

Devel::NYTProf

A Perl Performance Experience
By
Andrew Maguire

Usage

<http://search.cpan.org/.../Devel/NYTPProf.pm>

- `perl -d:NYTPProf ...`
- `nytprofhtml`
- Open `index.html`

- `export NYTPROF=file=$dir/nytprof.out`
- `nytprofhtml -f $dir/nytprof.out -o $dir --no-flame`

Performance Environment

- Stable environment
- Create and run baseline for comparison
LOOP
- Analyse results
- Change code
- Run profiler
END LOOP

Demo of interface

- <http://timbunce.github.io/.../sample-report/index.html>
- nytpperf/index.html

\$&

- \$& instead of () placeholders

```
$name =~ s/\d+(st|nd|th)\b/$&/oix;
```

```
$name =~ s/(\d\d?(?:st|nd|th))\b/$1/oix;
```

- Perl 5.20 reimplemented algorithm to avoid performance penalty.

Performance

Readability

Maintainability

- my \$self vs. \$_[0] *and* shift

```
sub function1 {  
    my $self = shift;  
    return $self->method(@_);  
}
```

```
sub function2 {  
    return shift->method(@_);  
}
```

- Intermediate variables

```
my $person = $people->{$person_id};  
my $address = $person->{$address_id};
```

```
$people->{$person_id}->{$address_id};
```

4688387	77.7s	4688387	55.5s	<pre> *\$field = sub { shift->_field_accessor(\$field, @_); # spent 55.5s making 4688387 calls to EEC::DataMatching::Record::_field_accessor, avg 12µs/call } </pre>
4346219	6.44s			<pre> *\$field = sub { my \$self = shift; </pre>
4346219	70.6s	4346219	129s	<pre> return \$self->_field_accessor(\$field, @_); # spent 129s making 4346219 calls to EEC::DataMatching::Record::_field_accessor, avg 30µs/call } </pre>

				sub new
				# spent 31.8s (14.1+17.7) within EEC::DataMatching::Record::new which was called 213492 times, avg 149µs/call # 121049 times (7.37s+9.39s) by EEC::DataMatching::Loader::EECFilter::load at line 59 of ../../backend/perl/EEC/DataMatching/Loader/EECFilter.pm, avg 138µs/call # 87374 times (6.32s+7.86s) by EEC::DataMatching::Loader::EECFilter::load at line 97 of ../../backend/perl/EEC/DataMatching/Loader/EECFilter.pm, avg 162µs/call # 5069 times (438ms+473ms) by EEC::DataMatching::Loader::load at line 323 of ../../backend/perl/EEC/DataMatching/Loader.pm, avg 180µs/call
213492	386ms			{ my \$class = shift;
213492	504ms			my \$db = shift or croak "DataMatching::DB object not supplied";
213492	3.22s	213492	1.14s	croak "Invalid DataMatching::DB object (\$db)" unless \$db->isa('EEC::DataMatching::DB'); # spent 1.14s making 213492 calls to UNIVERSAL::isa, avg 5µs/call
213492	400ms			my \$data = shift or croak "Data not supplied";
213492	369ms			my \$persons = shift;
213492	824ms			my \$self = {DB => \$db, DATA => \$data, };
213492	1.37s			bless \$self, \$class;
213492	1.62s	213492	8.82s	my \$source = \$self->source or croak "Source not supplied"; # spent 8.82s making 213492 calls to EEC::DataMatching::Record::_ANON_ [../../backend/perl/EEC/DataMatching/Record.pm:74], avg 41µs/call
213492	1.39s	213492	7.76s	my \$row = \$self->row or croak "Row not supplied"; # spent 7.76s making 213492 calls to EEC::DataMatching::Record::_ANON_ [../../backend/perl/EEC/DataMatching/Record.pm:74], avg 36µs/call
213492	624ms			\$self->{DATA}->{persons} = \$persons;
213492	1.85s			return \$self;
				}

