



HipHop for



przedstawia

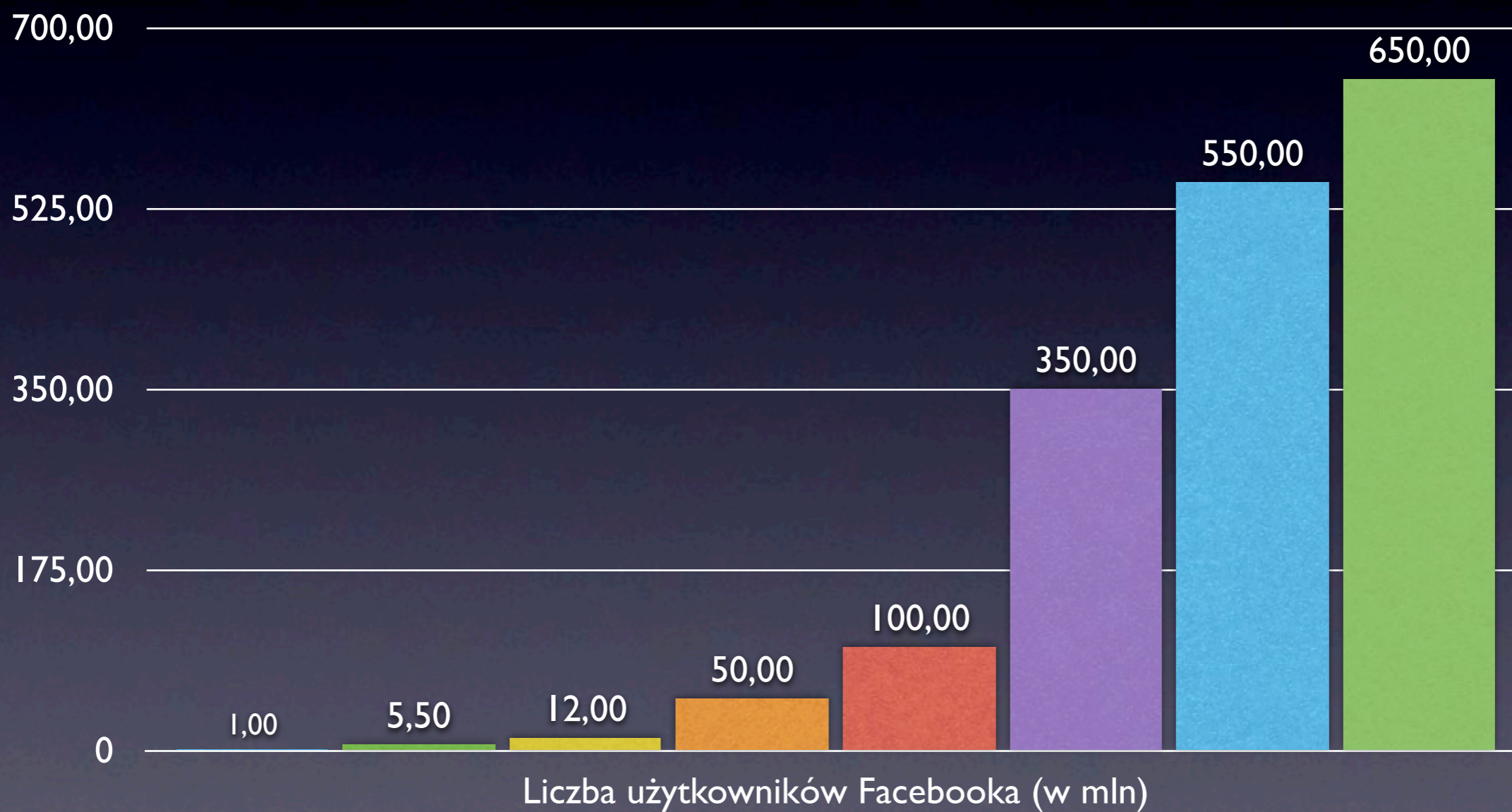
PIAP

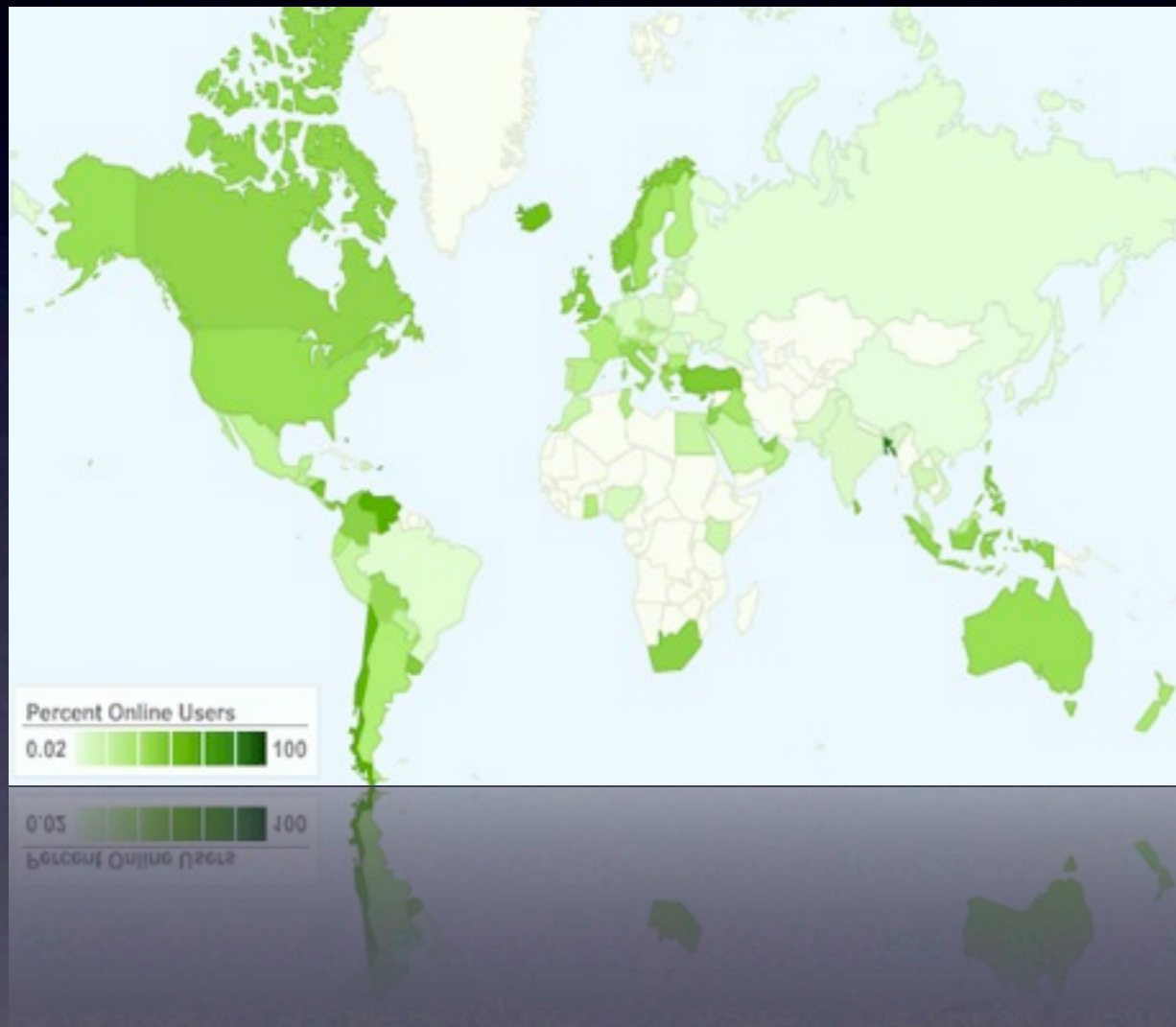


