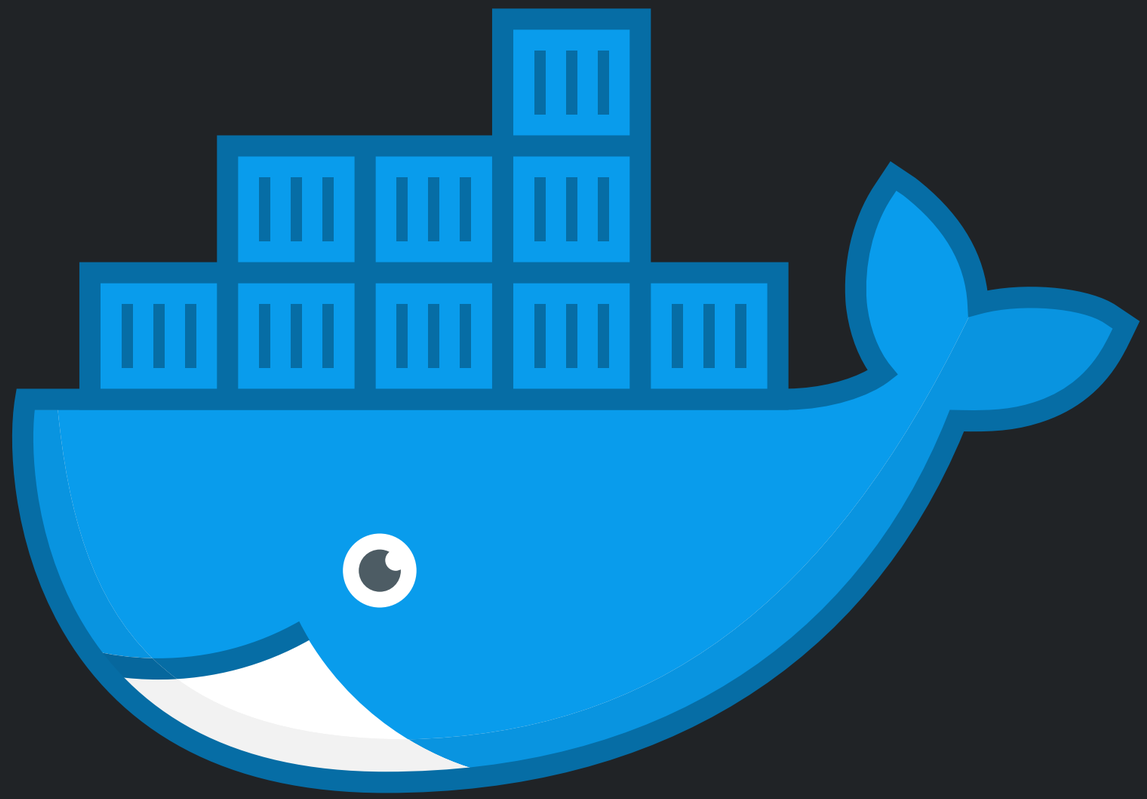


Rule Duration (Pie)




Instruments
jsimard-mbp133 〉 swiftlint.app
Run 1 of 1 | 00:00:12
Track Filter
All
Instruments
Threads
CPUs
00:00.000
00:01.000
00:02.000
00:03.000
00:04.000
00:05.000
00:06.000
00:07.000
00:08.000
00:09.000
00:10.000
00:11.000
CPU Usage
Life Cycle
Initializing
Main Thread 0x3d9598
swiftlint (76497)
_dispatch_worker…read3 0x3d95d7
swiftlint (76497)
Time Profiler 〉 Profile 〉 Root
Weight
Self Weight
Symbol Name
6.99 s 12.2% 0 s closure #1 in Array.parallelMap<A>(transform:) SwiftLintFramework
6.98 s 12.2% 0 s partial apply for thunk for @callee_owned (@in Rule) -> (@owned LintResult?) SwiftLintFramework
6.98 s 12.2% 0 s thunk for @callee_owned (@in Rule) -> (@owned LintResult?) SwiftLintFramework
6.98 s 12.2% 0 s partial apply for closure #2 in Linter.getStyleViolations(benchmark:) SwiftLintFramework
6.97 s 12.2% 0 s closure #2 in Linter.getStyleViolations(benchmark:) SwiftLintFramework
6.97 s 12.1% 2.00 ms Rule.lint(file:regions:benchmark:superfluousDisableCommandRule:) SwiftLintFramework
1.52 s 2.6% 0 s protocol witness for Rule.validate(file:) in conformance LetVarWhitespaceRule SwiftLintFramework
1.52 s 2.6% 0 s LetVarWhitespaceRule.validate(file:) SwiftLintFramework
1.39 s 2.4% 1.00 ms LetVarWhitespaceRule.attributeLineNumbers(file:) SwiftLintFramework
1.39 s 2.4% 0 s File.match(pattern:with:range:) SwiftLintFramework
1.37 s 2.4% 0 s File.match(pattern:range:) SwiftLintFramework
1.37 s 2.4% 0 s File.matchesAndSyntaxKinds(matching:range:) SwiftLintFramework
1.29 s 2.2% 0 s File.matchesAndTokens(matching:range:) SwiftLintFramework
85.00 ms 0.1% 0 s Collection.map<A>(_:) libswiftCore.dylib
1.00 ms 0.0% 0 s _swift_release_dealloc libswiftCore.dylib
2.00 ms 0.0% 0 s Collection.map<A>(_:) libswiftCore.dylib
16.00 ms 0.0% 0 s _ArrayProtocol.filter(_:) libswiftCore.dylib
1.00 ms 0.0% 0 s _swift_release_dealloc libswiftCore.dylib
1.00 ms 0.0% 0 s Collection.map<A>(_:) libswiftCore.dylib
89.00 ms 0.1% 0 s LetVarWhitespaceRule.varLetLineNumbers(file:structure:attributeLines:) SwiftLintFramework
25.00 ms 0.0% 0 s LetVarWhitespaceRule.skippedLineNumbers(file:) SwiftLintFramework
5.00 ms 0.0% 0 s Set.contains(_:) libswiftCore.dylib
2.00 ms 0.0% 1.00 ms EnumeratedIterator.next() libswiftCore.dylib
2.00 ms 0.0% 0 s StringProtocol<A>.trimmingCharacters(in:) libswiftFoundation.dylib
1.00 ms 0.0% 0 s outlined copy of _VariantSetBuffer SwiftLintFramework
1.00 ms 0.0% 1.00 ms _CFRelease CoreFoundation
1.00 ms 0.0% 0 s Dictionary<A, B>.substructure.getter SwiftLintFramework
1.00 ms 0.0% 1.00 ms swift_unknownRelease libswiftCore.dylib
1.00 ms 0.0% 0 s static CharacterSet.whitespaces.getter libswiftFoundation.dylib
288.00 ms 0.5% 0 s protocol witness for Rule.validate(file:) in conformance VerticalParameterAlignmentOnCallRule SwiftLintFramework
260.00 ms 0.4% 0 s protocol witness for Rule.validate(file:) in conformance ColonRule SwiftLintFramework
170.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance ClosureEndIndentationRule SwiftLintFramework
154.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance MultipleClosuresWithTrailingClosureRule SwiftLintFramework
153.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance EmptyParenthesesWithTrailingClosureRule SwiftLintFramework
139.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance UnusedClosureParameterRule SwiftLintFramework
134.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance TrailingCommaRule SwiftLintFramework
132.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance ClosureParameterPositionRule SwiftLintFramework
128.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance NestingRule SwiftLintFramework
127.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance LargeTupleRule SwiftLintFramework
119.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance FunctionBodyLengthRule SwiftLintFramework
117.00 ms 0.2% 0 s protocol witness for Rule.validate(file:) in conformance RedundantVoidReturnRule SwiftLintFramework
Heaviest Stack Trace
57123.0 swiftlint (76497)
7499.0 _dispatch_worker_thread3...
7393.0 start_wqthread
7393.0 _pthread_wqthread
7393.0 _dispatch_worker_thread3
7393.0 _dispatch_root_queue_drain
7393.0 _dispatch_client_callout
7393.0 _dispatch_apply_invoke
7393.0 _dispatch_client_callout2
7393.0 thunk for @callee_owned (@...
7393.0 partial apply for closure #2 i...
7393.0 closure #2 in closure #2 in...
7393.0 partial apply for closure #1 i...
7393.0 closure #1 in closure #2 in...
7392.0 autoreleasepool<A>(invokin...
7381.0 partial apply for thunk for @...
7381.0 thunk for @callee_owned ()...
7381.0 partial apply for closure #2 i...
7381.0 closure #2 in closure #1 in c...
7378.0 partial apply for closure #1 i...
7378.0 closure #1 in LintCommand....
7370.0 Linter.styleViolations.getter
7370.0 Linter.getStyleViolations(be...
7033.0 Array.parallelFlatMap<A>(tr...
7020.0 Array.parallelMap<A>(transf...
6993.0 static DispatchQueue.concu...
6993.0 _swift_dispatch_apply_current
6993.0 dispatch_apply_f
6993.0 _dispatch_sync_function_inv...
6993.0 _dispatch_client_callout
6993.0 _dispatch_apply_serial
6993.0 _dispatch_client_callout2
6991.0 thunk for @callee_owned (@...
6991.0 partial apply for closure #1 i...
6990.0 closure #1 in Array.parallel...
6976.0 partial apply for thunk for @...
6976.0 thunk for @callee_owned (@...
6976.0 partial apply for closure #2 i...
Input Filter
Involves Symbol
Call Tree
Call Tree Constraints
Data Mining