				sub new
				# spent 16.6s (13.8+2.84) within EEC::DataMatching::Record::new which was called 213488 times, avg 78µs/call # 121045 times (7.89s+1.58s) by EEC::DataMatching::Loader::EECFilter::load at line 59 of ../../backend/perl/EEC/DataMatching/Loader/EECFilter.pm, avg 78µs/call # 87374 times (5.53s+1.19s) by EEC::DataMatching::Loader::EECFilter::load at line 97 of ../../backend/perl/EEC/DataMatching/Loader/EECFilter.pm, avg 77µs/call # 5069 times (348ms+70.7ms) by EEC::DataMatching::Loader::load at line 323 of ../../backend/perl/EEC/DataMatching/Loader.pm, avg 83µs/call
213488	441ms			{ my \$class = shift;
213488	476ms			my \$db = shift or croak "DataMatching::DB object not supplied";
213488	3.72s	213488	1.20s	croak "Invalid DataMatching::DB object (\$db)" unless \$db->isa('EEC::DataMatching::DB'); # spent 1.20s making 213488 calls to UNIVERSAL::isa, avg 6µs/call
213488	424ms			my \$data = shift or croak "Data not supplied";
213488	415ms			my \$persons = shift;
213488	621ms			ref(\$data) eq 'HASH' or croak "Data is not a hash";
213488	540ms			exists(\$data->{source}) or croak "Source not supplied";
213488	549ms			exists(\$data->{row}) or croak "Row not supplied";
213488	1.39s			my \$self = { %\$data };
213488	604ms			\$self->{DB} = \$db;
213488	1.80s	213488	1.64s	\$self->{LOGGER} = \$db->logger; # spent 1.64s making 213488 calls to EEC::DataMatching::DB::logger, avg 8µs/call
213488	499ms			\$self->{PERSONS} = \$persons;
213488	1.42s			bless \$self, \$class;
213488	1.91s			return \$self;
				}

				<pre># spent 7.23s within EEC::DataMatching::Record::db which was called 1320836 times, avg 5µs/call: # 1104761 times (6.03s+0s) by EEC::DataMatching::Record::logger at line 124, avg 5µs/call # 208423 times (1.16s+0s) by EEC::DataMatching::FilterRecord::insert at line 12 of ../../backend/perl/EEC/DataMat # 5069 times (26.8ms+0s) by EEC::DataMatching::Record::insert at line 928, avg 5µs/call # 2583 times (14.1ms+0s) by EEC::DataMatching::Record::check_town_suburb_city_combo at line 762, avg 5µs/call sub db { my \$self = shift; return \$self->{DB}; }</pre>
1320836	1.96s			
1320836	10.6s			
				<pre># spent 35.2s (22.9+12.4) within EEC::DataMatching::Record::logger which was called 1104761 times, avg 32µs/call: # 667840 times (14.1s+7.71s) by EEC::DataMatching::Record::field_accessor at line 156, avg 33µs/call # 426734 times (8.59s+4.53s) by EEC::DataMatching::Record::names at line 309, avg 31µs/call # 10132 times (209ms+109ms) by EEC::DataMatching::Record::a_address at line 277, avg 31µs/call # 29 times (472µs+311µs) by EEC::DataMatching::Record::log_error at line 138, avg 27µs/call # 26 times (414µs+296µs) by EEC::DataMatching::Record::check_town_suburb_city_combo at line 814, avg 27µs/call sub logger { my \$self = shift; return \$self->db->logger; # spent 6.32s making 1104761 calls to EEC::DataMatching::DB::logger, avg 6µs/call # spent 6.03s making 1104761 calls to EEC::DataMatching::Record::db, avg 5µs/call }</pre>
1104761	1.66s			
1104761	17.1s	2209522	12.4s	

				<pre># spent 1.26s within EEC::DataMatching::Record::db which was called 216072 times, avg 6µs/call: # 208419 times (1.22s+0s) by EEC::DataMatching::FilterRecord::insert at line 12 of ../../backend/pe # 5069 times (25.4ms+0s) by EEC::DataMatching::Record::insert at line 948, avg 5µs/call # 2584 times (15.6ms+0s) by EEC::DataMatching::Record::check_town_suburb_city_combo at line 784, ; sub db { return \$_[0]->{DB}; }</pre>
216072	2.24s			

				<pre># spent 5.02s within EEC::DataMatching::Record::logger which was called 1110690 times, avg 5µs/call # 667656 times (3.07s+0s) by EEC::DataMatching::Record::field_accessor at line 152, avg 5µs/call # 207865 times (971ms+0s) by EEC::DataMatching::Record::names at line 360, avg 5µs/call # 207865 times (861ms+0s) by EEC::DataMatching::Record::names at line 330, avg 4µs/call # 10132 times (43.9ms+0s) by EEC::DataMatching::Record::a_address at line 268, avg 4µs/call # 9163 times (37.8ms+0s) by EEC::DataMatching::Record::reject at line 560, avg 4µs/call # 4853 times (22.8ms+0s) by EEC::DataMatching::Record::validate_address at line 653, avg 5µs/call # 2864 times (12.7ms+0s) by EEC::DataMatching::Record::reject_if_not_nz at line 863, avg 4µs/call # 244 times (1.51ms+0s) by EEC::DataMatching::Record::field_accessor at line 149, avg 6µs/call # 26 times (104µs+0s) by EEC::DataMatching::Record::check_town_suburb_city_combo at line 836, a # 22 times (96µs+0s) by EEC::DataMatching::Record::log_error at line 131, avg 4µs/call sub logger { return \$_[0]->{LOGGER}; }</pre>
1110690	9.52s			