Mariusz Gil

BIO

facebook®

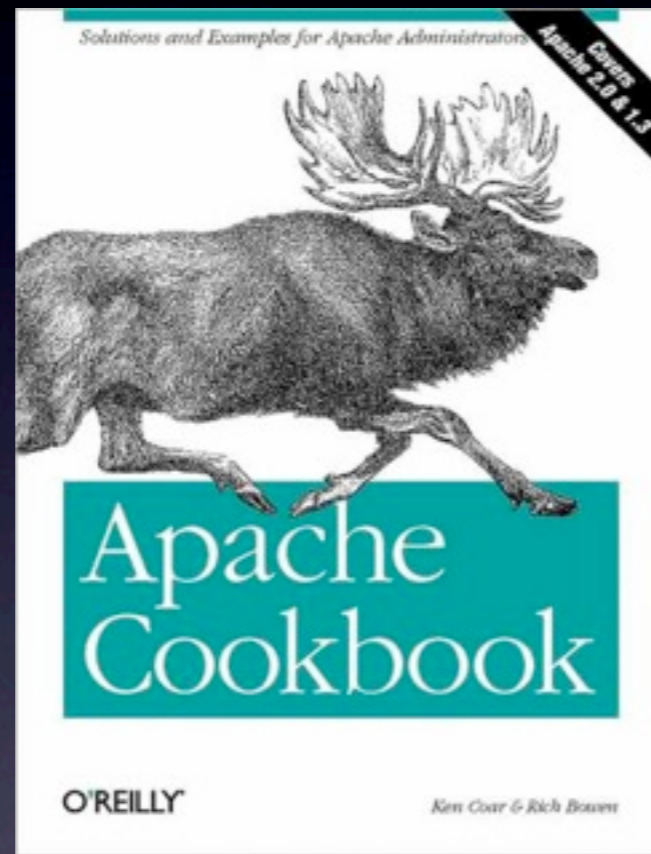
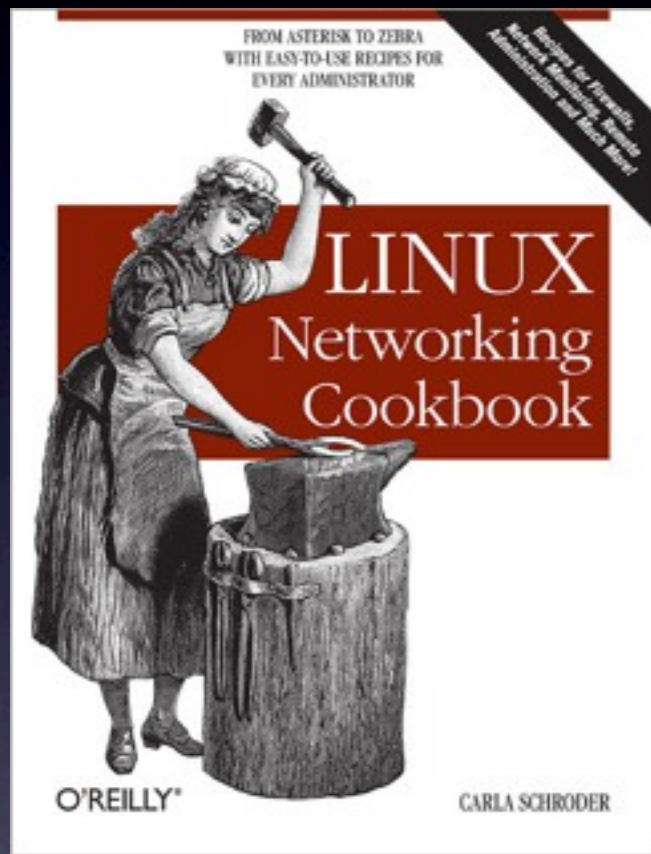