Time Profiler 〉 Profile 〉 Root
Weight
Self Weight
Symbol Name
6.99 s 12.2% 0 s closure #1 in Array.parallelMap<A>(transform:) SwiftLintFramework
6.98 s 12.2% 0 s partial apply for thunk for @callee_owned (@in Rule) -> (@owned LintResult?) SwiftLintFramework
6.98 s 12.2% 0 s thunk for @callee_owned (@in Rule) -> (@owned LintResult?) SwiftLintFramework
6.98 s 12.2% 0 s partial apply for closure #2 in Linter.getStyleViolations(benchmark:) SwiftLintFramework
6.97 s 12.2% 0 s closure #2 in Linter.getStyleViolations(benchmark:) SwiftLintFramework
6.97 s 12.1% 2.00 ms Rule.lint(file:regions:benchmark:superfluousDisableCommandRule:) SwiftLintFramework
1.52 s 2.6% 0 s protocol witness for Rule.validate(file:) in conformance LetVarWhitespaceRule SwiftLintFramework
1.52 s 2.6% 0 s LetVarWhitespaceRule.validate(file:) SwiftLintFramework
1.39 s 2.4% 1.00 ms LetVarWhitespaceRule.attributeLineNumbers(file:) SwiftLintFramework
1.39 s 2.4% 0 s File.match(pattern:with:range:) SwiftLintFramework
1.37 s 2.4% 0 s File.match(pattern:range:) SwiftLintFramework
1.37 s 2.4% 0 s File.matchesAndSyntaxKinds(matching:range:) SwiftLintFramework
1.29 s 2.2% 0 s File.matchesAndTokens(matching:range:) SwiftLintFramework
85.00 ms 0.1% 0 s Collection.map<A>(_:) libswiftCore.dylib
1.00 ms 0.0% 0 s _swift_release_dealloc libswiftCore.dylib
2.00 ms 0.0% 0 s Collection.map<A>(_:) libswiftCore.dylib
16.00 ms 0.0% 0 s _ArrayProtocol.filter(_:) libswiftCore.dylib
1.00 ms 0.0% 0 s _swift_release_dealloc libswiftCore.dylib
1.00 ms 0.0% 0 s Collection.map<A>(_:) libswiftCore.dylib
89.00 ms 0.1% 0 s LetVarWhitespaceRule.varLetLineNumbers(file:structure:attributeLines:) SwiftLintFramework
25.00 ms 0.0% 0 s LetVarWhitespaceRule.skippedLineNumbers(file:) SwiftLintFramework
5.00 ms 0.0% 0 s Set.contains(_:) libswiftCore.dylib

