Profiling PHP applications

Ike Devolder / @BlackIkeEagle / @studioemma

Who likes slow applications?
Photo by rihaij on Pixabay

Only faster execution time?

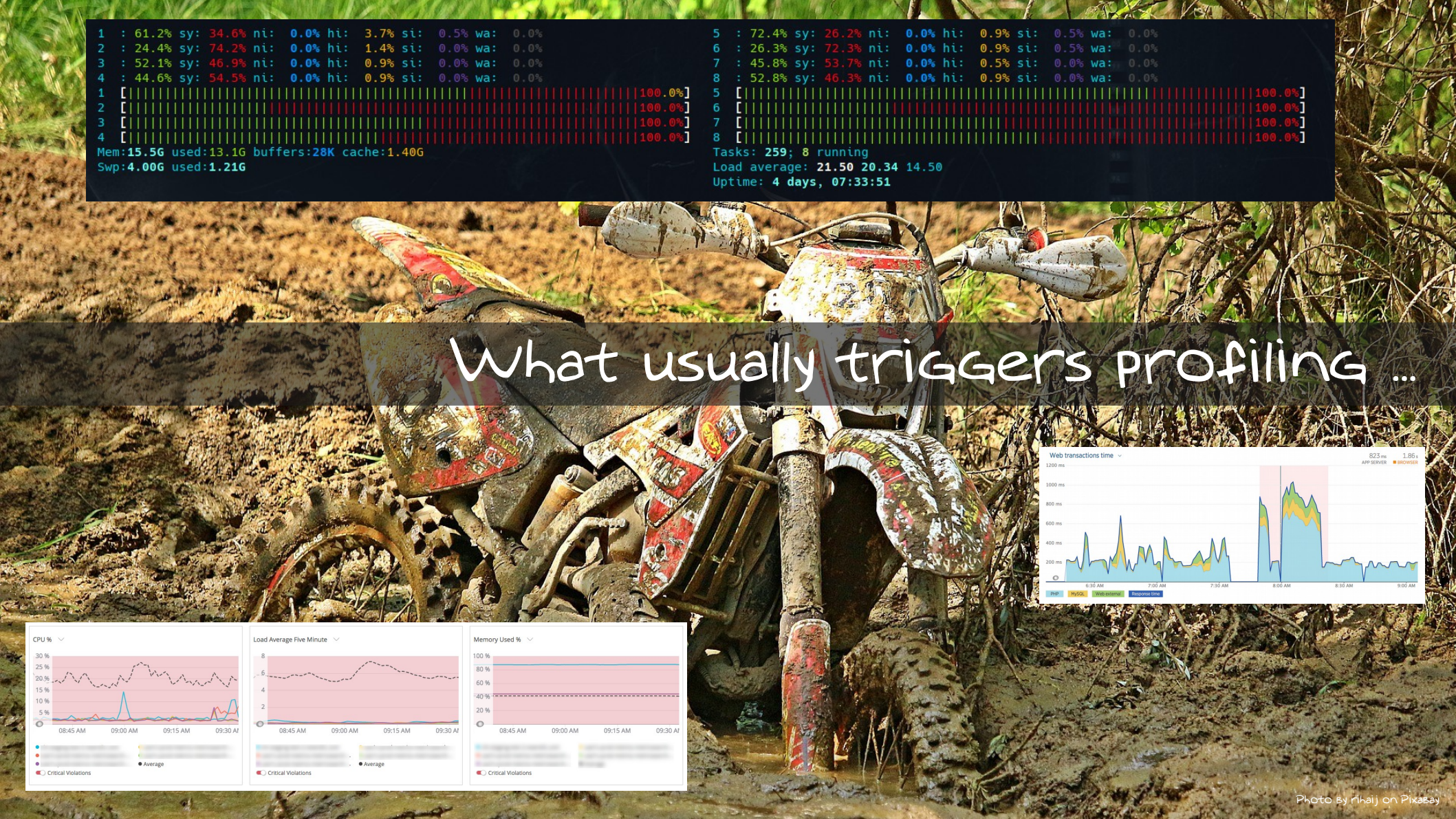
Photo by Narcis Ciocan on Pixabay

Reduce resource usage?

Photo by sharon1090 on Pixabay

More RPM?

Photo by skeeze on Pixabay


1 : 61.2% sy: 34.6% ni: 0.0% hi: 3.7% si: 0.5% wa: 0.0%
2 : 24.4% sy: 74.2% ni: 0.0% hi: 1.4% si: 0.0% wa: 0.0%
3 : 52.1% sy: 46.9% ni: 0.0% hi: 0.9% si: 0.0% wa: 0.0%
4 : 44.6% sy: 54.5% ni: 0.0% hi: 0.9% si: 0.0% wa: 0.0%
5 : 72.4% sy: 26.2% ni: 0.0% hi: 0.9% si: 0.5% wa: 0.0%
6 : 26.3% sy: 72.3% ni: 0.0% hi: 0.9% si: 0.5% wa: 0.0%
7 : 45.8% sy: 53.7% ni: 0.0% hi: 0.5% si: 0.0% wa: 0.0%
8 : 52.8% sy: 46.3% ni: 0.0% hi: 0.9% si: 0.0% wa: 0.0%
Mem:15.5G used:13.1G buffers:28K cache:1.40G
Swp:4.00G used:1.21G
Tasks: 259; 8 running
Load average: 21.50 20.34 14.50
Uptime: 4 days, 07:33:51
What usually triggers profiling ...
Web transactions time
CPU %
Load Average Five Minute
Memory Used %
Critical Violations
Average
Photo by rihaij on Pixabay

Hi,

I'm Ike,
@BlackIkeEagle on Twitter

Developer / Operations
at Studio Emma

studio emma

internet expertise

Magento®

Professional solution partner

PIMCORE®

Gold solution partner

Profiling

is the process of application runtime characteristics measurement for performance analysis and optimization

Photo by Ronald Plett on Pixabay

Knowledge Gained?

Knowledge Gained?

– where

Knowledge Gained?

- where
- what

Knowledge Gained?

- where
- what
- how much

Why?

My application is 'fast'

Why?

My application is 'fast'

– define fast

Why?

My application is 'fast'

- define fast
- can it handle a Burst?

WHY?

My application is 'fast'

- define fast
- can it handle a Burst?
- optimizing has Benifits

No profiler needed, I know where the bottleneck is.

No profiler needed, I know where the bottleneck is.

Nope, that's shooting in the dark

Points of interest

Points of interest

Execution time

Points of interest

Execution time

CPU time

Points of interest

Execution time

CPU time

Memory usage

Points of interest

Execution time

CPU time

Memory usage

Disk IO

Points of interest

Execution time

CPU time

Memory usage

Disk IO

Network time

Points of interest

- Execution time
- CPU time
- Memory usage
- Disk IO
- Network time
- External delays

How can we profile

How can we profile

PHP?

How can we profile

PHP?

Thats not really profiling

```php
<?php

$start = microtime(true);
// ...
$this->code();
// ...
$time = microtime(true) - $start;
```

