


LAST NIGHT I DRIFTED OFF WHILE READING A LISP BOOK



SUDDENLY, I WAS BATHED IN A SUFFUSION OF BLUE.

AT ONCE, JUST LIKE THEY SAID, I FELT A GREAT ENLIGHTENMENT. I SAW THE NAKED STRUCTURE OF LISP CODE UNFOLD BEFORE ME.



THE PATTERNS AND METAPATTERNS DANCED. SYNTAX FADED, AND I SWAM IN THE PURITY OF QUANTIFIED CONCEPTION. OF IDEAS MANIFEST.

TRULY, THIS WAS THE LANGUAGE FROM WHICH THE GODS WROUGHT THE UNIVERSE



NO, IT'S NOT.



IT'S NOT?

I MEAN, OSTENSIBLY, YES. HONESTLY, WE HACKED MOST OF IT TOGETHER WITH PERL.

Perl

Baseless Myths & Startling Realities

by Tim Bunce, July 2008

Prefer ‘Good Developers’ over ‘Good Languages’

“For all program aspects investigated, **the performance variability that derives from differences among programmers** of the same language—as described by the bad-to-good ratios—**is on average as large or larger than the variability found among the different languages.**”

— An empirical comparison of C, C++, Java, Perl, Python, Rexx, and Tcl. IEEE Computer Journal October 2000

Who am I?



Tim Bunce

Author of the Perl DBI module

Using Perl since 1991

Involved in the development of Perl 5

“Pumpkin” for 5.4.x maintenance releases

<http://blog.timbunce.org>

~ Myths ~



~ Myths ~

Perl is dead

Perl is hard to read / test / maintain

Perl 6 is killing Perl 5

~ Myths ~

Perl is dead

Perl is hard to read / test / maintain

Perl 6 is killing Perl 5



Perl 5



Perl 5 isn't the new kid on the block

Perl is 21 years old

Perl 5 is 14 years old

A mature language with a mature culture



Perl 0: embryo

Perl 1: infant

Perl 2: toddler

Perl 3: child

Perl 4: preteen

Perl 5: adolescent

Buzz != Jobs

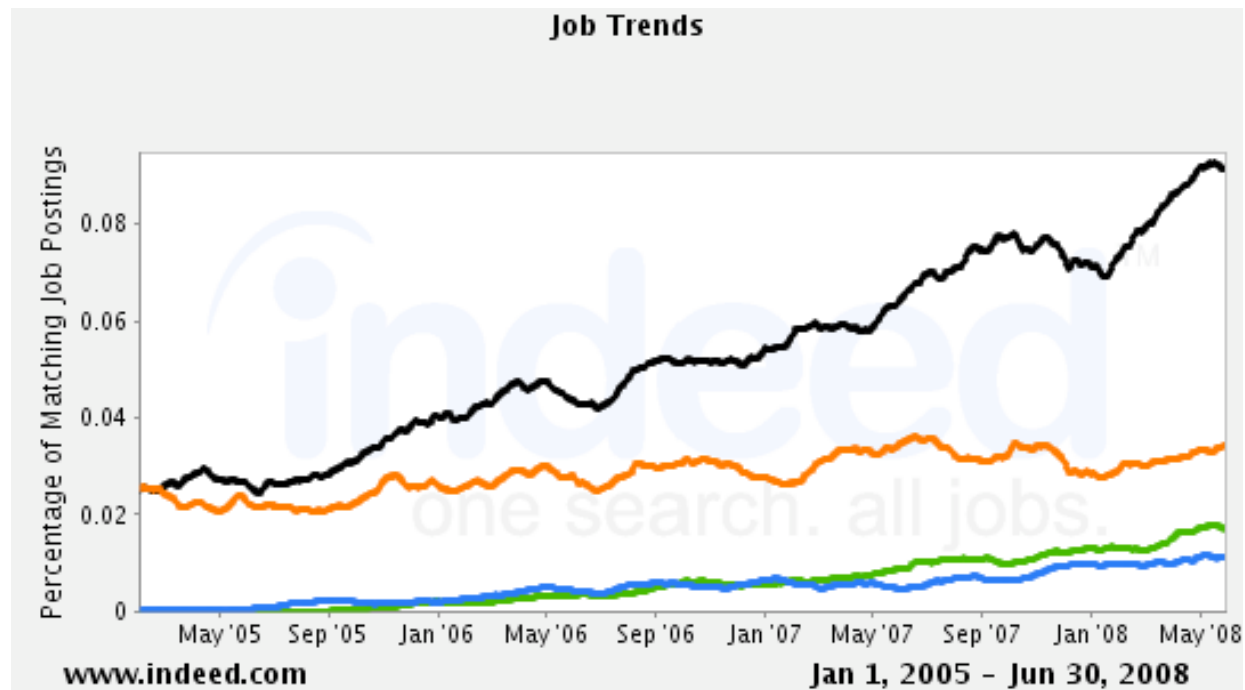
Perl5 hasn't been generating buzz recently