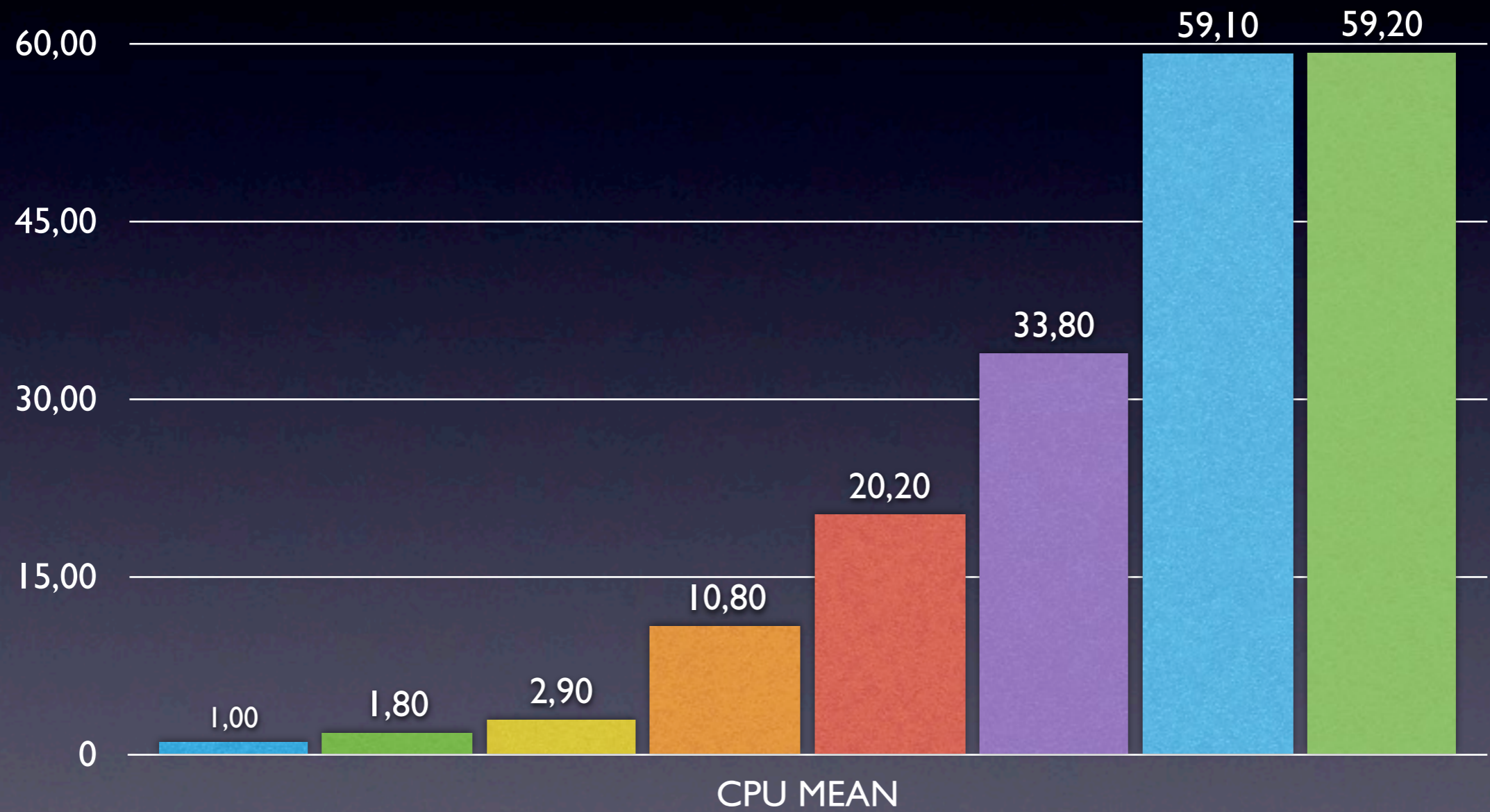


- 650 mln użytkowników
- 70% użytkowników pochodzi spoza USA
- 700 mld odśłon mies.
- 100 mln zdjęć dziennie
- 9.5% ruchu Internetu



- 60.000 serwerów
- 9 data centers w USA
- 4.5 MW zużycia prądu
- 25.000 stóp powierzchni
- 50 mln \$ kosztów
- Projekt Prineville





<http://shootout.alioth.debian.org/u64q/benchmark.php?test=all&lang=all>

Wniosek?

PHP jest wolne

Tak, ale...

||



- PHP łatwo się uczy
- PHP łatwo się pisze
- PHP łatwo się czyta
- PHP łatwo się debuguje

Jak przyspieszyć aplikację w PHP?

- Cache danych i widoków
- Opcode caching
- Optymalizacja algorytmów i kodu
- Buildowanie kodu aplikacji
- Przenoszenie kodu do PHP Extensions
- Tuning PHP: kompilacja z ICC, patche