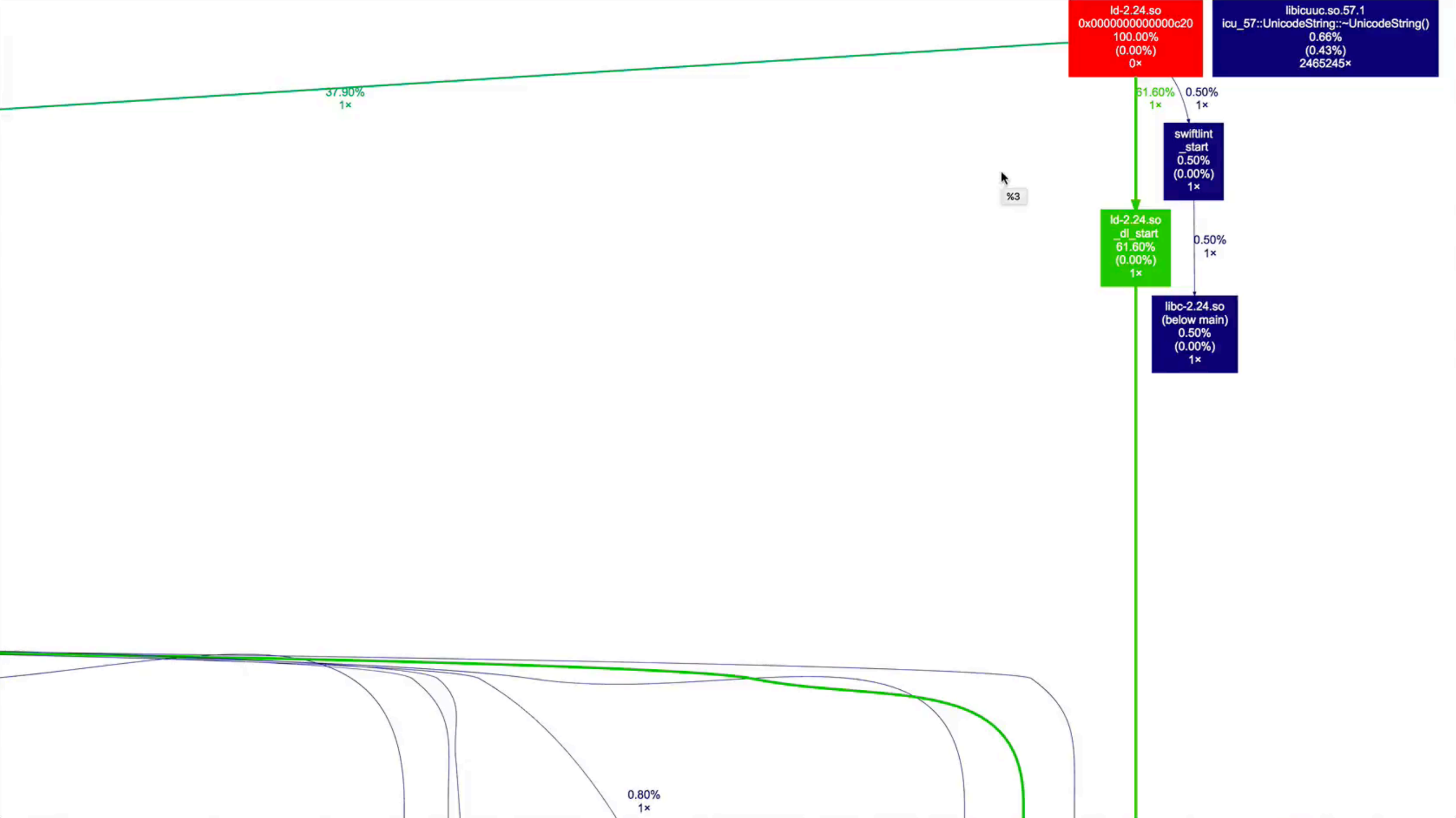


docker

```
$ docker run -it --rm swift bash
$ > apt-get update && apt-get install -y valgrind
$ > git clone https://github.com/realm/SwiftLint.git
$ > cd SwiftLint
$ > swift build
$ > valgrind --tool=callgrind .build/debug/swiftlint
$ brew install qcachegrind gprof2dot
$ gprof2dot -f callgrind callgrind.out.*
    | dot -Tsvg -o dotgraph.svg
$ open -a Safari dotgraph.svg
$ open -a qcachegrind
```

```
$ docker run -it --rm swift bash
$ > apt-get update && apt-get install -y valgrind
$ > git clone https://github.com/realm/SwiftLint.git
$ > cd SwiftLint
$ > swift build
$ > valgrind --tool=callgrind .build/debug/swiftlint
$ brew install qcachegrind gprof2dot
$ gprof2dot -f callgrind callgrind.out.*
    | dot -Tsvg -o dotgraph.svg
$ open -a Safari dotgraph.svg
$ open -a qcachegrind
```

```
$ docker run -it --rm swift bash
$ > apt-get update && apt-get install -y valgrind
$ > git clone https://github.com/realm/SwiftLint.git
$ > cd SwiftLint
$ > swift build
$ > valgrind --tool=callgrind .build/debug/swiftlint
$ brew install qcachegrind gprof2dot
$ gprof2dot -f callgrind callgrind.out.*
    | dot -Tsvg -o dotgraph.svg
$ open -a Safari dotgraph.svg
$ open -a qcachegrind
```


ld-2.24.so
0x0000000000000c20
100.00%
(0.00%)
0×
libicuuc.so.57.1
icu_57::UnicodeString::~UnicodeString()
0.66%
(0.43%)
2465245×
37.90%
1×
61.60%
1×
0.50%
1×
swiftlint
_start
0.50%
(0.00%)
1×
%3
ld-2.24.so
_dl_start
61.60%
(0.00%)
1×
0.50%
1×
libc-2.24.so
(below main)
0.50%
(0.00%)
1×
0.80%
1×

```
$ docker run -it --rm swift bash
$ > apt-get update && apt-get install -y valgrind
$ > git clone https://github.com/realm/SwiftLint.git
$ > cd SwiftLint
$ > swift build
$ > valgrind --tool=callgrind .build/debug/swiftlint
$ brew install qcachegrind gprof2dot
$ gprof2dot -f callgrind callgrind.out.*
    | dot -Tsvg -o dotgraph.svg
$ open -a Safari dotgraph.svg
$ open -a qcachegrind
```

/Users/jsimard/Projects/talks/swift_linux_perf/callgrind.out.27441 [.build/x86_64-unknown-linux/debug/swiftlint lint --path Source/swiftlint]