How can we profile: XdeBug

XdeBug

How can we profile: Xdebug

Xdebug php extension

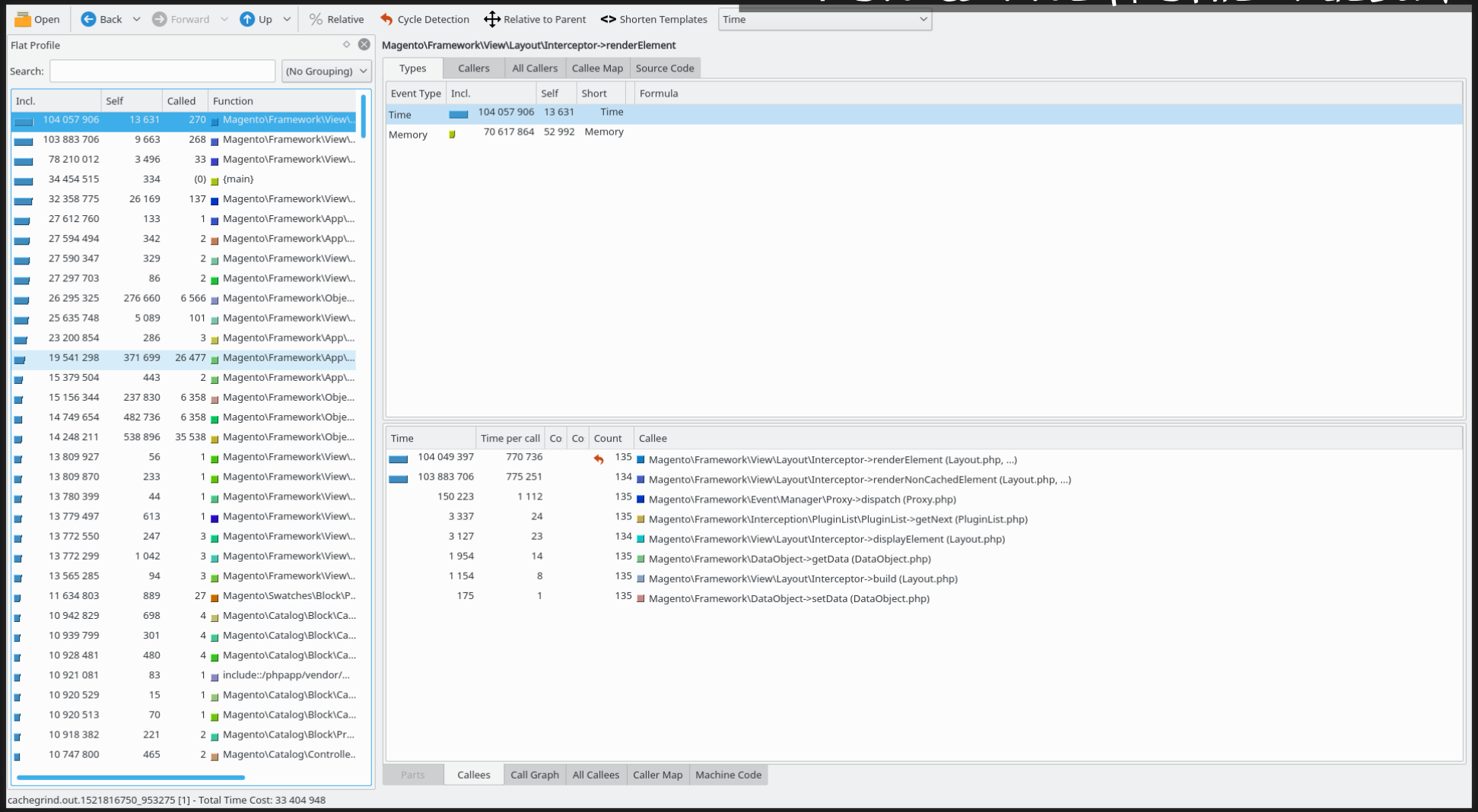
Some configuration needed

External tool to analyze

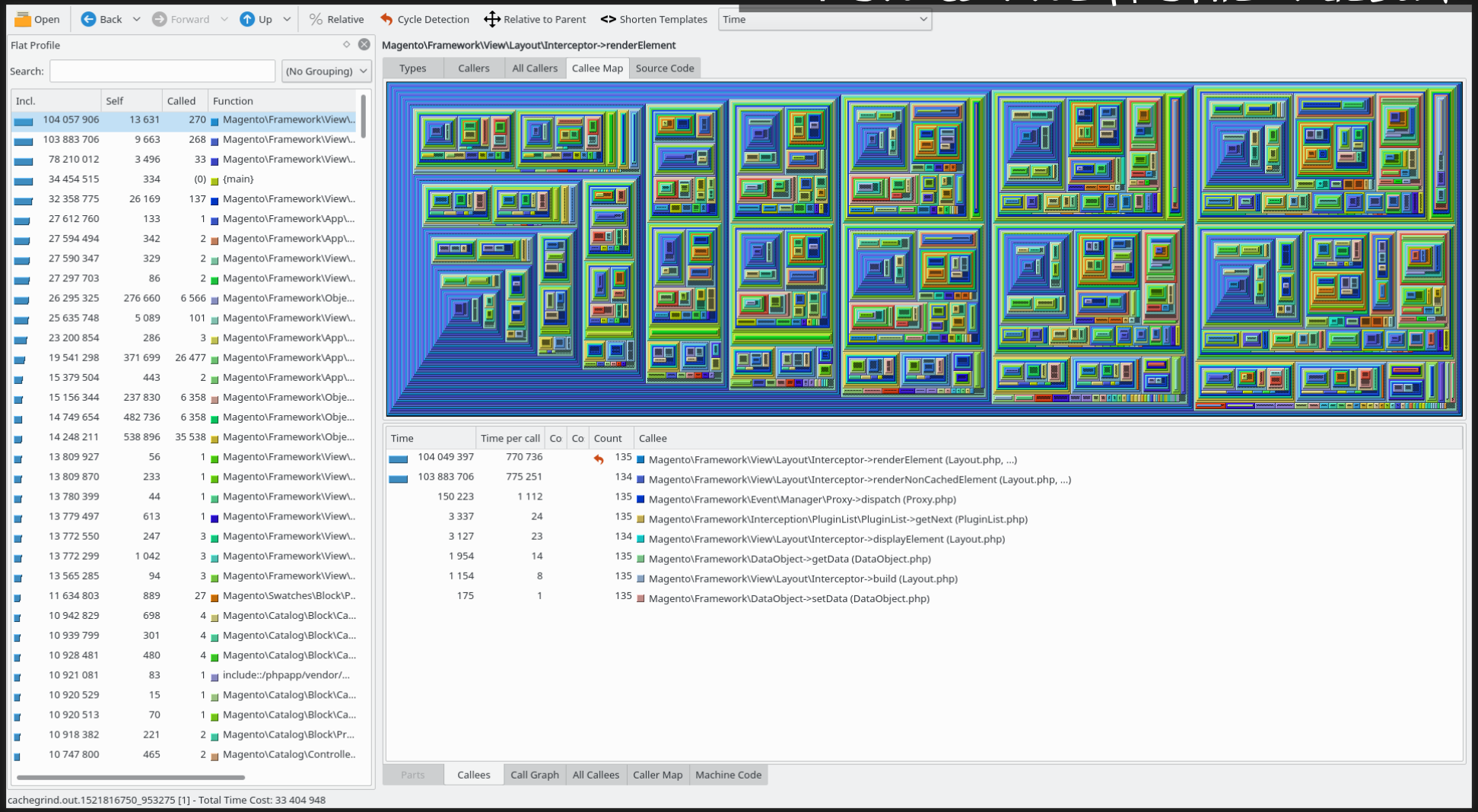
How can we profile: Xdebug

```
xdebug.profiler_enable=1
xdebug.profiler_output_dir=/xdebug
xdebug.profiler_output_name=cachegrind.out.%u
```

How can we profile: XDebug



How can we profile: XDebug



It's Hard

How can we profile: XdeBug

Photo by Aaron Sunday on Pixabay

How can we profile: Xhprof

Xhprof

How can we profile: Xhprof

Tideways php extension

Configuration needed

Xhgui to view and compare

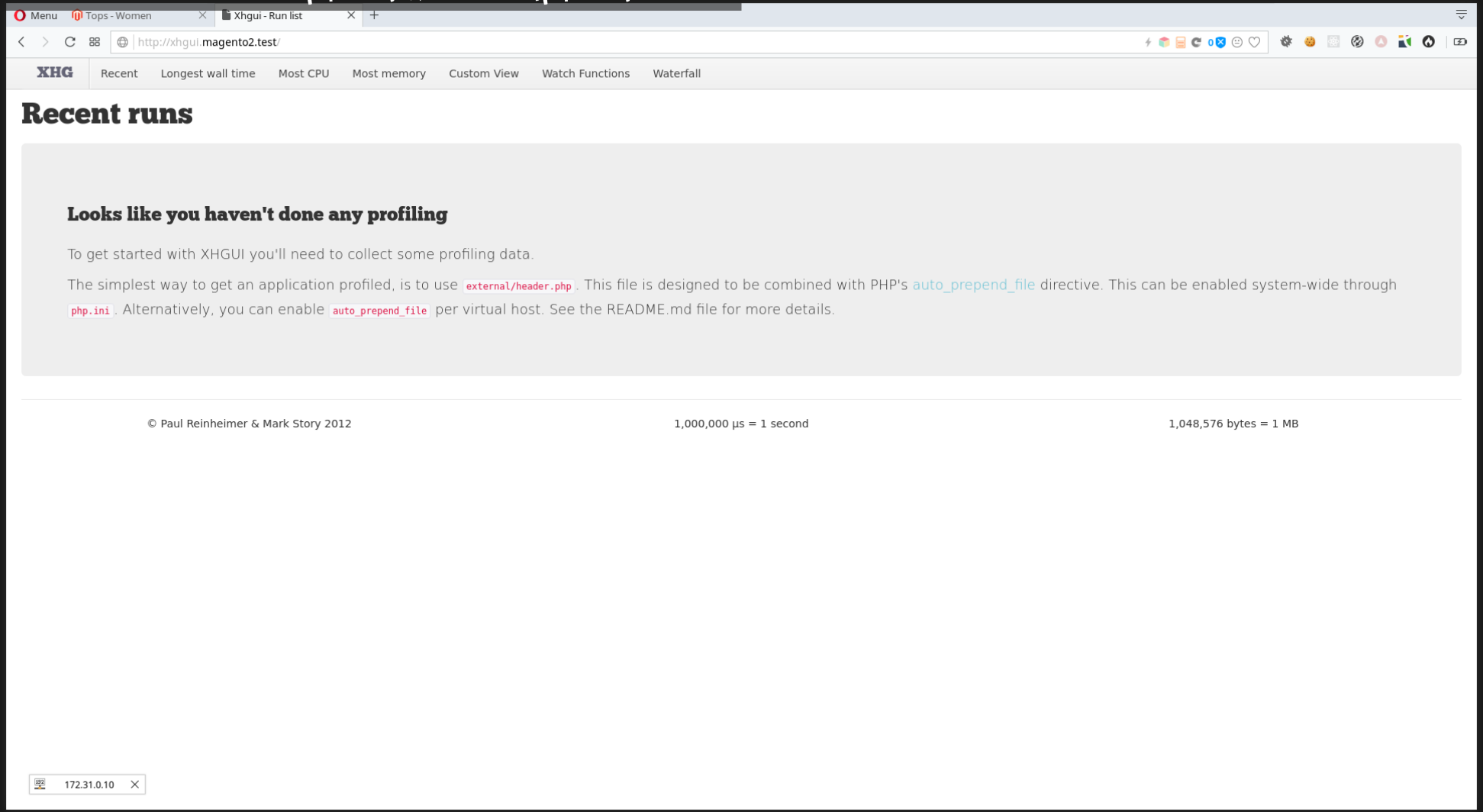
How can we profile: Xhprof

```
extension=tideways.so
tideways.auto_prepend_library=0
auto_prepend_file=/usr/share/xhprof/prepend.php
```