Facebook ma wszystko

**Ale dalej walczy o
szybkość swojej aplikacji**

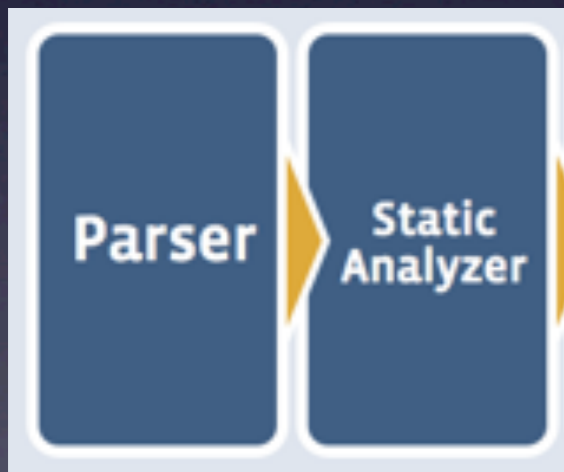
Facebook HipHop dla PHP

Historia

HPHP / HPHPi

Jak to działa?

```
Debian 6.0 64bit [Uruchomiona]
mariusz@debian:~/hiphop/hiphop-php/src$ cat ./test.php
<?php
  echo "Hello world";
?>
mariusz@debian:~/hiphop/hiphop-php/src$ hphp/hphp test.php --keep-tempdir=1 --log=3
running hphp...
creating temporary directory /tmp/hphp_xilAmH ...
parsing inputs...
parsing inputs took 0'00" (8 ms) wall time
pre-optimizing...
pre-optimizing took 0'00" (16 ms) wall time
inferring types...
inferring types took 0'00" (0 ms) wall time
post-optimizing...
post-optimizing took 0'00" (0 ms) wall time
creating CPP files...
creating CPP files took 0'00" (105 ms) wall time
compiling and linking CPP files...
fatal: Not a git repository (or any of the parent directories): .git\n
compiling and linking CPP files took 2'24" (144262 ms) wall time
running executable /tmp/hphp_xilAmH/program --file test.php...
Hello world
all files saved in /tmp/hphp_xilAmH ...
running hphp took 2'25" (145593 ms) wall time
mariusz@debian:~/hiphop/hiphop-php/src$ _
```