It's just getting on with the job

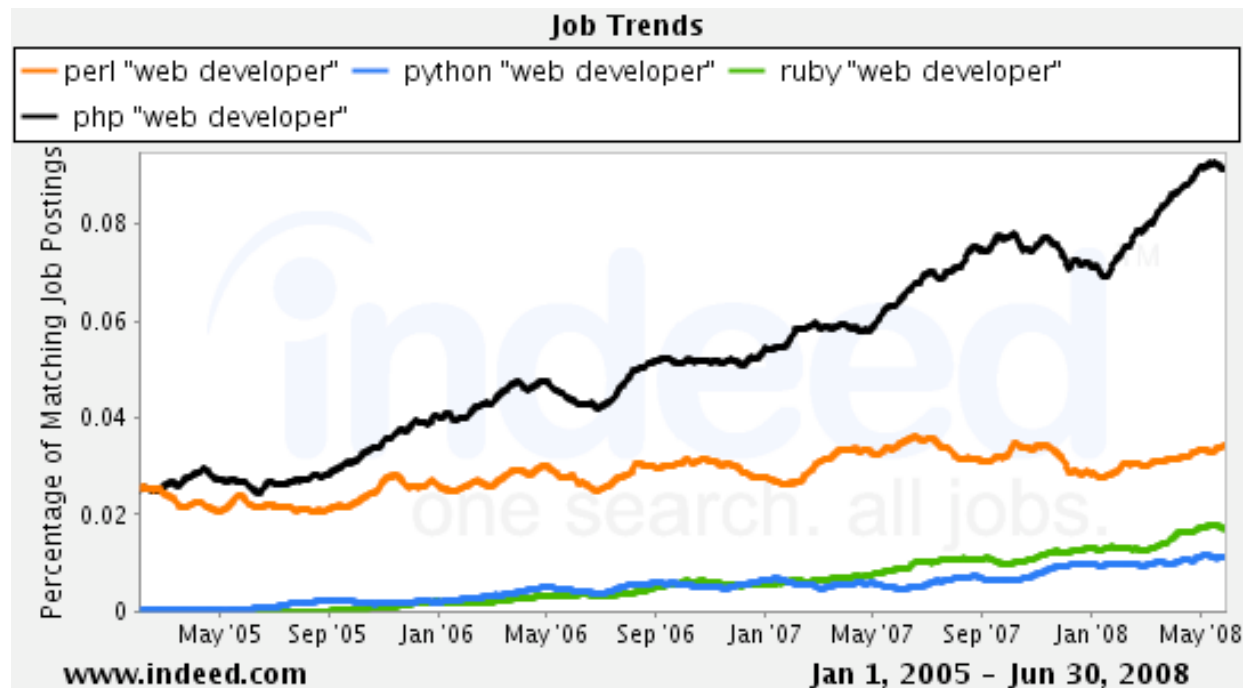
Lots of jobs

- just not all in web development

Guess the Languages

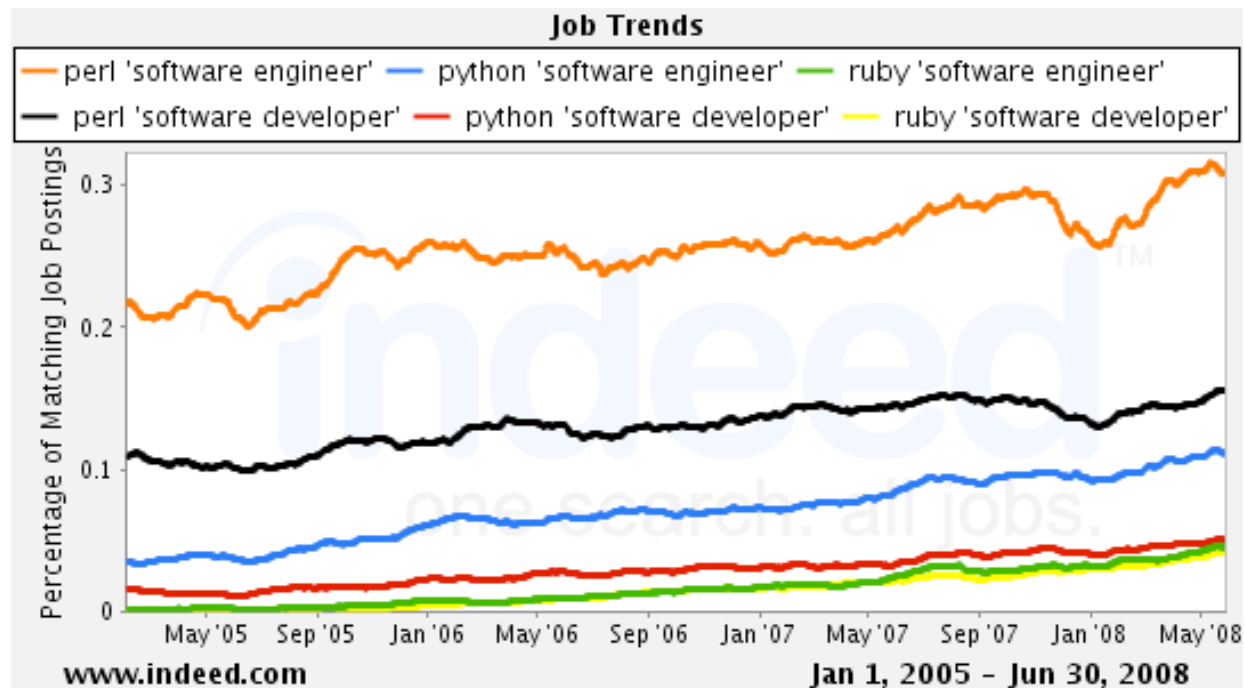


“web developer”



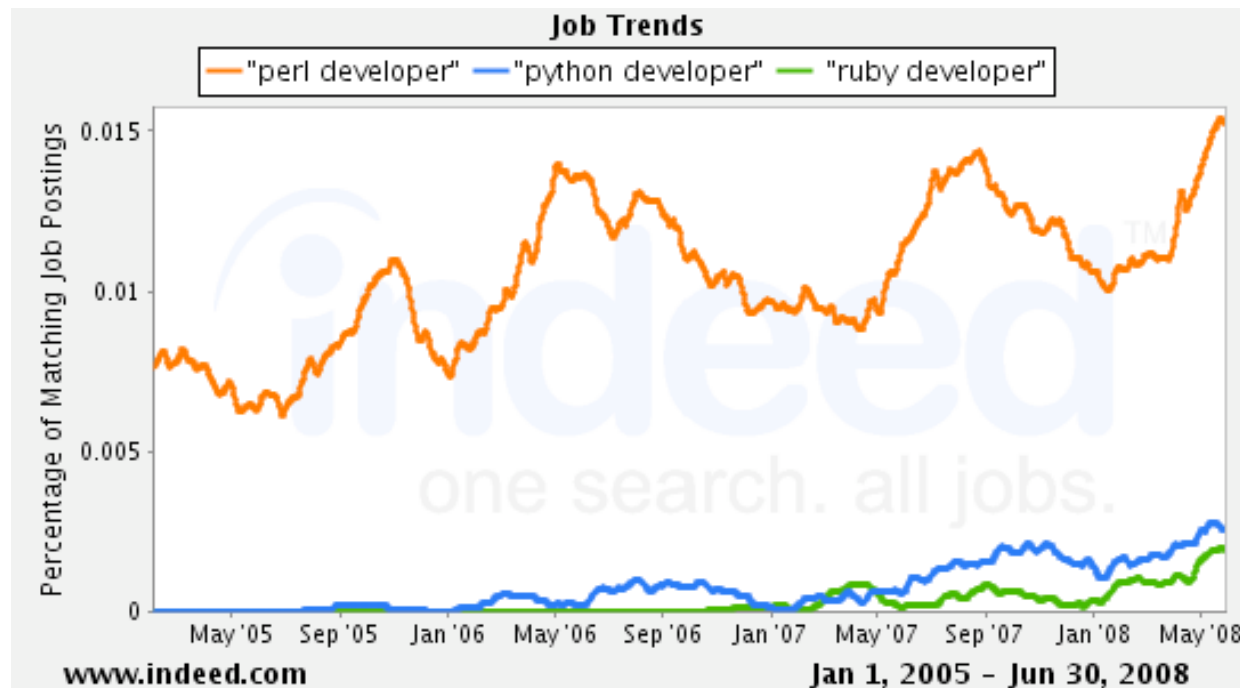
Yes, Perl is growing more slowly than others
but these are just “web developer” jobs

“software engineer”



Perl is mentioned in many more software engineer/developer jobs.

“foo developer”



Perl is the primary focus of more developer jobs.
Want a fun new job? Become a Perl developer!

Massive Module Market

Large and vibrant developer community

Over 15,000 distributions (58,000 modules)

Over 6,700 'authors' (who make releases)

One quarter of all CPAN distributions have been updated in the last 4 months!