Instruction Fetch

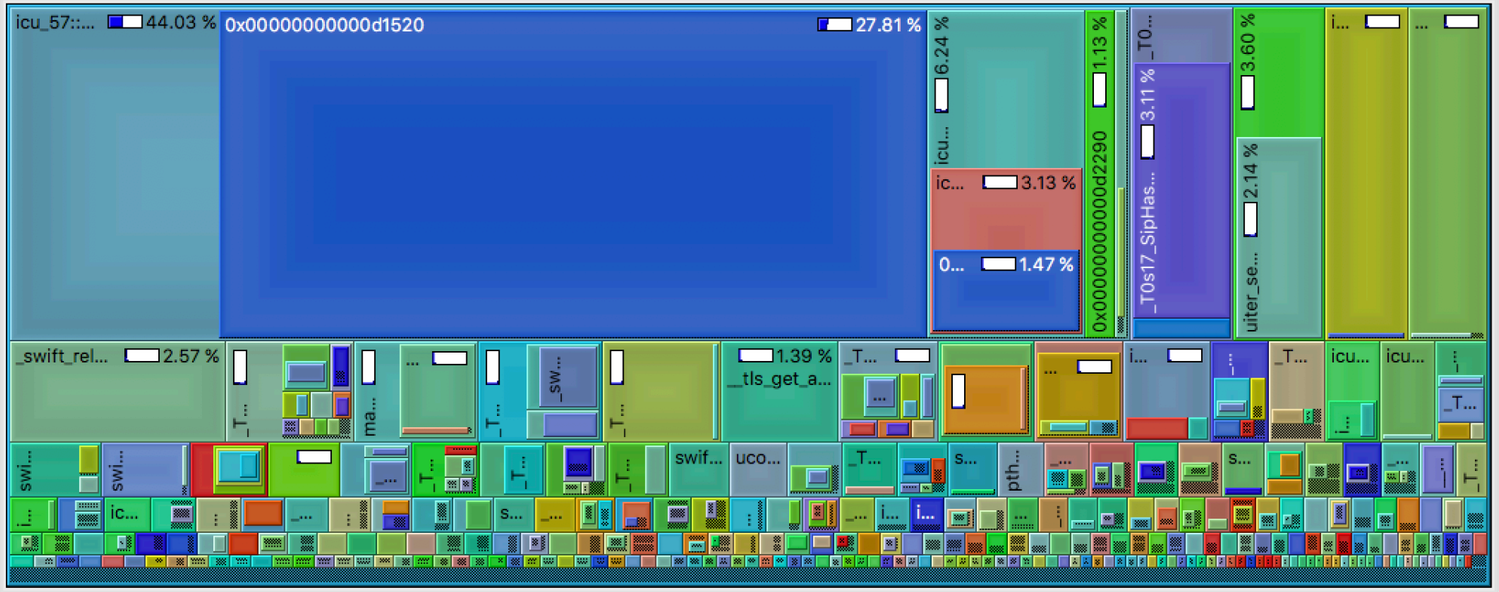
Flat Profile

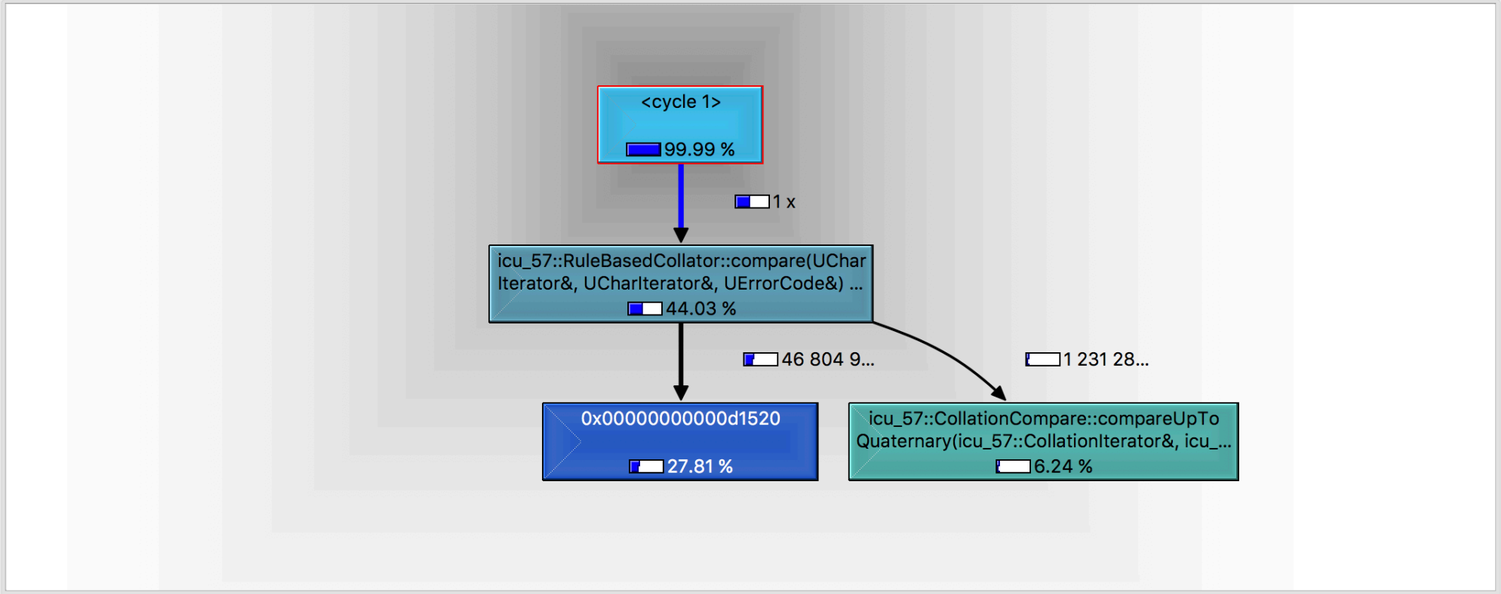
Search: (No Grouping)

Incl.	Self	Called	Function	Location
99.99	44.16	81	<cycle 1>	libsourcekitdInProc.so
44.03	8.34	1 328 540	icu_57::RuleBasedCollator:...	libicui18n.so.57.1
29.28	29.28	49 276 263	0x00000000000d1520	libicuuc.so.57.1
6.24	3.11	1 231 308	icu_57::CollationCompare::...	libicui18n.so.57.1
4.05	0.72	131 910	_T0s7UnicodeO9hashASCI...	libswiftCore.so
3.60	1.45	1 328 382	_swift_stdlib_unicode_com...	libswiftCore.so
3.27	3.27	151 361	icu_57::RegexMatcher::Ma...	libicui18n.so.57.1
3.13	1.66	2 471 452	icu_57::FCDUIterCollationIt...	libicui18n.so.57.1
3.11	3.11	2 148 814	_T0s17_SipHash13Context...	libswiftCore.so
3.09	3.09	1 333 731	swift_getGenericMetadata ...	libswiftCore.so
2.57	2.57	12 277 817	_swift_release_ <cycle 1>	libswiftCore.so
2.15	2.15	10 904 773	_swift_retain_	libswiftCore.so
2.14	2.14	2 656 922	uiter_setUTF8_57	libicuuc.so.57.1
1.53	0.73	266 716	_T0s16IndexingIteratorV4n...	libswiftCore.so
1.49	0.61	621 095	malloc <cycle 1>	libc-2.24.so: malloc.c, arer
1.48	0.63	20 764	_T021SourceKittenFramew...	swiftlint: SwiftDeclarationK
1.41	1.41	1 332 163	_T0SS37_compareDetermi...	libswiftCore.so
1.39	1.39	6 577 225	__tls_get_addr <cycle 1>	ld-2.24.so: dl-tls.c
1.28	0.29	5 050 173	swift_rt_swift_retain	swiftlint
1.22	0.34	610 624	free	libc-2.24.so: malloc.c
1.19	0.44	3 475	_T0s10CollectionPsE5inde...	libswiftCore.so
1.14	0.02	563 612	swift_slowDealloc <cycle 1>	libswiftCore.so
1.13	1.13	2 461 210	0x00000000000d2290	libicuuc.so.57.1
1.13	0.19	4 846 653	swift_rt_swift_retain	libswiftCore.so
1.06	0.15	15 999	_dl_lookup_symbol_x <cycl...	ld-2.24.so: dl-lookup.c
1.06	0.51	2 038 588	_T0Sawcp	libswiftCore.so
1.04	0.85	2 463 087	icu_57::CollationIterator::~...	libicui18n.so.57.1
0.91	0.79	26 070	do_lookup_x	ld-2.24.so: dl-lookup.c, lds
0.89	0.86	622 552	_int_malloc	libc-2.24.so: malloc.c
0.88	0.88	610 066	_int_free	libc-2.24.so: malloc.c, arer
0.80	0.64	958 837	swift_getEnumCaseSingle...	libswiftCore.so
0.75	0.05	826 648	swift_rt_swift_getEnumCas...	libswiftCore.so
0.70	0.28	475 530	_T0Sa9subscriptxSicfg <c...	libswiftCore.so
0.67	0.49	90 604	_T0s16StrideToIteratorV4n...	libswiftCore.so
0.66	0.43	2 465 245	icu_57::UnicodeString::~U...	libicuuc.so.57.1
0.66	0.66	2 256	icu_57::RegexMatcher::fin...	libicui18n.so.57.1
0.62	0.24	8 699	_T010Foundation8NSStrin...	libFoundation.so
0.60	0.50	578 679	swift_storeEnumTagSingle...	libswiftCore.so
0.58	0.58	285 356	swift_getGenericWitnessT...	libswiftCore.so
0.55	0.00	(0)	0x0000000000000c20	ld-2.24.so
0.55	0.31	2 234 812	icu_57::UnicodeSet::contai...	libicuuc.so.57.1
0.51	0.18	342	_T0s7UnicodeO9hashUTF...	libswiftCore.so
0.47	0.47	2 462 568	icu_57::FCDUIterCollationIt...	libicui18n.so.57.1