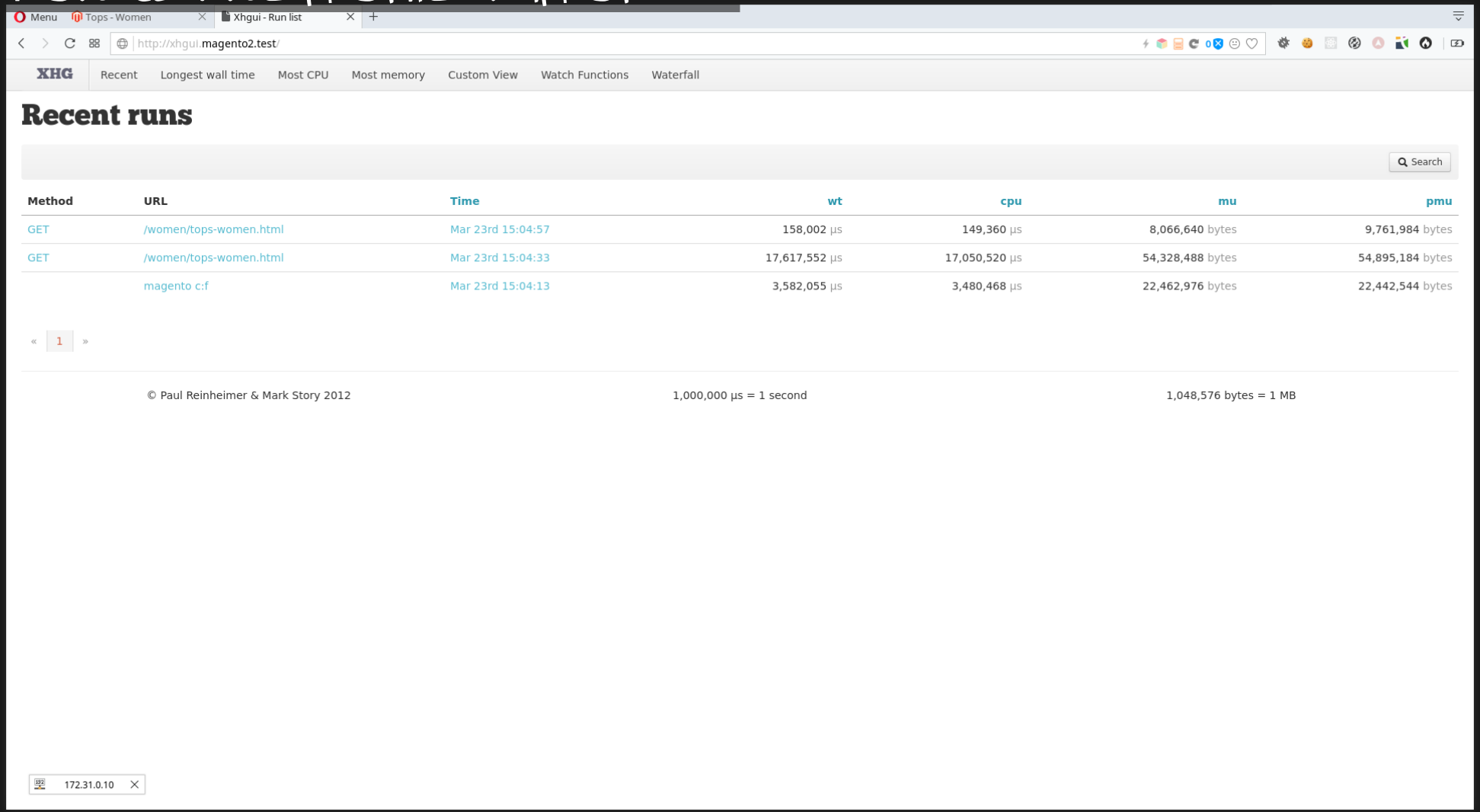
How can we profile: Xhprof

```php
<?php

if (extension_loaded('tideways')) {
    tideways_enable(
        TIDEWAYS_FLAGS_CPU
        | TIDEWAYS_FLAGS_MEMORY
        | TIDEWAYS_FLAGS_NO_SPANS
    );

    register_shutdown_function(
        function() {
            ignore_user_abort(true);
            flush();

            $data['profile'] = tideways_disable();

            // ...
```

How can we profile: Xhprof



How can we profile: Xhprof



How can we profile: Xhprof

Menu | Tops - Women | Xhgui - Profile - /wome

http://xhgui.magento2.test/run/view?id=5ab5179402d00e013503ff12

XHG | Recent | Longest wall time | Most CPU | Most memory | Custom View | Watch Functions | Waterfall

Profile data for GET /women/tops-women.html

THIS RUN

URL /women/tops-women.html
Time Mar 23rd 15:04:33
ID 5ab5179402d00e013503ff12
Wall Time 17,617,552 µs
CPU Time 17,050,520 µs
Memory Usage 54,328,488 bytes
Peak Memory Usage 54,895,184 bytes

GET

No GET data

SERVER

HOSTNAME
3af5f753e167
PWD
/phpapp
HOME
/var/www
DEBIAN_FRONTEND
noninteractive
affinity:container
=53e42ae1af7cff0904225f3fef60218a6ae2083991c5c7188d376b6704ab5e3b
PROFILER
xhprof
SHLVL
0
DEVELOPMENT
1
C_UID
1000
PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
C_GID
1000
USER
www-data
HTTP_COOKIE
mage-translation-storage=%7B%7D; mage-translation-file-version=%7B%7D; form_key=rN9d5od56Ql76qv9; mage-cache-storage=%7B%7D; mage-cache-storage-s=%7B%7D; mage-cache-sessid=true; mage-messages=; recently_viewed_product=%7B%

172.31.0.10

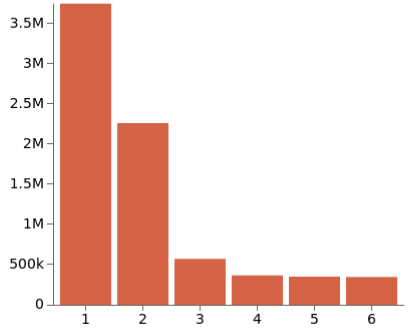
Jump to functions | View Callgraph | View Flamegraph | Compare this run

Watch Functions

Function	Call Count	ewt	emu	epmu

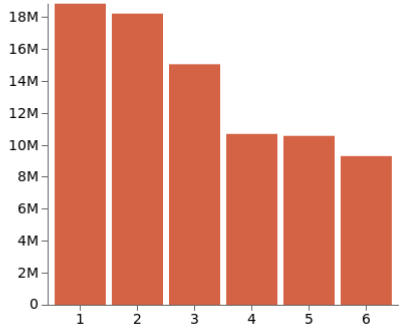
You have no watch functions setup. Add a watch function now.

Self Wall Time



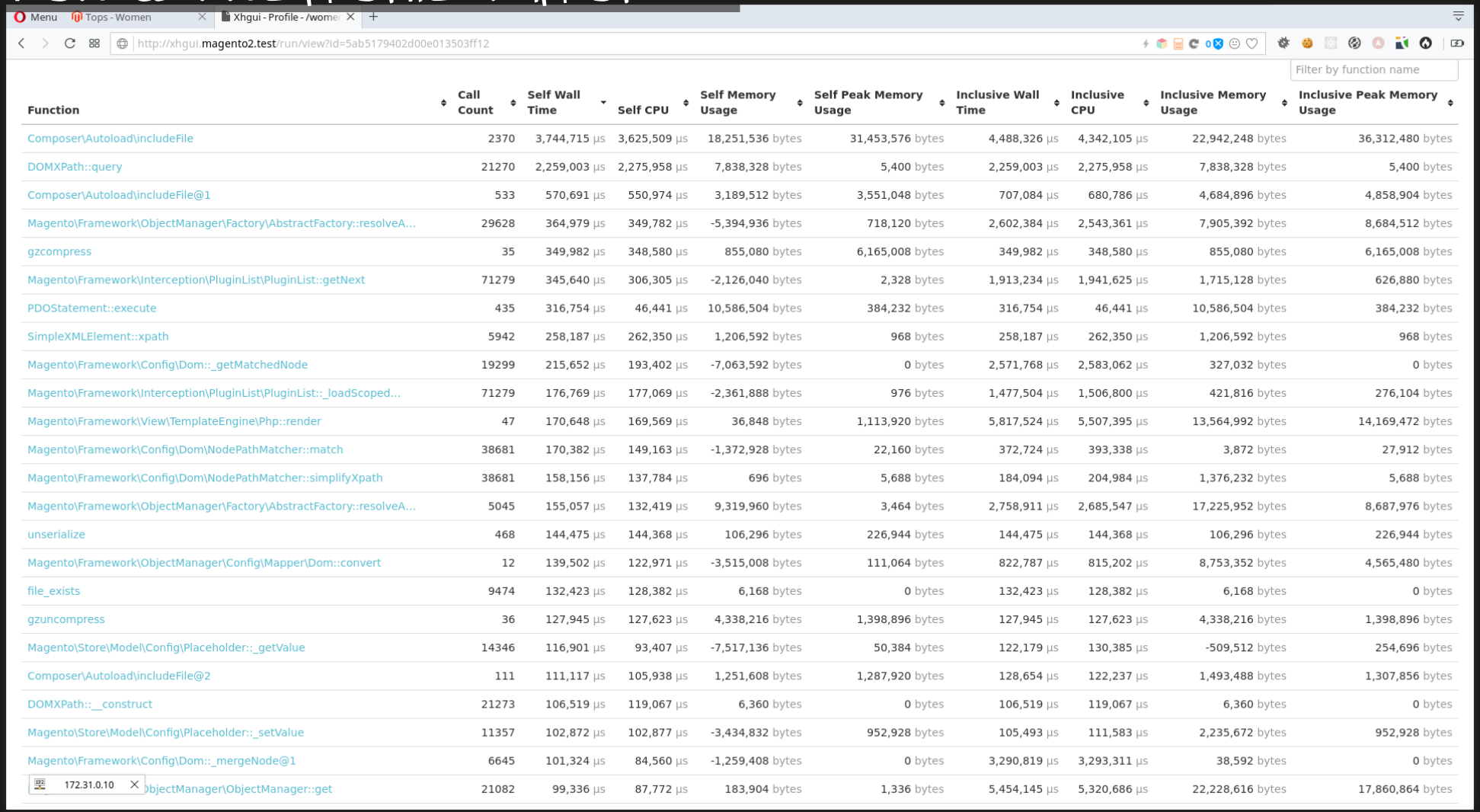
Composer\Autoload\includeFile
3,744,715 µs
DOMXPath::query
2,259,003 µs
Composer\Autoload\includeFile@1
570,691 µs
Magento\Framework\ObjectManager\Factory\AbstractFactory::resolveArgument
364,979 µs
gzcompress
349,982 µs
Magento\Framework\Interception\PluginList\PluginList::getNext
345,640 µs