Half of all updated in the last 17 months!

Top Modules

Many gems, including...

DBI DBD::* DBIx::Class Rose::DB::Object

Catalyst Moose DateTime

Algorithm::* Statistics::* Thread::*

XML::* HTML::* WWW::* Parse::*

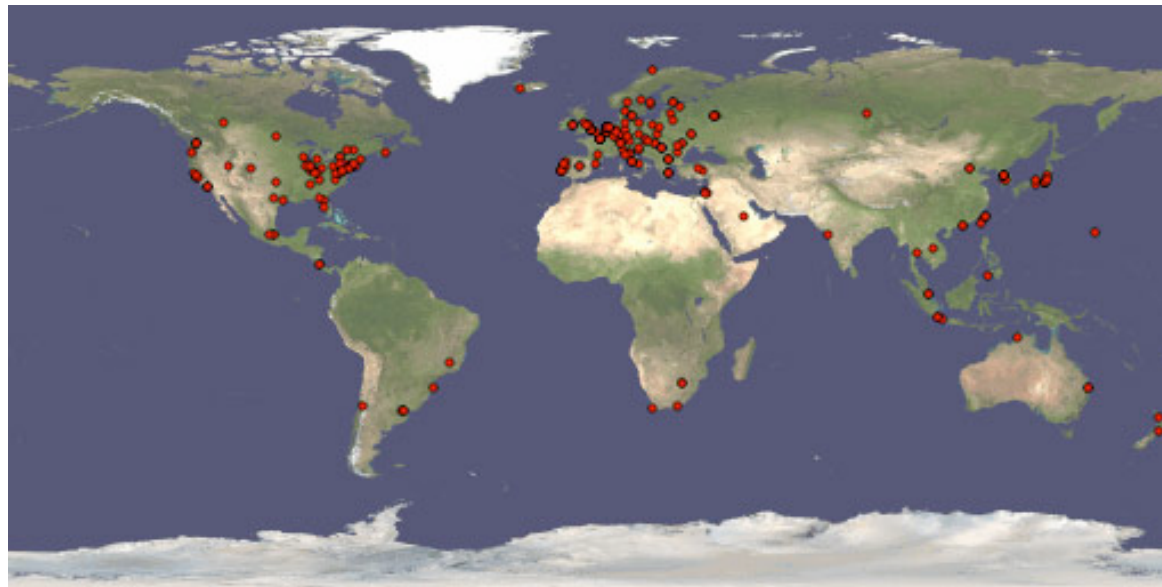
Net::* Email::* POE::* Locale::*

Test::* Devel::Cover Perl::Critic perltidy



Comprehensive Perl Archive Network

360 mirrors in 51 regions (TLDs)



Developer Services

Upload a perl module distribution
and you automatically get...

- *global distribution and archiving*
- *namespace ownership and management*
- *a bug tracking queue at rt.cpan.org*
- *a forum at cpanforum.com*
- *smoke testing on many platforms*

search.cpan.org

For each and every distribution:

Browse well formatted inter-linked docs

Links to forums, bug tracking, ratings,
annotated documentation, dependency
analysis, smoke test results, and more...

620,000 unique visitors per month

140,000 page views per day

41,000 visits per day



in

[Stevan Little](#) > [Moose-0.38](#)

[permalink](#)

Moose-0.38

- This Release**
- Other Releases**
- Links**
- CPAN Testers**
- Rating**
- License**
- Special Files**

Moose-0.38 [\[Download\]](#) [\[Browse\]](#) 15 Feb 2008

[\[Discussion Forum \]](#) [\[View/Report Bugs \(2\) \]](#) [\[Dependencies \]](#) [\[Other Tools \]](#)

PASS (62) NA (1) [\[View Reports \]](#) [\[Perl/Platform Version Matrix \]](#)

★★★★★ (3 Reviews) [\[Rate this distribution \]](#)

Perl ([Artistic](#) and [GPL](#))

[Build.PL](#) [MANIFEST](#) [Makefile.PL](#)
[Changes](#) [META.yml](#) [README](#)



Modules



















Moose	A postmodern object system for Perl 5	0.38
Moose::Meta::Attribute	The Moose attribute metaclass	0.31

Documentation

Moose::Cookbook	How to cook a Moose
Moose::Cookbook::FAQ	Frequently asked questions about Moose

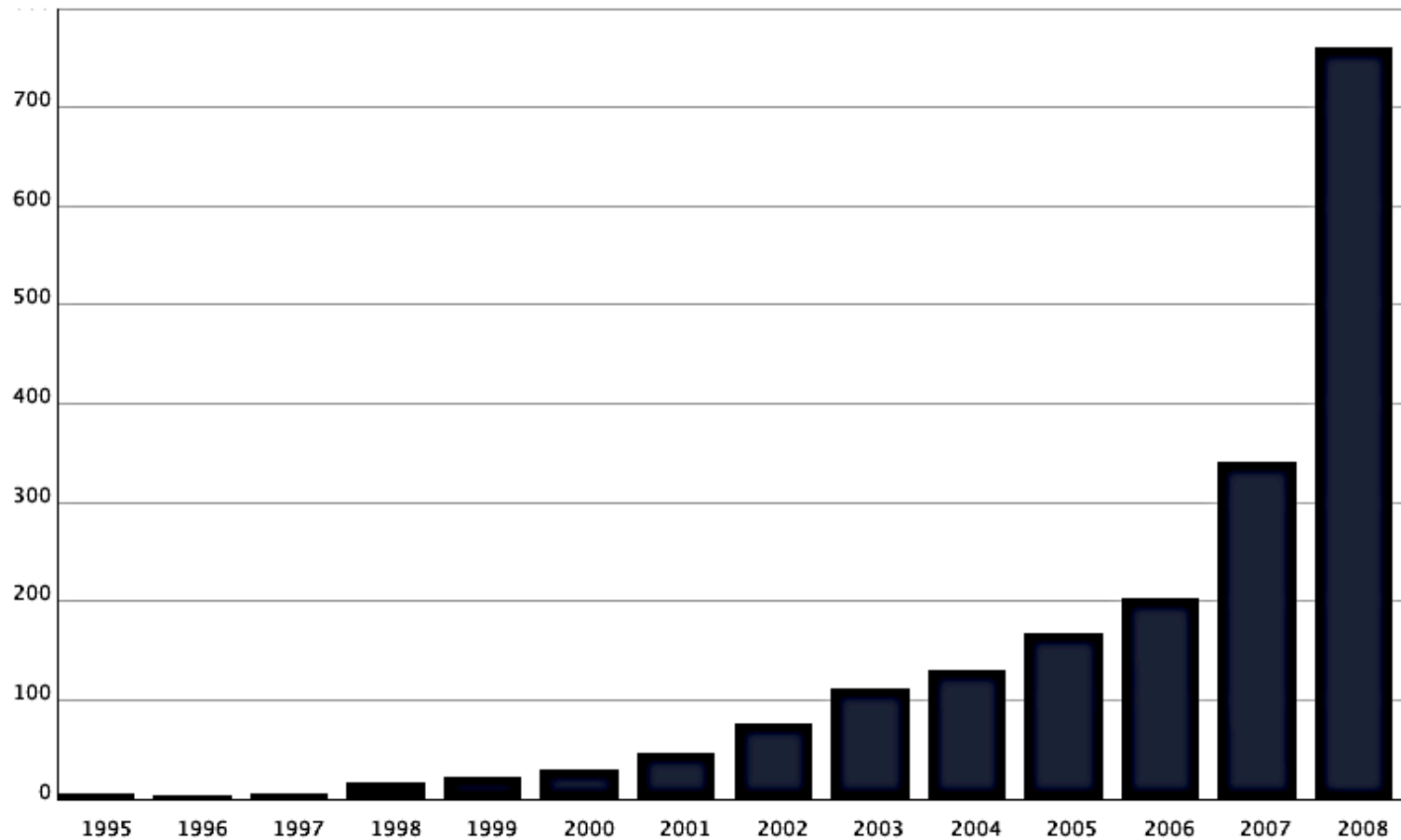
Dependency Analysis available for all Modules