<cycle 1>

Types | Callers | All Callers | Callee Map | Source Code





Parts | Callees | Call Graph | All Callees | Caller Map | Machine Code

callgrind.out.27441 [1] - Total Instruction Fetch Cost: 5 205 159 048

Bare Metal 🤘

```
$ doctl compute droplet create perf --size 16gb \
    --image ubuntu-16-10-x64 --region sfo1
$ > apt-get update && apt-get dist-upgrade
$ > reboot
$ > apt-get install -y clang libblocksruntime0 libcurl4-openssl-dev \
      linux-tools-common linux-tools-generic linux-tools-`uname -r`
$ > SWIFT_VERSION=swift-4.0-RELEASE
$ > BASE_URL=https://swift.org/builds/swift-4.0-release/ubuntu1610
$ > URL=$BASE_URL/$SWIFT_VERSION/$SWIFT_VERSION-ubuntu16.10.tar.gz
$ > curl $URL | tar xz --directory $HOME --strip-components=1
$ > export PATH=$HOME/usr/bin:$PATH
$ > export LINUX_SOURCEKIT_LIB_PATH=$HOME/usr/lib
$ > git clone https://github.com/realm/SwiftLint.git
$ > git clone https://github.com/brendangregg/FlameGraph.git
$ > cd SwiftLint
$ > swift build
$ > perf record -g .build/debug/swiftlint
$ > perf script > out.perf
$ > ../FlameGraph/stackcollapse-perf.pl out.perf > out.folded
$ > ../FlameGraph/flamegraph.pl out.folded > flamegraph.svg
```