Memory Hogs



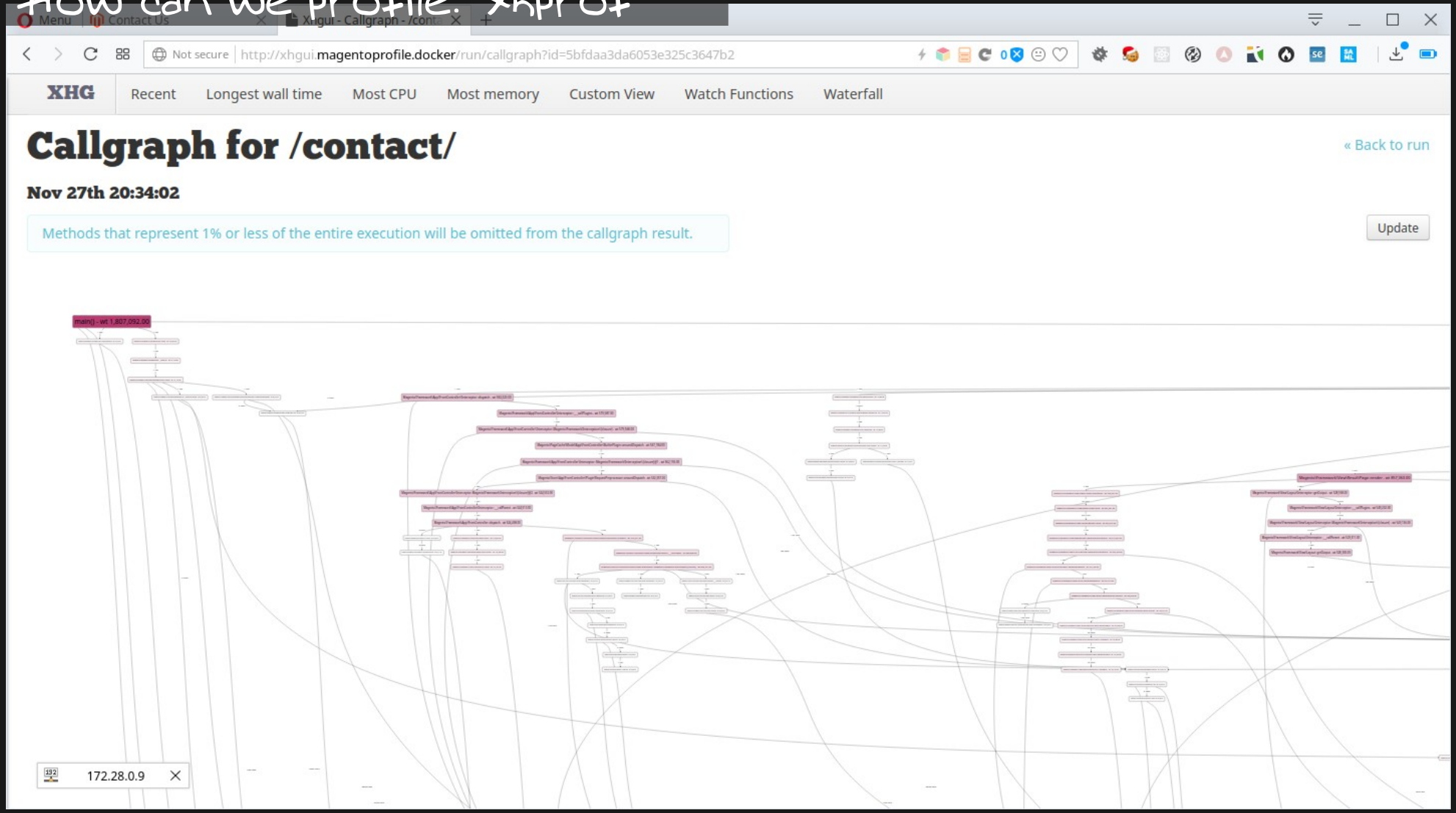
explode
18,874,232 bytes
Composer\Autoload\includeFile
18,251,536 bytes
array_merge
15,076,584 bytes
FilesystemIterator::current
10,708,592 bytes
PDOStatement::execute
10,586,504 bytes
Magento\Framework\ObjectManager\Factory\AbstractFactory::resolveArgumentsInRuntime
9,319,960 bytes

How can we profile: Xhprof



Function	Call Count	Self Wall Time	Self CPU	Self Memory Usage	Self Peak Memory Usage	Inclusive Wall Time	Inclusive CPU	Inclusive Memory Usage	Inclusive Peak Memory Usage
Composer\Autoload\includeFile	2370	3,744,715 µs	3,625,509 µs	18,251,536 bytes	31,453,576 bytes	4,488,326 µs	4,342,105 µs	22,942,248 bytes	36,312,480 bytes
DOMXPath::query	21270	2,259,003 µs	2,275,958 µs	7,838,328 bytes	5,400 bytes	2,259,003 µs	2,275,958 µs	7,838,328 bytes	5,400 bytes
Composer\Autoload\includeFile@1	533	570,691 µs	550,974 µs	3,189,512 bytes	3,551,048 bytes	707,084 µs	680,786 µs	4,684,896 bytes	4,858,904 bytes
Magento\Framework\ObjectManager\Factory\AbstractFactory::resolveA...	29628	364,979 µs	349,782 µs	-5,394,936 bytes	718,120 bytes	2,602,384 µs	2,543,361 µs	7,905,392 bytes	8,684,512 bytes
gzcompress	35	349,982 µs	348,580 µs	855,080 bytes	6,165,008 bytes	349,982 µs	348,580 µs	855,080 bytes	6,165,008 bytes
Magento\Framework\Interception\PluginList\PluginList::getNext	71279	345,640 µs	306,305 µs	-2,126,040 bytes	2,328 bytes	1,913,234 µs	1,941,625 µs	1,715,128 bytes	626,880 bytes
PDOStatement::execute	435	316,754 µs	46,441 µs	10,586,504 bytes	384,232 bytes	316,754 µs	46,441 µs	10,586,504 bytes	384,232 bytes
SimpleXMLElement::xpath	5942	258,187 µs	262,350 µs	1,206,592 bytes	968 bytes	258,187 µs	262,350 µs	1,206,592 bytes	968 bytes
Magento\Framework\Config\Dom::_getMatchedNode	19299	215,652 µs	193,402 µs	-7,063,592 bytes	0 bytes	2,571,768 µs	2,583,062 µs	327,032 bytes	0 bytes
Magento\Framework\Interception\PluginList\PluginList::_loadScoped...	71279	176,769 µs	177,069 µs	-2,361,888 bytes	976 bytes	1,477,504 µs	1,506,800 µs	421,816 bytes	276,104 bytes
Magento\Framework\View\TemplateEngine\Php::render	47	170,648 µs	169,569 µs	36,848 bytes	1,113,920 bytes	5,817,524 µs	5,507,395 µs	13,564,992 bytes	14,169,472 bytes
Magento\Framework\Config\Dom\NodePathMatcher::match	38681	170,382 µs	149,163 µs	-1,372,928 bytes	22,160 bytes	372,724 µs	393,338 µs	3,872 bytes	27,912 bytes
Magento\Framework\Config\Dom\NodePathMatcher::simplifyXpath	38681	158,156 µs	137,784 µs	696 bytes	5,688 bytes	184,094 µs	204,984 µs	1,376,232 bytes	5,688 bytes
Magento\Framework\ObjectManager\Factory\AbstractFactory::resolveA...	5045	155,057 µs	132,419 µs	9,319,960 bytes	3,464 bytes	2,758,911 µs	2,685,547 µs	17,225,952 bytes	8,687,976 bytes
unserialize	468	144,475 µs	144,368 µs	106,296 bytes	226,944 bytes	144,475 µs	144,368 µs	106,296 bytes	226,944 bytes
Magento\Framework\ObjectManager\Config\Mapper\Dom::convert	12	139,502 µs	122,971 µs	-3,515,008 bytes	111,064 bytes	822,787 µs	815,202 µs	8,753,352 bytes	4,565,480 bytes
file_exists	9474	132,423 µs	128,382 µs	6,168 bytes	0 bytes	132,423 µs	128,382 µs	6,168 bytes	0 bytes
gzuncompress	36	127,945 µs	127,623 µs	4,338,216 bytes	1,398,896 bytes	127,945 µs	127,623 µs	4,338,216 bytes	1,398,896 bytes
Magento\Store\Model\Config\Placeholder::_getValue	14346	116,901 µs	93,407 µs	-7,517,136 bytes	50,384 bytes	122,179 µs	130,385 µs	-509,512 bytes	254,696 bytes
Composer\Autoload\includeFile@2	111	111,117 µs	105,938 µs	1,251,608 bytes	1,287,920 bytes	128,654 µs	122,237 µs	1,493,488 bytes	1,307,856 bytes
DOMXPath::__construct	21273	106,519 µs	119,067 µs	6,360 bytes	0 bytes	106,519 µs	119,067 µs	6,360 bytes	0 bytes
Magento\Store\Model\Config\Placeholder::_setValue	11357	102,872 µs	102,877 µs	-3,434,832 bytes	952,928 bytes	105,493 µs	111,583 µs	2,235,672 bytes	952,928 bytes
Magento\Framework\Config\Dom::_mergeNode@1	6645	101,324 µs	84,560 µs	-1,259,408 bytes	0 bytes	3,290,819 µs	3,293,311 µs	38,592 bytes	0 bytes
bjectManager\ObjectManager::get	21082	99,336 µs	87,772 µs	183,904 bytes	1,336 bytes	5,454,145 µs	5,320,686 µs	22,228,616 bytes	17,860,864 bytes