<http://bbbike.radzeit.de/~slaven/cpantestersmatrix.cgi>

<u>Moose</u>		16 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Test::LongString</u>		108 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Test::Builder</u>		Core module
<u>Scalar::Util</u>		Core module
<u>Test::Exception</u>		28 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Test::Harness</u>		Core module
<u>Sub::Uplevel</u>		34 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Exporter</u>		Core module
<u>Carp</u>		Core module
<u>Filter::Simple</u>		Core module
<u>Class::MOP</u>		20 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>File::Spec</u>		Core module
<u>Sub::Name</u>		16 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Sub::Exporter</u>		81 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Params::Util</u>		99 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Sub::Install</u>		102 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
<u>Data::OptList</u>		89 PASSEs, 0 UNKNOWNs, 0 FAILs, 0 NAs
Chance of success		100%

Average Uploads/Month

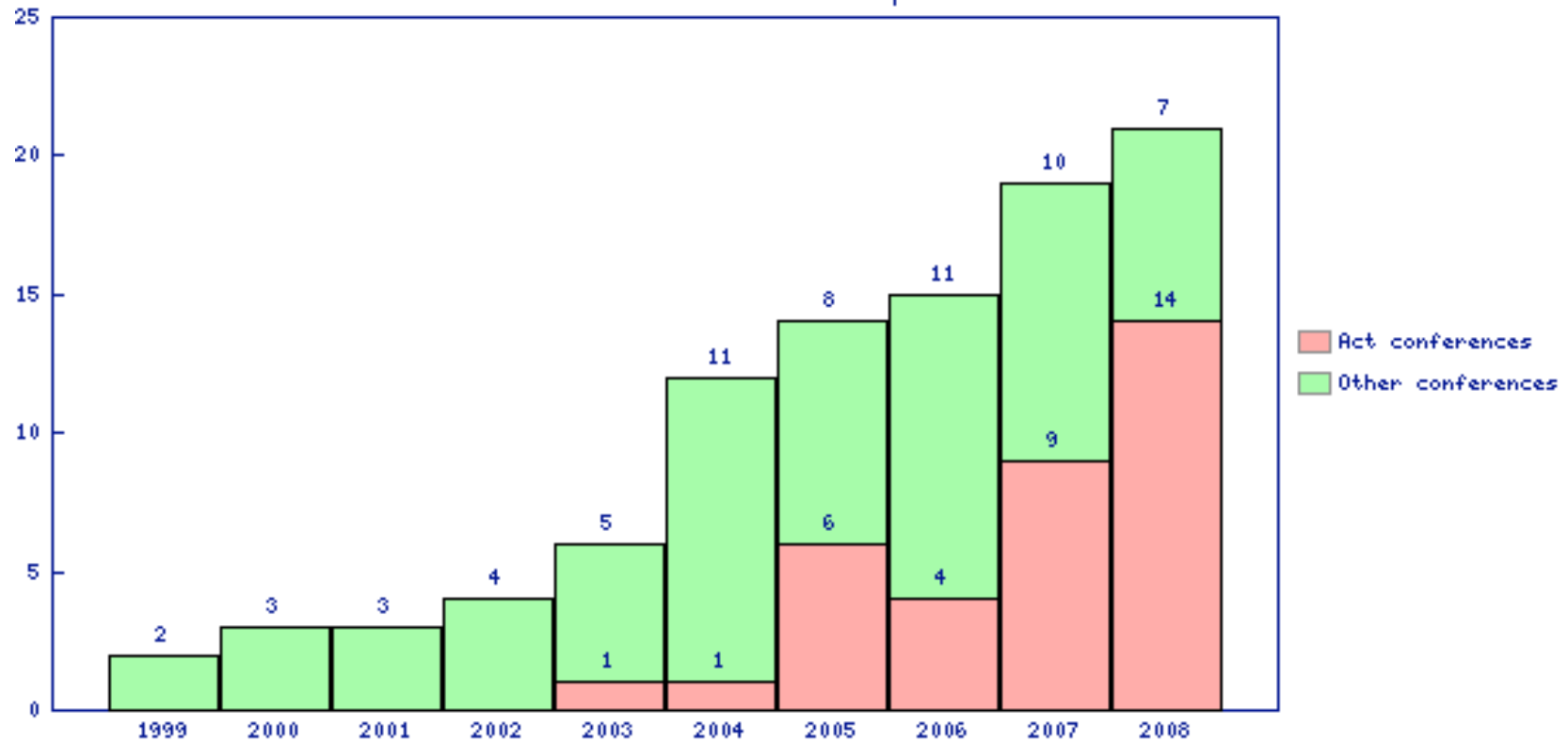
Created on: Wednesday January 30, 7:33 PM



Perl Mongers

Grass Roots Community & Conferences

Act-hosted Perl conferences/workshops/hackathons



~ Myths ~

BUSTED!
Perl is dead

Perl is hard to read / test / maintain

Perl 6 is killing Perl 5

~ Myths ~

BUSTED!
Perl is dead

Perl is hard to read / test / maintain

Perl 6 is killing Perl 5

“True greatness is measured by how much freedom you give to others, not by how much you can coerce others to do what you want.”