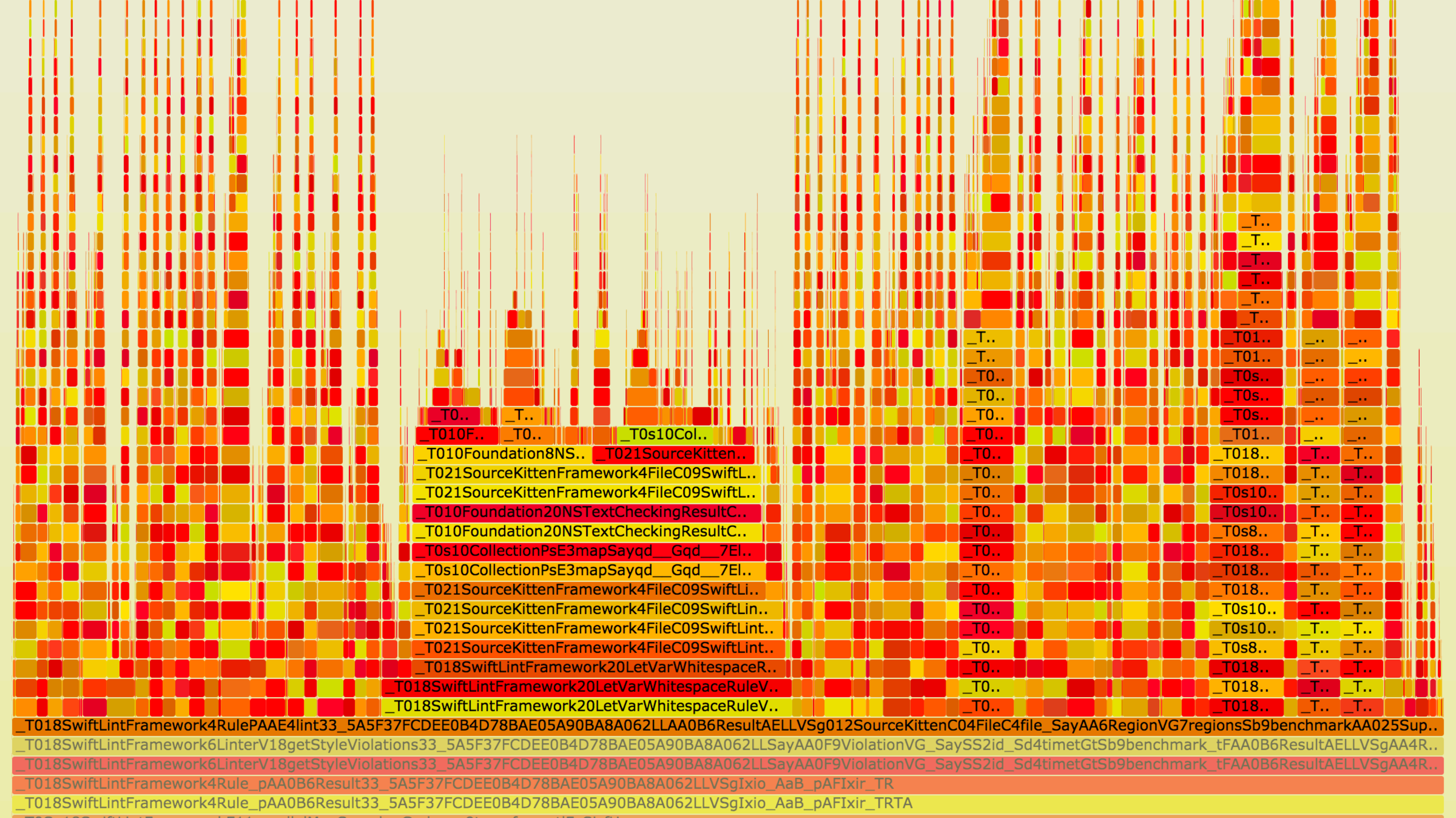
```
$ doctl compute droplet create perf --size 16gb \
    --image ubuntu-16-10-x64 --region sfo1
$ > apt-get update && apt-get dist-upgrade
$ > reboot
$ > apt-get install -y clang libblocksruntime0 libcurl4-openssl-dev \
      linux-tools-common linux-tools-generic linux-tools-`uname -r`
$ > SWIFT_VERSION=swift-4.0-RELEASE
$ > BASE_URL=https://swift.org/builds/swift-4.0-release/ubuntu1610
$ > URL=$BASE_URL/$SWIFT_VERSION/$SWIFT_VERSION-ubuntu16.10.tar.gz
$ > curl $URL | tar xz --directory $HOME --strip-components=1
$ > export PATH=$HOME/usr/bin:$PATH
$ > export LINUX_SOURCEKIT_LIB_PATH=$HOME/usr/lib
$ > git clone https://github.com/realm/SwiftLint.git
$ > git clone https://github.com/brendangregg/FlameGraph.git
$ > cd SwiftLint
$ > swift build
$ > perf record -g .build/debug/swiftlint
$ > perf script > out.perf
$ > ../FlameGraph/stackcollapse-perf.pl out.perf > out.folded
$ > ../FlameGraph/flamegraph.pl out.folded > flamegraph.svg
```

```
$ doctl compute droplet create perf --size 16gb \
    --image ubuntu-16-10-x64 --region sfo1
$ > apt-get update && apt-get dist-upgrade
$ > reboot
$ > apt-get install -y clang libblocksruntime0 libcurl4-openssl-dev \
      linux-tools-common linux-tools-generic linux-tools-`uname -r`
$ > SWIFT_VERSION=swift-4.0-RELEASE
$ > BASE_URL=https://swift.org/builds/swift-4.0-release/ubuntu1610
$ > URL=$BASE_URL/$SWIFT_VERSION/$SWIFT_VERSION-ubuntu16.10.tar.gz
$ > curl $URL | tar xz --directory $HOME --strip-components=1
$ > export PATH=$HOME/usr/bin:$PATH
$ > export LINUX_SOURCEKIT_LIB_PATH=$HOME/usr/lib
$ > git clone https://github.com/realm/SwiftLint.git
$ > git clone https://github.com/brendangregg/FlameGraph.git
$ > cd SwiftLint
$ > swift build
$ > perf record -g .build/debug/swiftlint
$ > perf script > out.perf
$ > ../FlameGraph/stackcollapse-perf.pl out.perf > out.folded
$ > ../FlameGraph/flamegraph.pl out.folded > flamegraph.svg
```

```
$ doctl compute droplet create perf --size 16gb \
    --image ubuntu-16-10-x64 --region sfo1
$ > apt-get update && apt-get dist-upgrade
$ > reboot
$ > apt-get install -y clang libblocksruntime0 libcurl4-openssl-dev \
      linux-tools-common linux-tools-generic linux-tools-`uname -r`
$ > SWIFT_VERSION=swift-4.0-RELEASE
$ > BASE_URL=https://swift.org/builds/swift-4.0-release/ubuntu1610
$ > URL=$BASE_URL/$SWIFT_VERSION/$SWIFT_VERSION-ubuntu16.10.tar.gz
$ > curl $URL | tar xz --directory $HOME --strip-components=1
$ > export PATH=$HOME/usr/bin:$PATH
$ > export LINUX_SOURCEKIT_LIB_PATH=$HOME/usr/lib
$ > git clone https://github.com/realm/SwiftLint.git
$ > git clone https://github.com/brendangregg/FlameGraph.git
$ > cd SwiftLint
$ > swift build
$ > perf record -g .build/debug/swiftlint
$ > perf script > out.perf
$ > ../FlameGraph/stackcollapse-perf.pl out.perf > out.folded
$ > ../FlameGraph/flamegraph.pl out.folded > flamegraph.svg
```

```
$ doctl compute droplet create perf --size 16gb \
    --image ubuntu-16-10-x64 --region sfo1
$ > apt-get update && apt-get dist-upgrade
$ > reboot
$ > apt-get install -y clang libblocksruntime0 libcurl4-openssl-dev \
      linux-tools-common linux-tools-generic linux-tools-`uname -r`
$ > SWIFT_VERSION=swift-4.0-RELEASE
$ > BASE_URL=https://swift.org/builds/swift-4.0-release/ubuntu1610
$ > URL=$BASE_URL/$SWIFT_VERSION/$SWIFT_VERSION-ubuntu16.10.tar.gz
$ > curl $URL | tar xz --directory $HOME --strip-components=1
$ > export PATH=$HOME/usr/bin:$PATH
$ > export LINUX_SOURCEKIT_LIB_PATH=$HOME/usr/lib
$ > git clone https://github.com/realm/SwiftLint.git
$ > git clone https://github.com/brendangregg/FlameGraph.git
$ > cd SwiftLint
$ > swift build
$ > perf record -g .build/debug/swiftlint
$ > perf script > out.perf
$ > ../FlameGraph/stackcollapse-perf.pl out.perf > out.folded
$ > ../FlameGraph/flamegraph.pl out.folded > flamegraph.svg
```