How can we profile: Xhprof



How can we profile: Xhprof

Local setup

Profiles everything

How can we profile: Xhprof

Local setup

Profiles everything

Not so fancy

How can we profile: Tideways

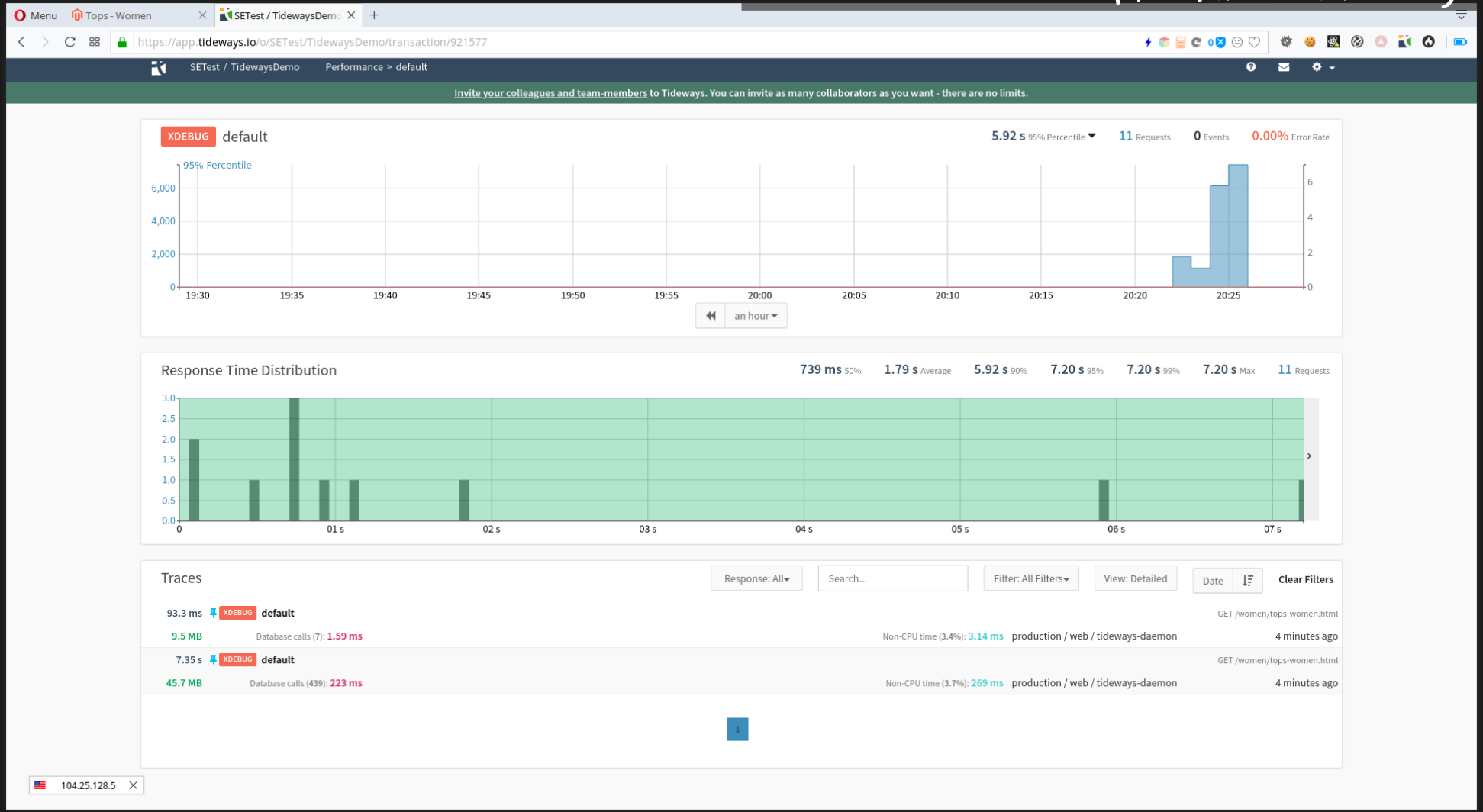
Tideways

How can we profile: Tideways

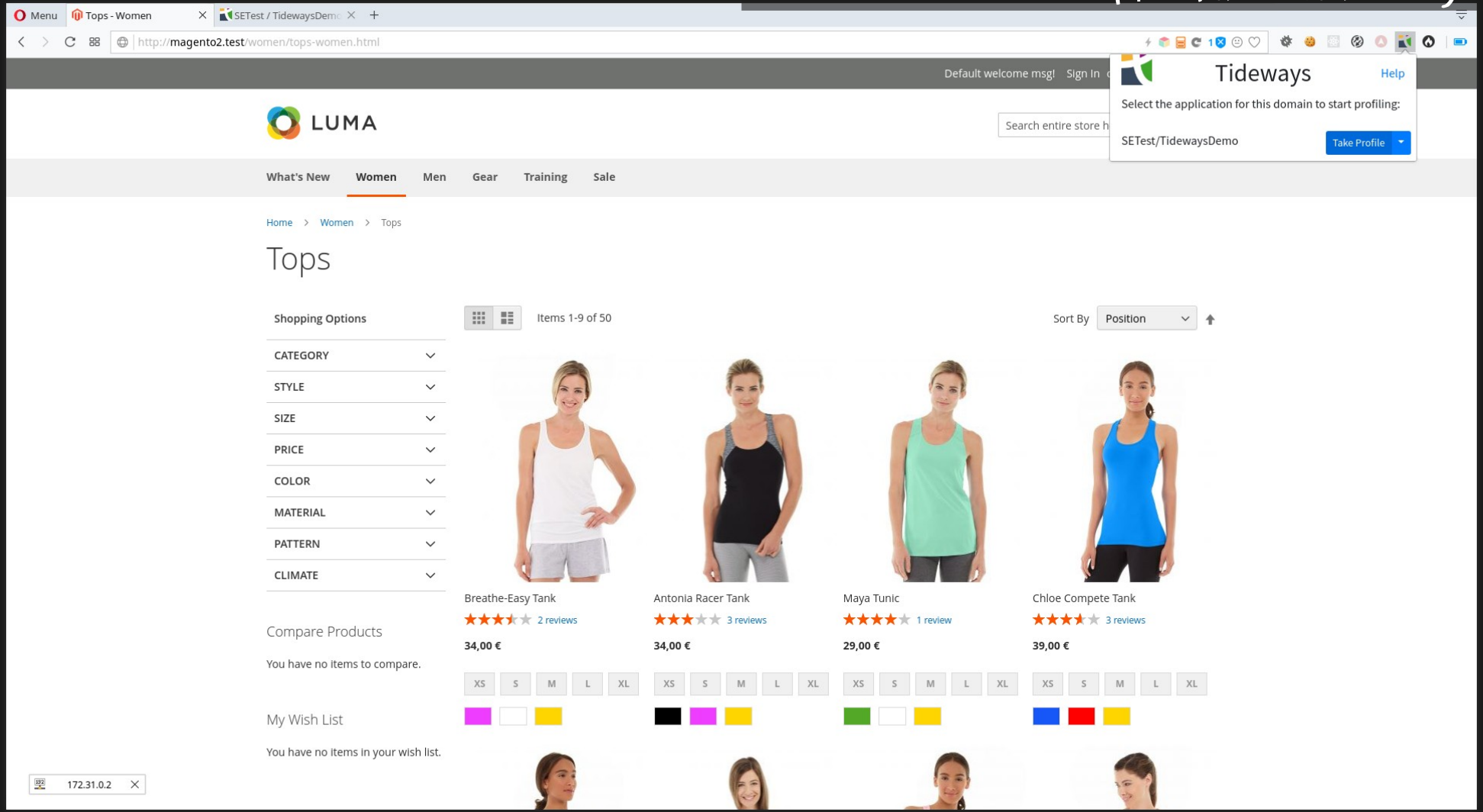
Tideways php extension
Daemon for communication
Browser extension
Configuration needed
Token to identify
Hosted web interface
Also has APM functionality