—Larry Wall

Guidelines and Tools

Perl Best Practices

Perl::Tidy

Perl::Critic

Test::*

Devel::Cover

Standards and Styles for Developing Maintainable Code



Perl
Best Practices

O'REILLY®

Damian Conway

Perl::Tidy

Perl code beautifier

Works beautifully - can be trusted

Supports *many* options for personal styles

Perl Best Practices recommended options

Perl::Critic

Static Code Analysis for Perl

Includes over 120 policies

Most based on Perl Best Practices

Grouped into levels and themes

Configurable and extensible
for local policies and styles

Test::*

Perl culture takes testing seriously

Excellent mature tools for testing

Test Anything Protocol - producers/consumers

http://en.wikipedia.org/wiki/Test_Anything_Protocol

Test::* modules make it easy to write tests

Over 200 Test::* distributions on CPAN

Perl Testing

*A Developer's
Notebook™*

Ian Langworth & chromatic

- Testing strategies
- Testing whole applications
- Testing Perl & non-Perl code
- Helpful testing libraries

O'REILLY®

Devel::Cover

Coverage Analysis for Perl

Tells you what code has been executed

Statement, branch, condition, subroutine, pod

Produces drill-down reports in HTML

Devel::Cover Reports

<http://pjcj.sytes.net/cover/latest/>

File	stmt	bran	cond	sub	pod	total
AcePerl-1.91	72.48	52.74	43.18	74.66	56.36	64.15
Algorithm-Annotate-0.10	100.00	50.00	n/a	100.00	0.00	87.80
Algorithm-C3-0.06	100.00	100.00	100.00	100.00	100.00	100.00
Algorithm-Dependency-1.102	75.13	41.60	31.43	85.00	92.11	67.14
Algorithm-Diff-1.1902	87.47	78.68	77.66	80.77	100.00	83.85

file	stmt	bran	cond	sub	pod	time	total
blib/lib/Algorithm/Dependency.pm	95.2	83.3	n/a	93.3	100.0	34.4	92.7
blib/lib/Algorithm/Dependency/Item.pm	100.0	50.0	33.3	100.0	100.0	8.2	86.8
blib/lib/Algorithm/Dependency/Ordered.pm	100.0	75.0	n/a	100.0	100.0	9.3	93.7
blib/lib/Algorithm/Dependency/Source.pm	86.3	55.0	46.7	90.9	100.0	22.2	75.5
blib/lib/Algorithm/Dependency/Source/File.pm	100.0	50.0	n/a	100.0	100.0	7.8	85.2
blib/lib/Algorithm/Dependency/Source/HoA.pm	100.0	50.0	n/a	100.0	100.0	0.5	90.5
blib/lib/Algorithm/Dependency/Weight.pm	97.7	44.4	n/a	100.0	100.0	13.6	85.9
inc/Class/Inspector.pm	41.1	19.6	0.0	43.5	83.3	2.1	35.6
inc/Test/ClassAPI.pm	77.2	40.5	28.6	100.0	50.0	1.9	68.4
Total	75.1	41.6	31.4	85.0	92.1	100.0	67.1

blib/lib/Algorithm/Dependency/Weight.pm			
<i>Criterion</i>	<i>Covered</i>	<i>Total</i>	<i>%</i>
statement	42	43	97.7
branch	8	18	44.4
condition			n/a
subroutine	12	12	100.0
pod	5	5	100.0
total	67	78	85.9

line	stmt	bran	cond	sub	pod	time	code
1							package Algorithm::Dependency::Weight ;

223							sub weight_all {
224	1			1	1	11	my \$self = shift;
225	1					13	my @items = \$self->source->items;
226	1	50				15	defined \$items[0] or return undef;
227	1					10	\$self->weight_hash(map { \$_->id } @items);
	6					67	

blib/lib/Algorithm/Dependency/Source.pm			
<i>Criterion</i>	<i>Covered</i>	<i>Total</i>	<i>%</i>
condition	7	15	<u>46.7</u>

and 3 conditions

line	!!	!&&!r	!&&r	condition
139	0	0	1037	defined \$_[0] && !ref(\$_[0])
	0	0	1037	defined \$_[0] && !ref(\$_[0]) && \$_[0] ne ''

or 3 conditions

line	!	!!&&r	!!&&!r	condition
140	1037	0	0	\$\$self{'loaded'} or \$self->load
161	11	3	0	\$\$self{'loaded'} or \$self->load
182	4	1	0	\$\$self{'loaded'} or \$self->load

~ Myths ~

BUSTED!
Perl is dead

Perl is hard to read / test / maintain

BUSTED!

Perl 6 is killing Perl 5

~ Myths ~

BUSTED!
Perl is dead

BUSTED!
Perl is hard to read / test / maintain

Perl 6 is killing Perl 5

Perl 6 *saved* Perl 5!

“Perl 5 had already started dying, because people were starting to see it as a dead-end language.

It seemed odd at the time, but when we announced Perl 6, Perl 5 suddenly took on a new life.”

—Larry Wall, 2002

Perl 6 *saved* Perl 5!

In 2000 perl was dying from the inside out

The Perl 6 RFC process “vented speen”

Perl 5 development has gone smoothly since

Much refactoring driven by Perl5-on-Parrot

Many new features inspired by Perl 6 work

Perl 5.10

Perl 5.10 was released in December 2007
Five years after 5.8.0, two years after 5.8.8

Refactored internals

many fixes, more speed, less memory

Switch statement, smart matching, named captures, state variables, defined-or, say, field hashes, pluggable regex engines, trie-based non-recursive pattern matching, and more...

<http://www.slideshare.net/rjbs/perl-510-for-people-who-arent-totally-insane>

A Culture of Testing

Another bonus from Perl 6:

Strong test suites for Perl 5 code are needed to ensure backwards compatibility

Strengthened culture of testing

Perl Test Suite

2002: Perl 5.8.0 had **26,725** core tests
+41,666 more for bundled libraries etc.

2007: Perl 5.10.0 has **78,883** core tests
+109,427 more for bundled libraries etc.

Module Test Suites

CPAN Testers Network:

Automated smoke testing of CPAN uploads

Runs the test suite included in distribution

over 60 different platforms

over 20 different perl versions

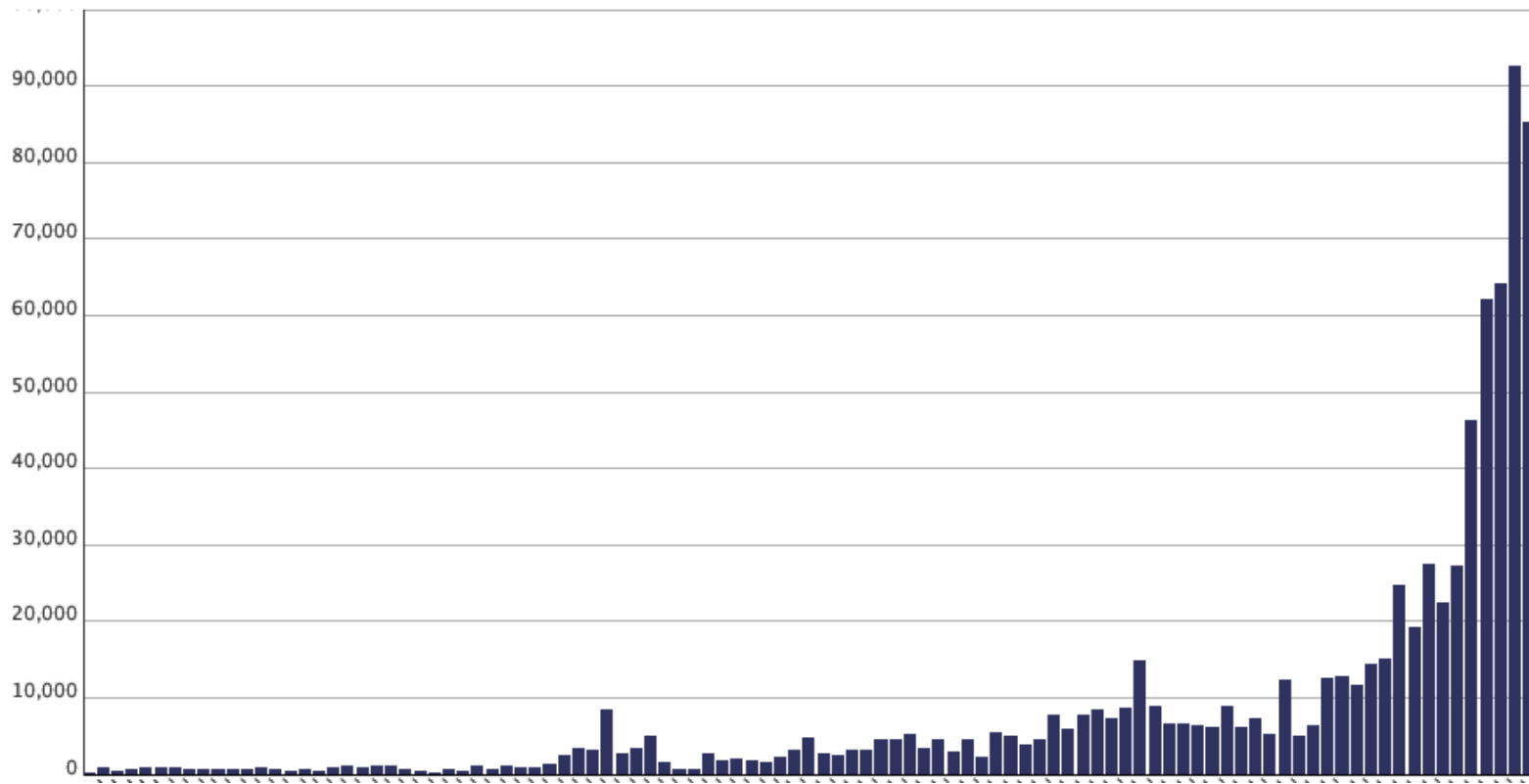
Immediate feedback for developers

Platform/Version analysis available for all Modules

<http://cpandeps.cantrell.org>

	MSWin32	VMS	cygwin	darwin	freebsd	linux	netbsd	openbsd	solaris
5.11.0			█			█	█		
5.10.0	█		█	█	█	█	█	█	█
5.9.5				█	█	█	█	█	█
5.9.1						█			
5.9.0						█			
5.8.8	█		█	█	█	█	█	█	█
5.8.7		█		█		█	█		█
5.8.6				█	█	█	█		
5.8.5				█	█	█	█		█
5.8.4				█	█	█	█		█
5.8.3				█		█	█		
5.8.2				█			█		
5.8.1				█			█		
5.8.0						█			
5.6.2				█	█	█	█		
5.6.1						█			
5.5.5					█	█			

~90,000 Reports/Month



Monthly - August 1999 to December 2007

~ Myths ~

BUSTED!
Perl is dead

BUSTED!
Perl is hard to read / test / maintain

BUSTED!
Perl 6 is killing Perl 5

~ Myths ~

BUSTED!
Perl is dead

BUSTED!
Perl is hard to read / test / maintain

BUSTED!
Perl 6 is killing Perl 5

Perl 6 Myths

“Perl 6 is not Perl”

“It'll never be finished”

“There's no code written in Perl 6”

“Perl 6 is not Perl”

“Perl 6 [...] is clearer, more direct, more expressive, and without many of the old false leads and rough edges in Perl 5.

So, Perl 6 is still Perl in that a programmer looking at a Perl 6 program will instantly recognize that it is "Perl", even if some of the details are different.”

—Larry Wall

“It’ll never be finished”

“If we'd done Perl 6 on a schedule, you'd have it by now. And it would be crap.”

—Larry Wall

"do it right" and "it's ready when it's ready"

“Truly radical and far-reaching improvements over the past few years.”

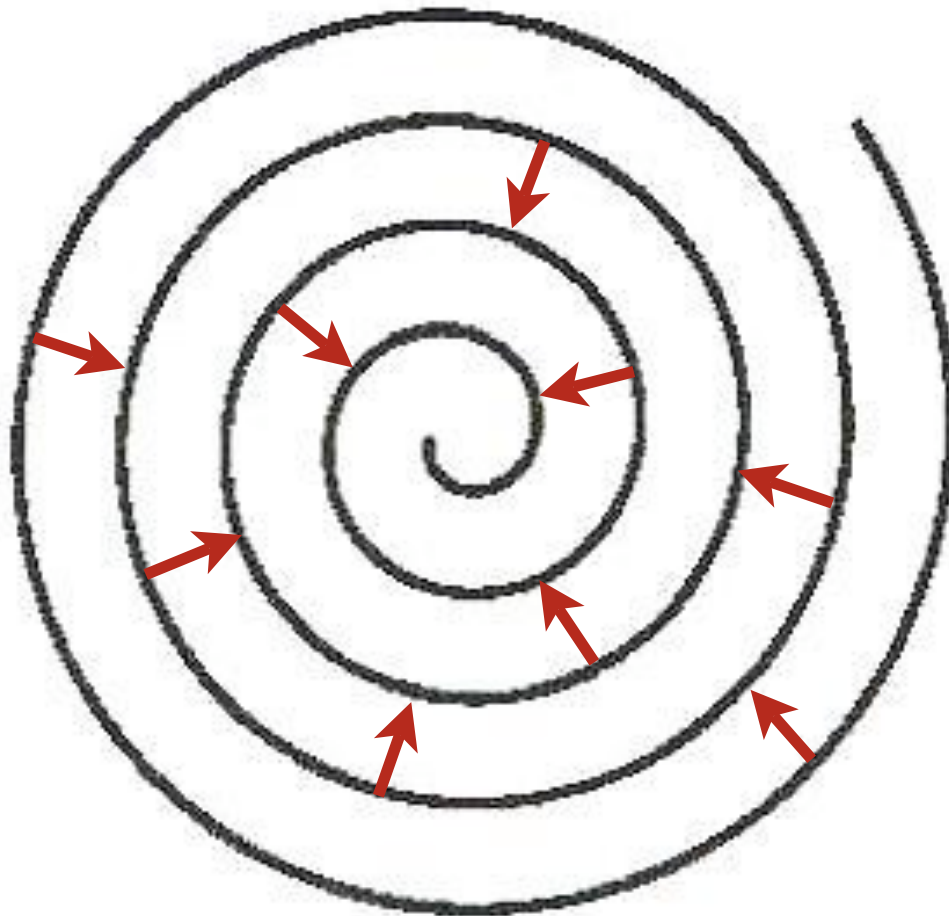
When will be be done?