				<pre># spent 161s (147+14.4) within EEC::DataMatching::Cache::eec_address which was called 1942309 times, avg 83µs/call # 1940474 times (147s+14.4s) by EEC::DataMatching::Cache::eec_address_street_cache at line 185, avg 83µs/call # 877 times (92.3ms+9.50ms) by EEC::DataMatching::DB::rate_addresses at line 1119 of ../../backend/perl/EEC/Data # 870 times (92.0ms+9.54ms) by EEC::DataMatching::DB::match_address at line 814 of ../../backend/perl/EEC/Data # 77 times (5.74ms+581µs) by EEC::DataMatching::DB::rate_addresses at line 1162 of ../../backend/perl/EEC/Dat # 11 times (993µs+105µs) by EEC::DataMatching::DB::dump_letters at line 2388 of ../../backend/perl/EEC/DataMa sub eec_address {</pre>
1942309	3.90s			<pre>my(\$self, \$address_id) = @;</pre>
1942309	12.2s	1942309	14.4s	<pre>my \$record = \$self->_eec_address_id_cache->{\$address_id}; # spent 14.4s making 1942309 calls to EEC::DataMatching::Cache::_eec_address_id_cache, avg 7µs/call</pre>
1942309	26.0s			<pre>my @array = map { \$_ eq '' ? undef : \$_ } split(/~/, \$record);</pre>
1942309	3.90s			<pre>my \$len = scalar(@address_cache_fields)-1;</pre>
1942309	3.36s			<pre>my \$addr = {};</pre>
1942309	7.00s			<pre>for(0..\$len) {</pre>
19423090	72.5s			<pre> \$addr->{@address_cache_fields[\$_]} = \$array[\$_]; }</pre>
1942309	16.9s			<pre>return \$addr; }</pre>

				<pre># spent 85.4s (71.0+14.4) within EEC::DataMatching::Cache::eec_address which was called 1942317 times, avg 44µs/call # 1940474 times (70.8s+14.4s) by EEC::DataMatching::Cache::eec_address_street_cache at line 181, avg 44µs/call # 882 times (55.3ms+9.42ms) by EEC::DataMatching::DB::rate_addresses at line 1130 of ../../backend/perl/EEC/Data # 871 times (56.1ms+10.3ms) by EEC::DataMatching::DB::match_address at line 819 of ../../backend/perl/EEC/DataMa # 78 times (2.76ms+581µs) by EEC::DataMatching::DB::rate_addresses at line 1173 of ../../backend/perl/EEC/DataM # 12 times (585µs+114µs) by EEC::DataMatching::DB::dump_letters at line 2403 of ../../backend/perl/EEC/DataMatcl sub eec_address {</pre>
1942317	3.83s			<pre>my(\$self, \$address_id) = @;</pre>
1942317	12.1s	1942317	14.4s	<pre>my \$record = \$self->_eec_address_id_cache->{\$address_id}; # spent 14.4s making 1942317 calls to EEC::DataMatching::Cache::_eec_address_id_cache, avg 7µs/call</pre>
1942317	2.75s			<pre>my %addr;</pre>
1942317	35.4s			<pre>@addr{@address_cache_fields} = map { \$_ eq '' ? undef : \$_ } split(/~/, \$record);</pre>
1942317	15.9s			<pre>return \%addr; }</pre>

Simplified eec_address()

- Cache was being created like this:

```
my $cache = $self->{EEC_ADDRESS_ID_CACHE} ||= {};  
...  
$cache->{$eec_address_id} =  
    join("~", map {  
        my $v = $address_rec->{$_};  
        defined($v) ? $v : "  
    } @address_cache_fields);
```

- Now becomes

```
$cache->{$eec_address_id} = $address_rec;
```

- With the cache read function now simply:

```
sub eec_address {  
    return $_[0]->{EEC_ADDRESS_ID_CACHE}->{$_[1]};  
}
```