30 day trial

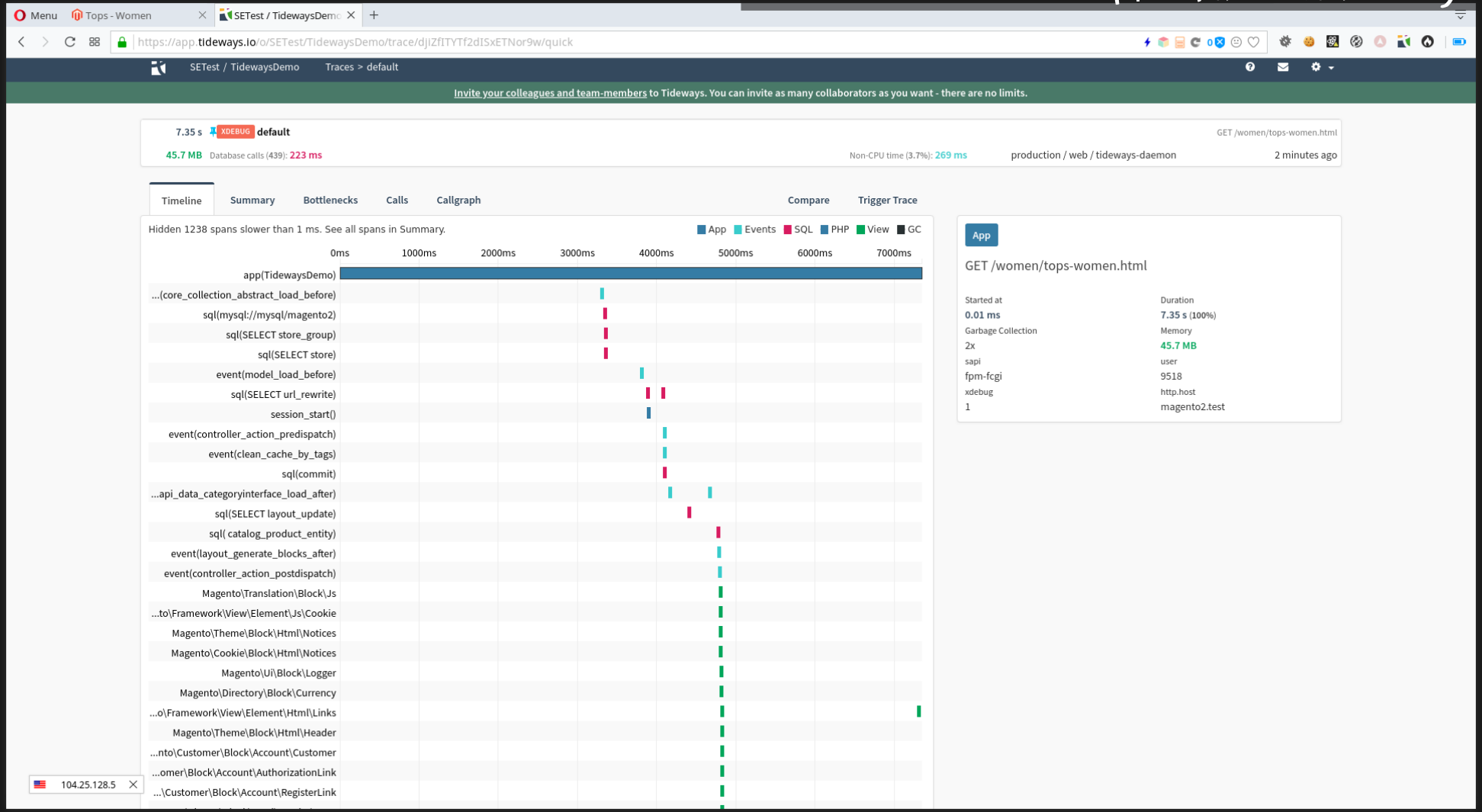
How can we profile: Tideways



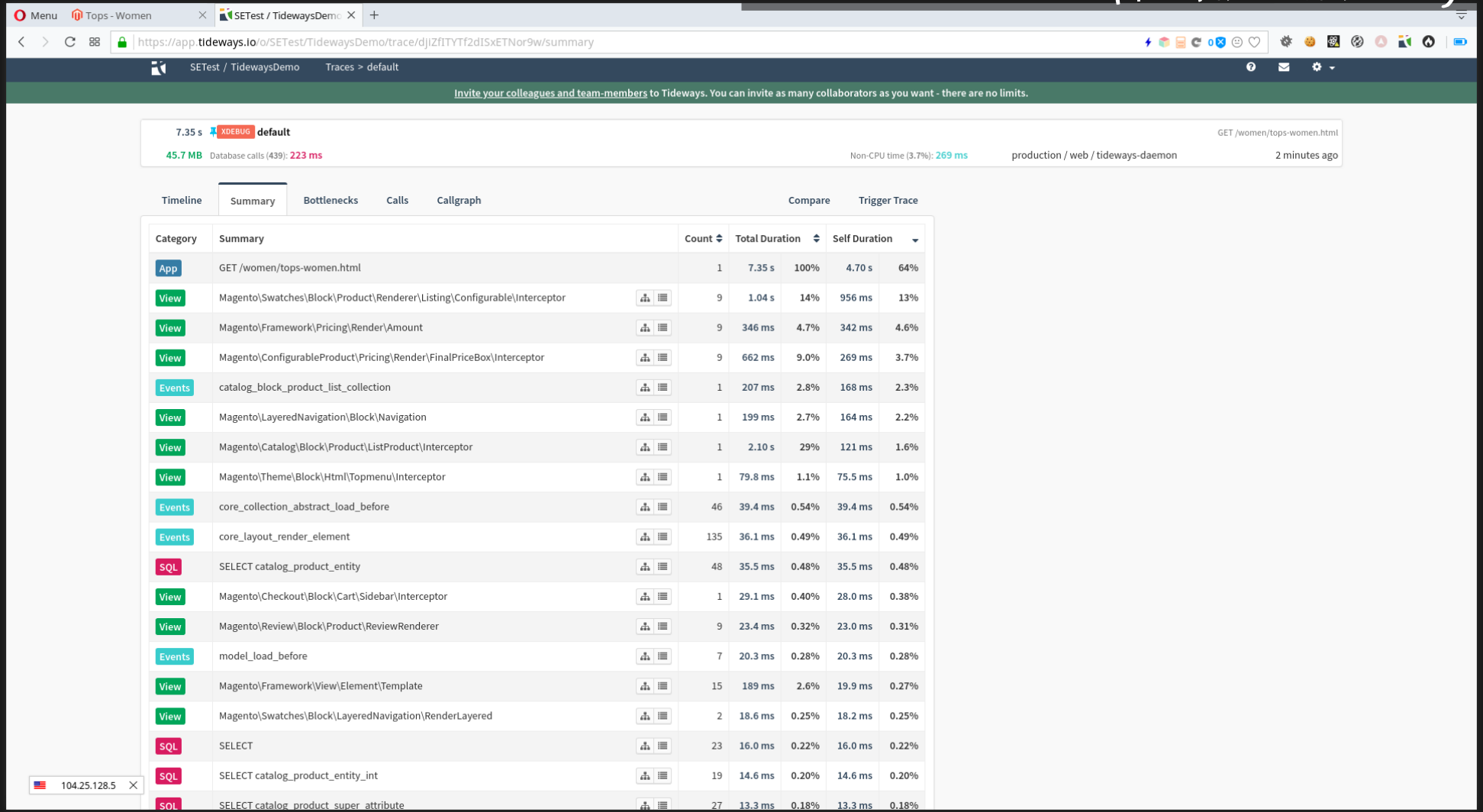
How can we profile: Tideways



How can we profile: Tideways

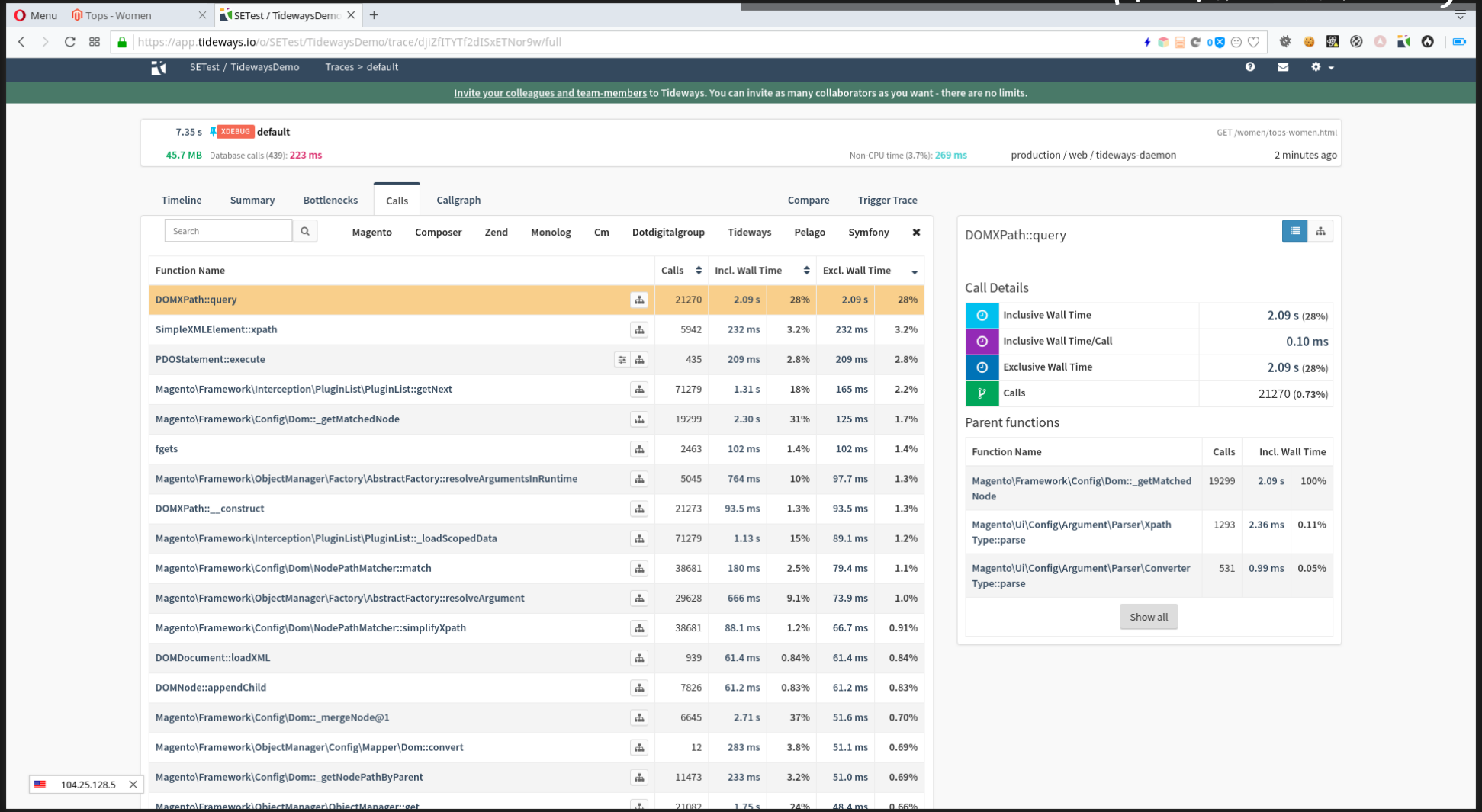


How can we profile: Tideways



Category	Summary	Count	Total Duration		Self Duration	
App	GET /women/tops-women.html	1	7.35 s	100%	4.70 s	64%
View	Magento\Swatches\Block\Product\Renderer\Listing\Configurable\Interceptor	9	1.04 s	14%	956 ms	13%
View	Magento\Framework\Pricing\Render\Amount	9	346 ms	4.7%	342 ms	4.6%
View	Magento\ConfigurableProduct\Pricing\Render\FinalPriceBox\Interceptor	9	662 ms	9.0%	269 ms	3.7%
Events	catalog_block_product_list_collection	1	207 ms	2.8%	168 ms	2.3%
View	Magento\LayeredNavigation\Block\Navigation	1	199 ms	2.7%	164 ms	2.2%
View	Magento\Catalog\Block\Product\ListProduct\Interceptor	1	2.10 s	29%	121 ms	1.6%
View	Magento\Theme\Block\Html\Topmenu\Interceptor	1	79.8 ms	1.1%	75.5 ms	1.0%
Events	core_collection_abstract_load_before	46	39.4 ms	0.54%	39.4 ms	0.54%
Events	core_layout_render_element	135	36.1 ms	0.49%	36.1 ms	0.49%
SQL	SELECT catalog_product_entity	48	35.5 ms	0.48%	35.5 ms	0.48%
View	Magento\Checkout\Block\Cart\Sidebar\Interceptor	1	29.1 ms	0.40%	28.0 ms	0.38%
View	Magento\Review\Block\Product\ReviewRenderer	9	23.4 ms	0.32%	23.0 ms	0.31%
Events	model_load_before	7	20.3 ms	0.28%	20.3 ms	0.28%
View	Magento\Framework\View\Element\Template	15	189 ms	2.6%	19.9 ms	0.27%
View	Magento\Swatches\Block\LayeredNavigation\RenderLayered	2	18.6 ms	0.25%	18.2 ms	0.25%
SQL	SELECT	23	16.0 ms	0.22%	16.0 ms	0.22%
SQL	SELECT catalog_product_entity_int	19	14.6 ms	0.20%	14.6 ms	0.20%
SQL	SELECT catalog_product_super_attribute	27	13.3 ms	0.18%	13.3 ms	0.18%