*“We're not doing the Waterfall [model of development] we're doing the **Whirlpool**, where the strange attractor whirls around **with feedback at many levels** but eventually converges on something in the middle.”*

—Larry Wall

in 'What criteria mark the closure of perl6 specification'

Specification is now relatively stable



“feedback at many levels”

Multiple implementations

Perl 6 compilers

Pugs - initial experimentation in Haskell

KindaPerl6 - perl5 - aiming to self-host

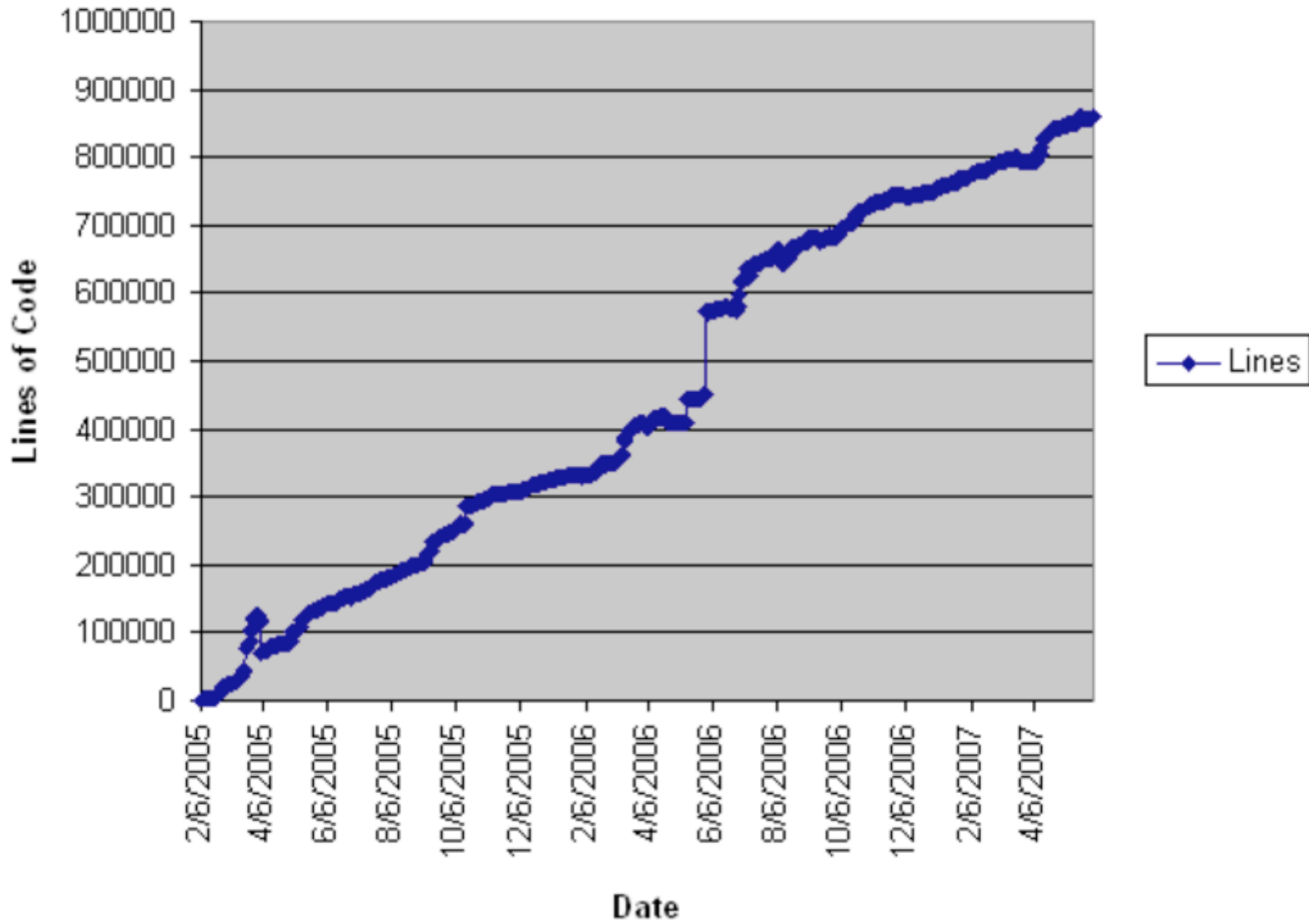
SMOP - Simple Meta Object Protocol

Rakudo - built on Parrot Compiler Toolchain

Sharing a common test suite



Pugs - 850,000 Lines of Code in 2 Years



“There's no Perl 6 code”

- > 35,000 lines of tests written in Perl 6
- > 10,000 lines of examples written in Perl 6

parrot





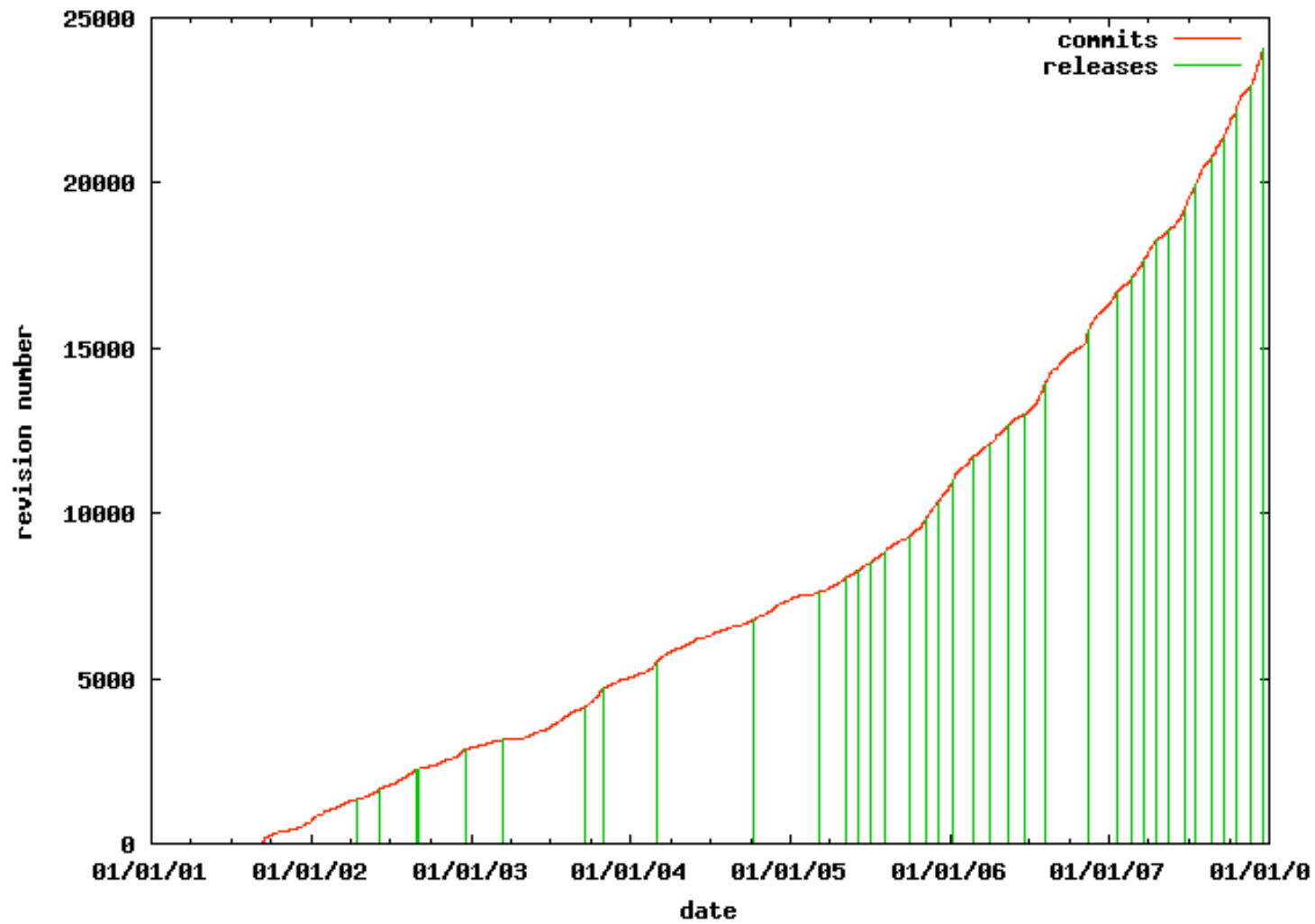
An advanced virtual machine for dynamic languages