Demo


_T..
_T..
_T..
_T..
_T..
_T..
_T..
_T01..
_..
_..
_T..
_T01..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T..
_T0..
_T010F..
_T0..
_T0s10Col..
_T0..
_T01..
_..
_..
_T010Foundation8NS..
_T021SourceKitten..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftL..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftL..
_T0..
_T0s10..
_T..
_T..
_T010Foundation20NSTextCheckingResultC..
_T0..
_T0s10..
_T..
_T..
_T010Foundation20NSTextCheckingResultC..
_T0..
_T0s8..
_T..
_T..
_T0s10CollectionPsE3mapSayqd__Gqd__7El..
_T0..
_T018..
_T..
_T..
_T0s10CollectionPsE3mapSayqd__Gqd__7El..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLi..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLin..
_T0..
_T0s10..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLint..
_T0..
_T0s10..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLint..
_T0..
_T0s8..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceR..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceRuleV..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceRuleV..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework4RulePAAE4lint33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLAA0B6ResultAELLVSg012SourceKittenC04FileC4file_SayAA6RegionVG7regionsSb9benchmarkAA025Sup..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TR
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TRTA

```
$ xcrun swift-demangle
_T018SwiftLintFramework20LetVarWhitespaceRuleV20att
ributeLineNumbers33_013BAF1EF367799B68F2E028EAD9
BE9DLLs3SetVySiG012SourceKittenC04FileC4file_tF

//
SwiftLintFramework.
  //
  LetVarWhitespaceRule.
    //
    (attributeLineNumbers in \_013BAF1EF367799B68F2E028EAD9BE9D)
      //
      (file : SourceKittenFramework.File)
        //
        -> Swift.Set<Swift.Int>
```

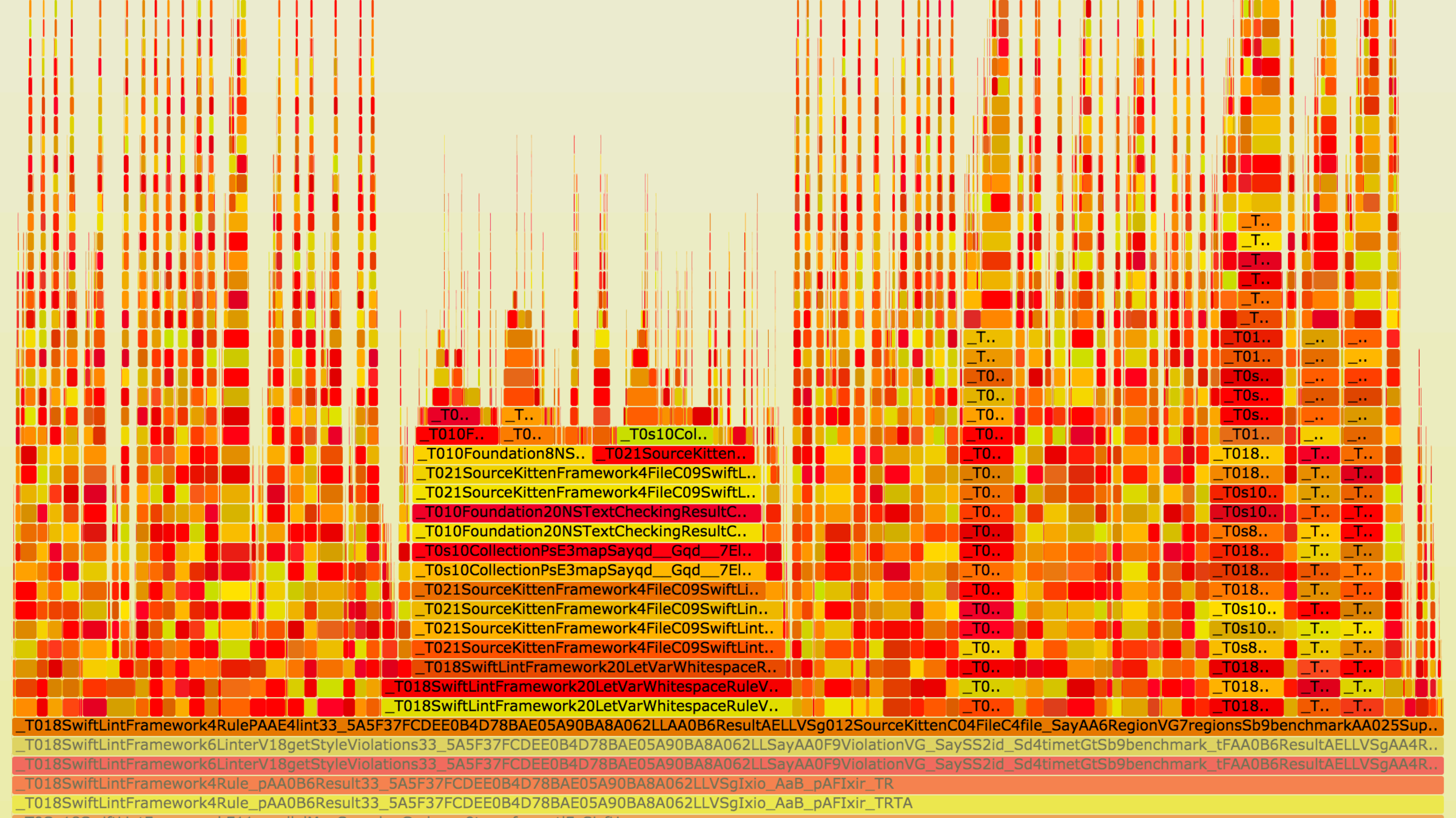
```
$ xcrun swift-demangle
_T018SwiftLintFramework20LetVarWhitespaceRuleV20att
ributeLineNumbers33_013BAF1EF367799B68F2E028EAD9
BE9DLLs3SetVySiG012SourceKittenC04FileC4file_tF

//
SwiftLintFramework.
  //
  LetVarWhitespaceRule.
    //
    (attributeLineNumbers in \_013BAF1EF367799B68F2E028EAD9BE9D)
      //
      (file : SourceKittenFramework.File)
        //
        -> Swift.Set<Swift.Int>
```