How can we profile: Tideways



How can we profile: Tideways

More than a profiler

Easy to use

How can we profile: Tideways

More than a profiler

Easy to use

Trial → Paying

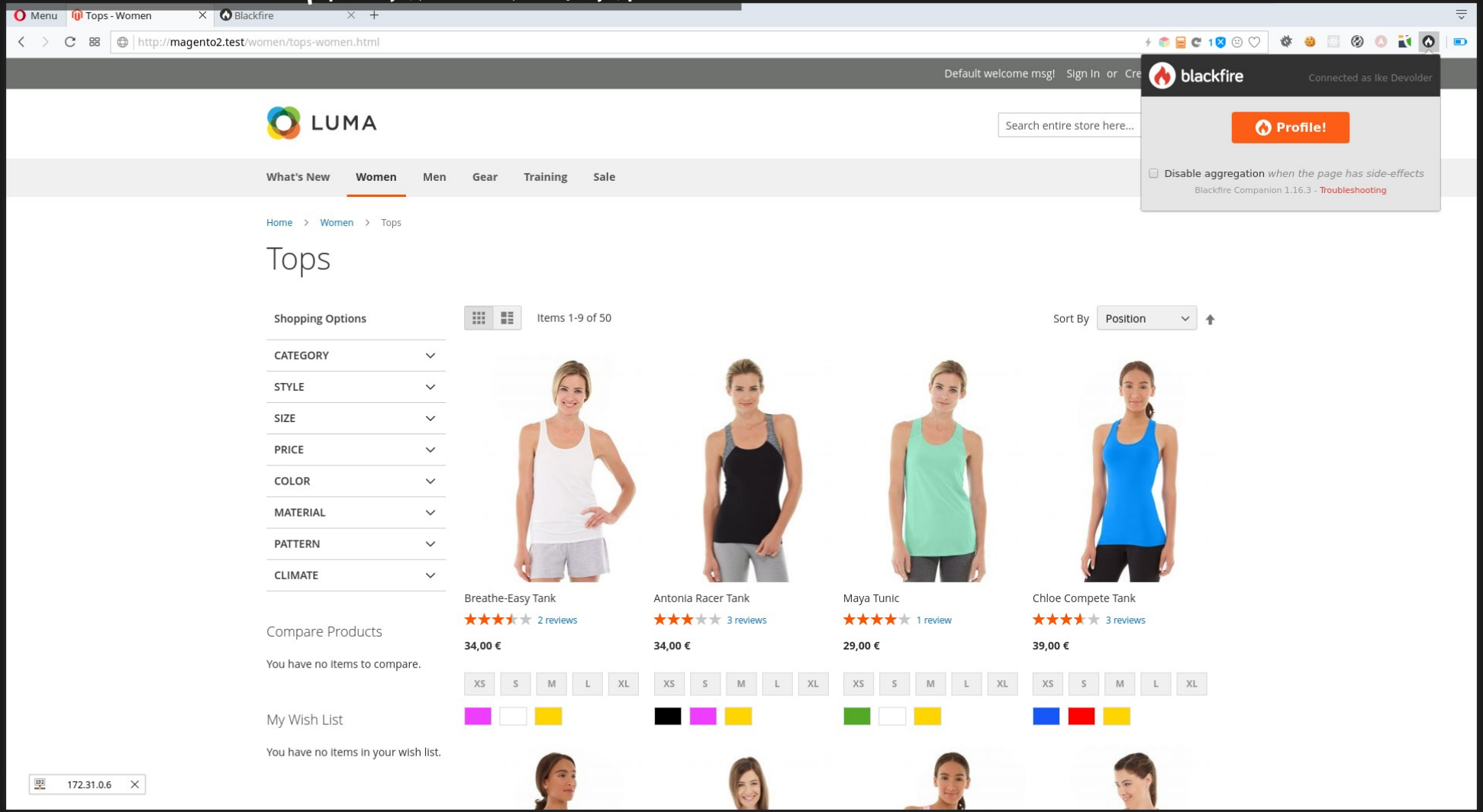
How can we profile: Blackfire

Blackfire

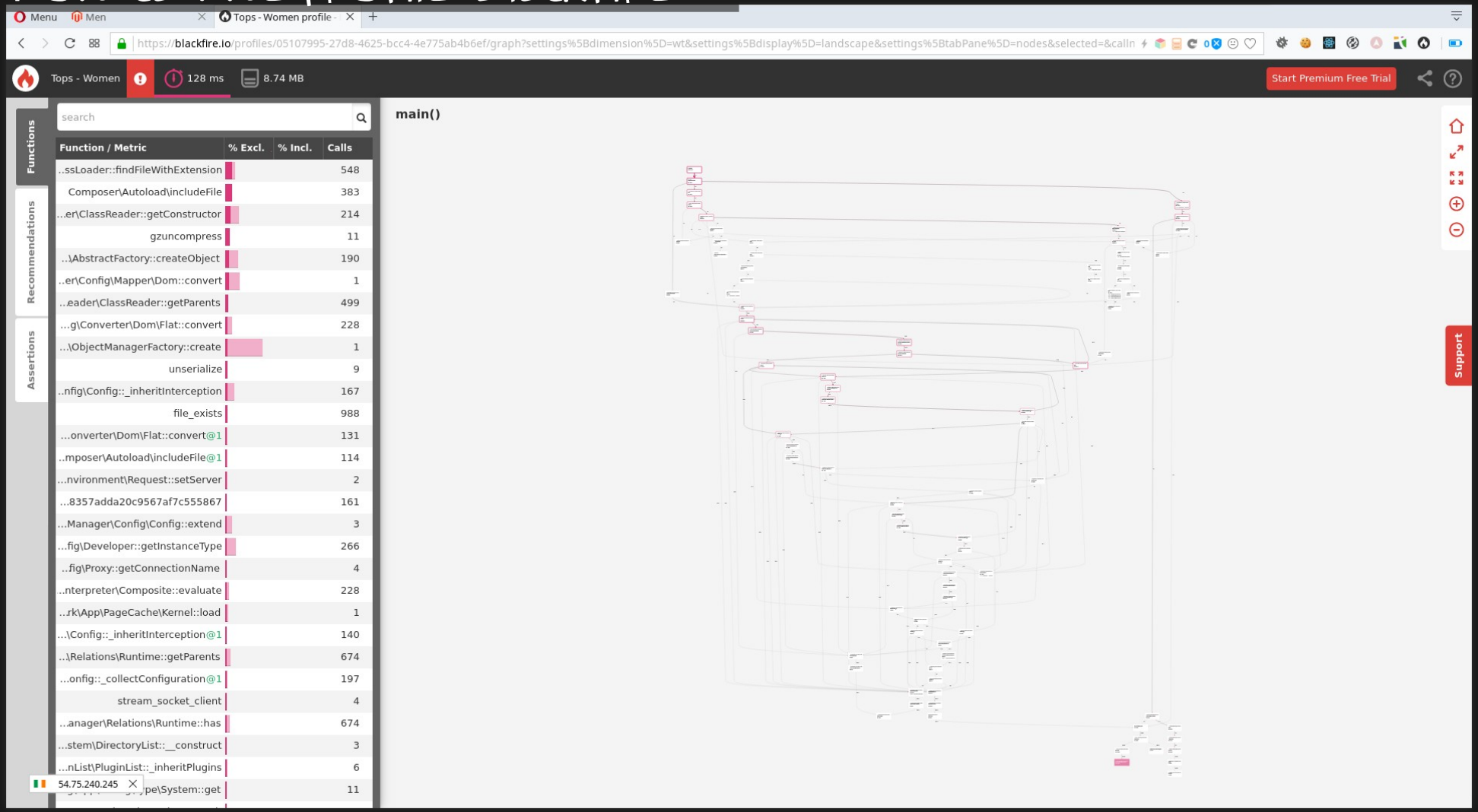
Blackfire php extension
Daemon for communication
Browser extension
Configuration needed
Token to identify
Hosted web interface

Freemium

How can we profile: Blackfire



How can we profile: Blackfire



How can we profile: Blackfire

Relatively easy to use

How can we profile: Blackfire

Relatively easy to use

Freemium

→ not all features

Comparing

XdeBug

Local use, easy to setup

Relatively hard to interpret the results correctly

Xhprof/Xhgui

Local use, some work to setup

Fairly easy to use, UI looks a bit dated

Tideways

Cloud based

30 day trial with all features

Easy to use, nice UI + additional APM functionality

Blackfire

Cloud based

Fremium, not all features for free testing

Easy to use

Great new tool: Phpspy

Phpspy

Great new tool: Phpspy

Sampling Profiler

Written in C

Must be build against the correct version of PHP

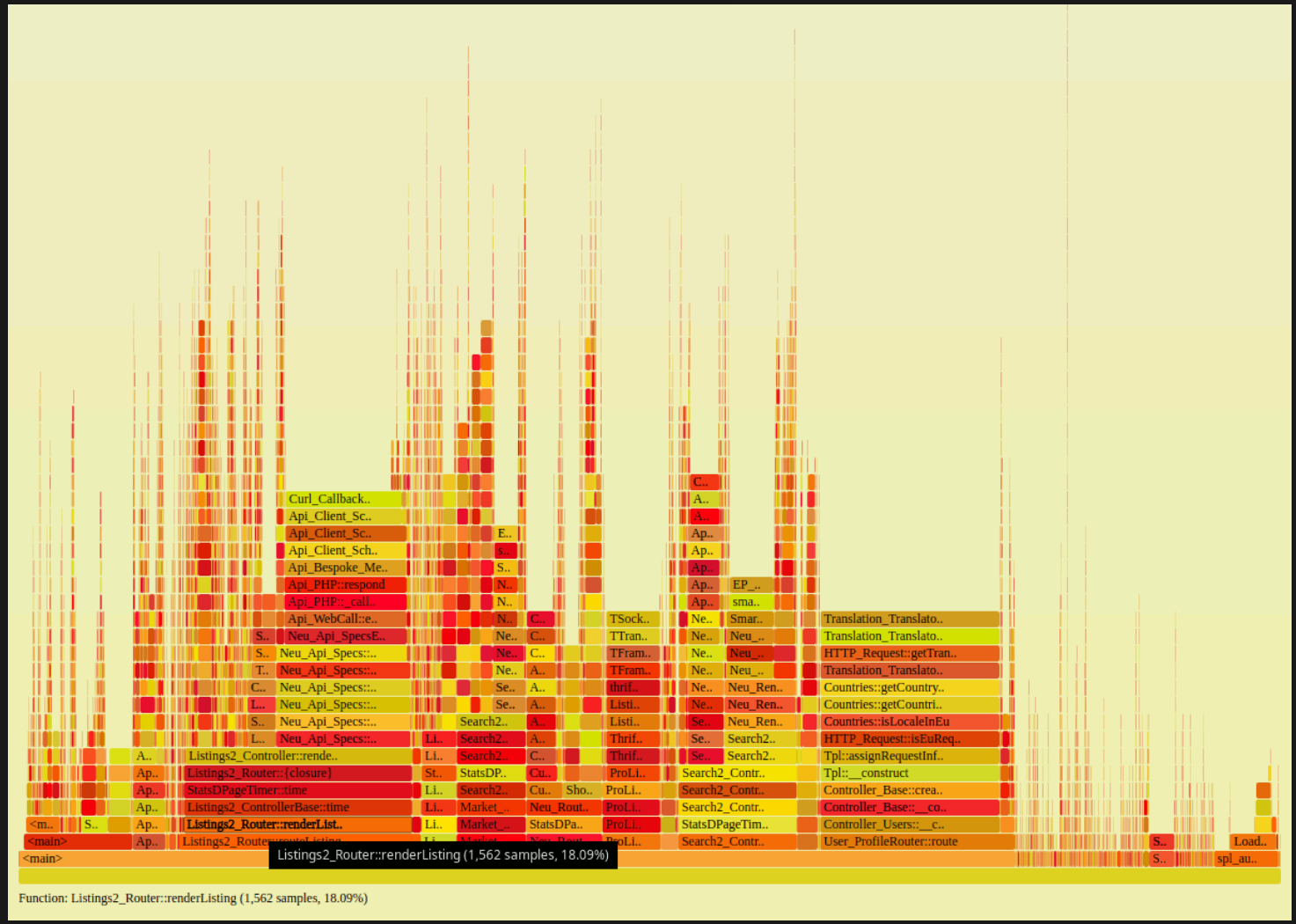
Linux only for the moment

Great new tool: Phpspy

```
phpspy -V73 -@ -P -x php
samp_count=1  total_err_count=0  func_count=5

tincl       texcl       incl        excl        incl%    excl%    func
1           1           1           1           100.00   100.00   stream_select <inter
1           0           1           0           100.00   0.00     system_with_timeout
1           0           1           0           100.00   0.00     run_test /home/adam/
1           0           1           0           100.00   0.00     run_all_tests /home/
1           0           1           0           100.00   0.00     <main> /home/adam/ph
```

Great new tool: Phpspy

Great new tool: Phpspy



Pick one
Learn it!

Photo by rihaij on Pixabay

How to approach profiling

How to approach profiling