Stos technologiczny

load balancer



serwer aplikacyjny



load balancer



serwer aplikacyjny

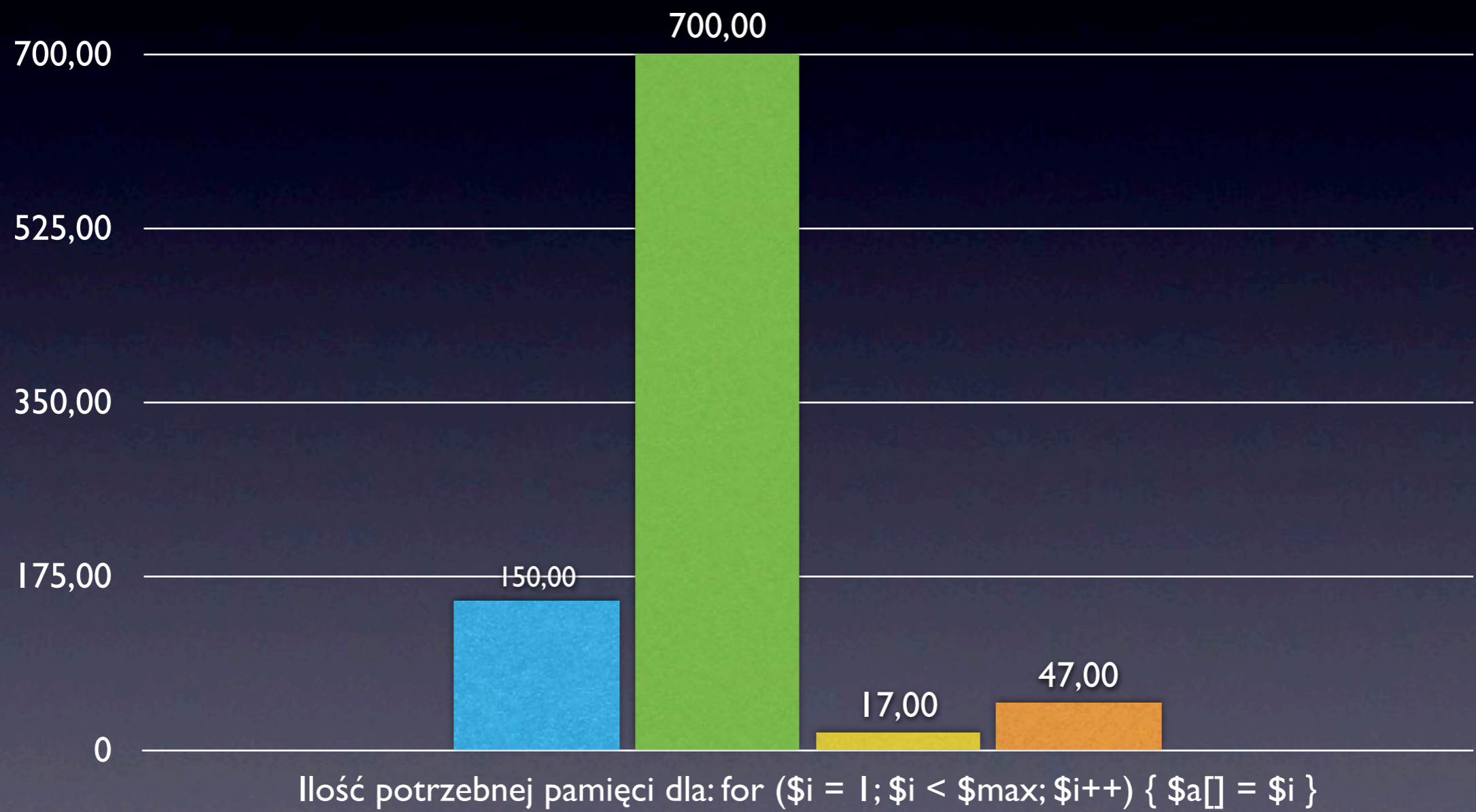


w 6 miesięcy po produkcyjnym wdrożeniu
90% ruchu obsługiwane jest przez HipHop