Advanced capabilities with JIT compiling

Already supports over 50 languages
about 20 of which are non-toy

Python, Ruby, PHP, Lua, Lisp, TCL, ...

Parrot development at r24049



Parrot Test Suite

11,086 tests in 557 files

53,884 tests if run for all 7 runcores

(in Jan 2008, many more now)

Parrot Compiler Toolkit

Write a compiler in an afternoon!

“Parrot is really quite wonderful. [...] Parrot lets you implement your own languages using Perl 6 rules for the grammar and Perl 6 for the compiler.”

- Simon Cozens

<http://blog.simon-cozens.org/post/view/1323>

Rakudo

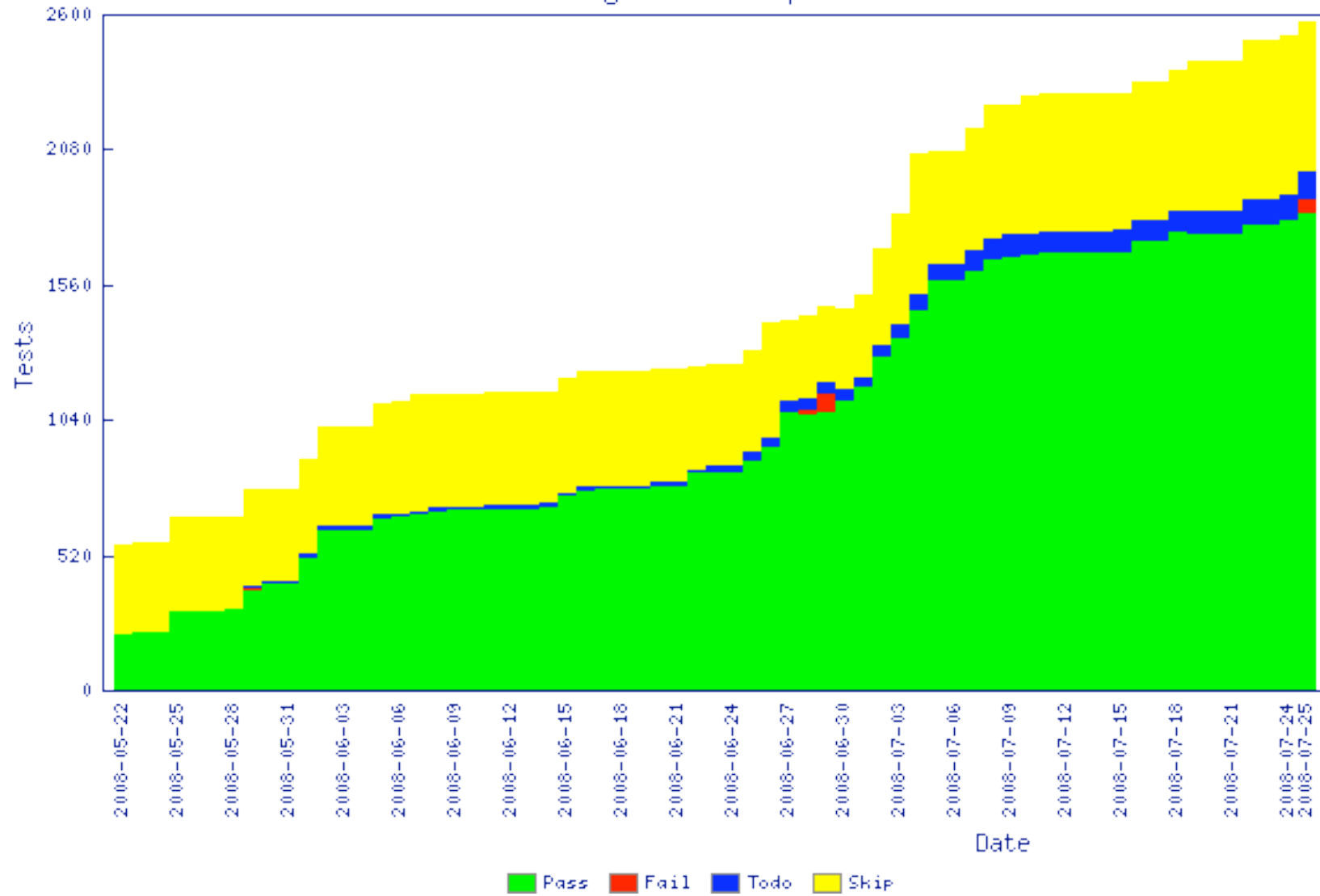
Perl 6 on Parrot

"Rakudo" short for "rakuda-do"

Japanese for "Way of the Camel"

"rakudo" also means "paradise"

Passing Rakudo Spectests



Community Resources

*“Awesome community.
Perl people tend to be laid-back and friendly.”*

perlmonks.org - Meditations and wisdom

use.perl.org - News and blogs

perlbuzz.com - Headlines and articles

comp.lang.perl.* - Newsgroups

Any questions?

<http://perlmonks.org>
<http://search.cpan.org>
<http://blog.timbunce.org>