Photo by Victor Van Welden on Unsplash

What first?

How to approach profiling

Photo by Victor Van Welden on Unsplash

What first?

- Slow things

How to approach profiling

Photo by Victor Van Welden on Unsplash

What first?

- Slow things
- popular things

How to approach profiling

Photo by Victor Van Welden on Unsplash

What first?

- Slow things
- popular things
- application initialisation

How to approach profiling

Analyse the profile

- hot path
- find the cause
- fix it
- …
- 👍

Photo by Aaron Sunday on Pixabay

demo

Photo by Ha Ess on Pixabay

Photo by rihaij on Pixabay

Popular bottlenecks

Photo by rihaij on Pixabay

Slow code

Popular Bottlenecks

Photo by rihaij on Pixabay

Slow code

Bad SQL queries

Popular Bottlenecks

Slow code

Bad SQL queries

Disk I/O

Popular Bottlenecks

Slow code

Bad SQL queries

Disk I/O

Bad external services

And then it happens, slowness in production

Photo by Aaron Sunday on Pixabay

Slowness in Production

First response

Don't Panic

Photo by rihaij on Pixabay

Slowness in Production

First response

Don't Panic

Add more CPU

Photo by rihaij on Pixabay

Slowness in Production

First response

Don't Panic

Add more CPU

Add more RAM

Photo by rihaij on Pixabay

Slowness in Production

First response

Don't Panic

Add more CPU

Add more RAM

Block malicious actors

Photo by rihaij on Pixabay

When system is normalized

- Find slow parts
- Start profiling

Some usefull tools in production

- Monitoring
- APM
- atop
- strace
- ...

Photo by rihaij on Pixabay

Closing

- find slower things while developing
- xhprof + xhgui
- don't trust profiles from systems under stress
- make popular pages cachable
- profile periodicaly

Questions?

I'm Ike, Developer / Operations
at Studio Emma

- @BlackIkeEagle
- PHP-WVL
- DockerWest
- Parchive
- BlackIkeEagle
- Archlinux TU
- VdeBug
- herecura.eu

Thanks

I'm Ike, Developer / Operations
at Studio Emma

- @BlackIkeEagle
- PHP-WVL
- DockerWest
- Parchive
- BlackIkeEagle
- Archlinux TU
- VdeBug
- herecura.eu