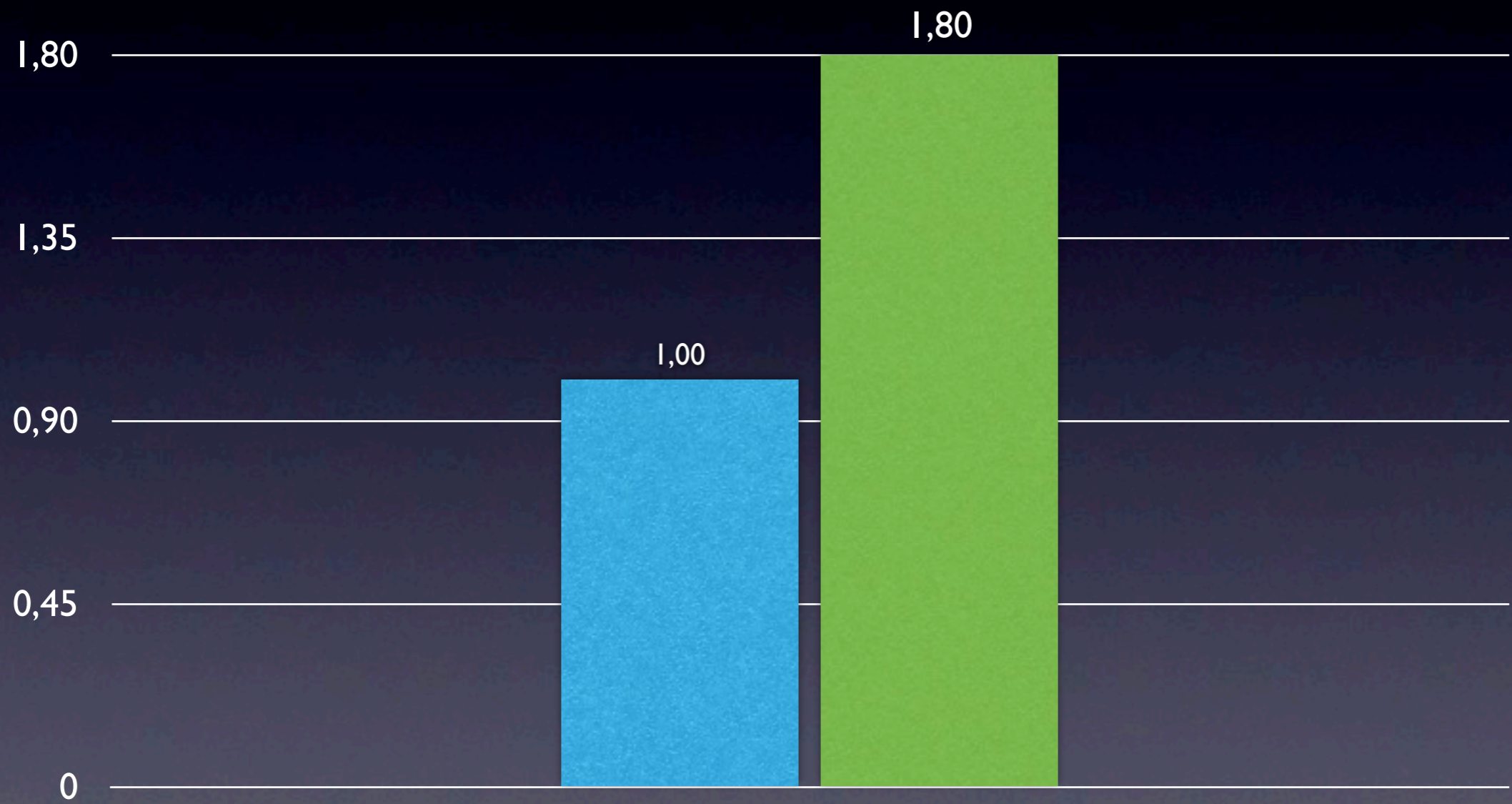


- 50% oszczędności CPU
- 50% oszczędności RAM
- 30% oszczędności CPU przy dwukrotnie większym ruchu z API
- szybsze wykonanie kodu
- brak Apache