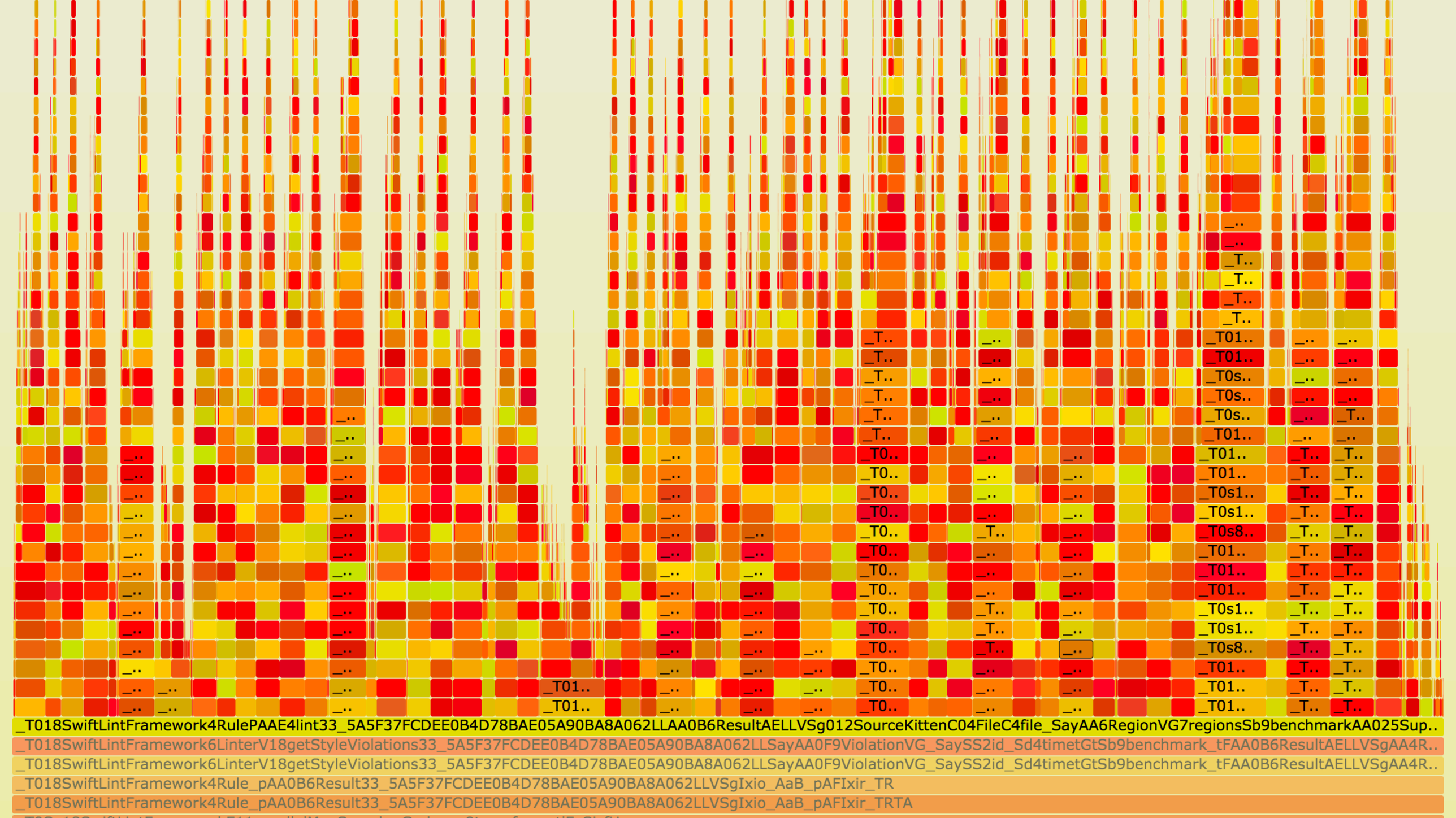
```
$ xcrun swift-demangle
_T018SwiftLintFramework20LetVarWhitespaceRuleV20att
ributeLineNumbers33_013BAF1EF367799B68F2E028EAD9
BE9DLLs3SetVySiG012SourceKittenC04FileC4file_tF

// Module
SwiftLintFramework.
  // Type (struct)
  LetVarWhitespaceRule.
    // Function
    (attributeLineNumbers in \_013BAF1EF367799B68F2E028EAD9BE9D)
      // Parameter
      (file : SourceKittenFramework.File)
        // Return type
        -> Swift.Set<Swift.Int>
```

```swift
public struct LetVarWhitespaceRule: ConfigurationProviderRule, OptInRule {

    /* ... */

    // Collects all the line numbers containing attributes but not declarations
    // other than let/var
    private func attributeLineNumbers(file: File) -> Set<Int> {
        let matches = file.match(pattern: "[@_a-z]+", with: [.attributeBuiltin])
        let matchLines = matches.map { file.line(offset: $0.location) }

        return Set<Int>(matchLines)
    }

    /* ... */

}
```

$ git diff

```
  // Collects all the line numbers containing attributes but not declarations
  // other than let/var
  private func attributeLineNumbers(file: File) -> Set<Int> {
-     let matches = file.match(pattern: "[@_a-z]+", with: [.attributeBuiltin])
-     let matchLines = matches.map { file.line(offset: $0.location) }
-
-     return Set<Int>(matchLines)
+     return Set(file.syntaxMap.tokens.flatMap({ token in
+         if token.type == SyntaxKind.attributeBuiltin.rawValue {
+             return file.line(byteOffset: token.offset)
+         }
+         return nil
+     }))
  }
```


_T..
_T..
_T..
_T..
_T..
_T..
_T..
_T01..
_..
_..
_T..
_T01..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T0s..
_..
_..
_T0..
_T..
_T0..
_T010F..
_T0..
_T0s10Col..
_T0..
_T01..
_..
_..
_T010Foundation8NS..
_T021SourceKitten..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftL..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftL..
_T0..
_T0s10..
_T..
_T..
_T010Foundation20NSTextCheckingResultC..
_T0..
_T0s10..
_T..
_T..
_T010Foundation20NSTextCheckingResultC..
_T0..
_T0s8..
_T..
_T..
_T0s10CollectionPsE3mapSayqd__Gqd__7El..
_T0..
_T018..
_T..
_T..
_T0s10CollectionPsE3mapSayqd__Gqd__7El..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLi..
_T0..
_T018..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLin..
_T0..
_T0s10..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLint..
_T0..
_T0s10..
_T..
_T..
_T021SourceKittenFramework4FileC09SwiftLint..
_T0..
_T0s8..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceR..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceRuleV..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework20LetVarWhitespaceRuleV..
_T0..
_T018..
_T..
_T..
_T018SwiftLintFramework4RulePAAE4lint33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLAA0B6ResultAELLVSg012SourceKittenC04FileC4file_SayAA6RegionVG7regionsSb9benchmarkAA025Sup..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TR
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TRTA


_T018SwiftLintFramework4RulePAAE4lint33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLAA0B6ResultAELLVSg012SourceKittenC04FileC4file_SayAA6RegionVG7regionsSb9benchmarkAA025Sup..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework6LinterV18getStyleViolations33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLSayAA0F9ViolationVG_SaySS2id_Sd4timetGtSb9benchmark_tFAA0B6ResultAELLVSgAA4R..
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TR
_T018SwiftLintFramework4Rule_pAA0B6Result33_5A5F37FCDEE0B4D78BAE05A90BA8A062LLVSgIxio_AaB_pAFIxir_TRTA

Tools

- Perf
- Valgrind
- Callgrind
- GNU gprof
- KCachegrind
- FlameGraph
- gprof2dot

Closing Thoughts

— Consider writing an app-specific benchmark mode
— Prefer measuring with Instruments.app if possible
— Prefer measuring on: bare metal > container > VM
— Prefer measuring your code: release > debug
— Callgrind is extremely slow, try to avoid it
— There's a tooling opportunity to convert perf/callgrind data to something Instruments compatible

Thank You!

Questions?

JP Simard – @simjp – jp@lyft.com