- mniej hardware

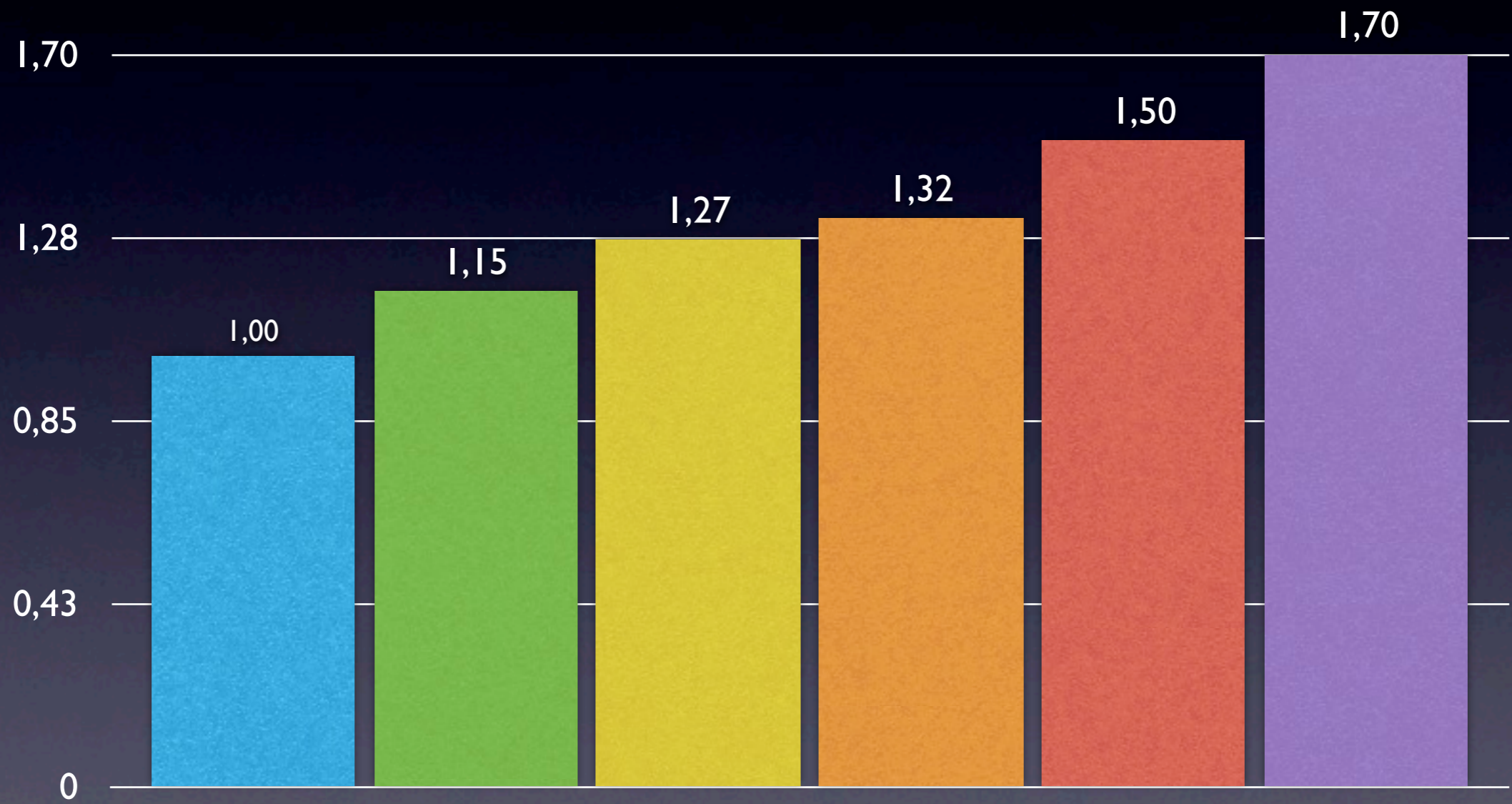
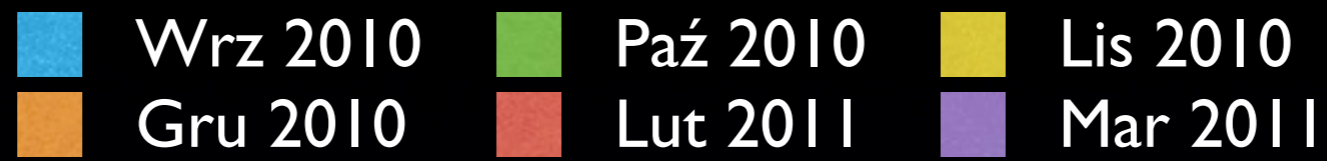
- PHP 5.2, \$max = 1.000.000
- PHP 5.2, \$max = 5.000.000
- HipHop, \$max = 1.000.000
- HipHop, \$max = 5.000.000



■ Lut 2010 ■ Wrz 2010



Relatywny wzrost szybkości HipHop dla PHP w pierwszych 6 miesiącach po premierze



Relatywny wzrost szybkości HipHop dla PHP w ostatnich 6 miesiącach

Ostatnie zmiany to...

- Support dla APC SHM
- Szybsza serializacja i kodowanie JSON
- Mniej liczenia referencji do zmiennych
- Szybsza kompilacja
- Mniejsze pliki wynikowe
- Lepsza alokacja pamięci

Świetnie to wygląda!

Gdzie jest haczyk?



- Wsparcie do PHP 5.2, 99.5% kompatybilności
- Problem z brakiem PHP Extensions
- Brak niektórych funkcji
- Wskazana znajomość języków C/C++



- XHProf ponad XDebug
- Specyficzne wymagania serwerowe
- Platforma 64 bitowa
- Błędy w HipHopie

Dla kogo?

Getting started



<http://github.com/facebook/hiphop-php/>

```
sudo apt-get install cmake g++ libboost-dev flex bison re2c libmysqlclient-  
dev libxml2-dev libmcrypt-dev libicu-dev openssl binutils-dev libcap-dev  
libgd2-xpm-dev zlib1g-dev libtbb-dev libonig-dev libpcre3-dev git-core  
autoconf libtool libcurl4-openssl-dev libboost-system-dev libboost-program-  
options-dev libboost-filesystem-dev
```

```
mkdir hiphop  
cd hiphop  
git clone git://github.com/facebook/hiphop-php  
cd hiphop-php  
export CMAKE_PREFIX_PATH=`/bin/pwd`/../  
export HPHP_HOME=`/bin/pwd`  
export HPHP_LIB=`/bin/pwd`/bin  
git submodule init  
git submodule update  
cd ..
```

```
wget http://www.monkey.org/~provos/libevent-1.4.13-stable.tar.gz  
tar -xzvf libevent-1.4.13-stable.tar.gz  
cd libevent-1.4.13-stable  
cp ../hiphop-php/src/third_party/libevent.fb-changes.diff .  
patch < libevent.fb-changes.diff  
./configure --prefix=$CMAKE_PREFIX_PATH  
make  
make install  
cd ..
```

```
wget http://curl.haxx.se/download/curl-7.20.0.tar.gz  
tar -xvzf curl-7.20.0.tar.gz  
cd curl-7.20.0  
cp ../hiphop-php/src/third_party/libcurl.fb-changes.diff .  
patch -p0 < libcurl.fb-changes.diff  
./configure --prefix=$CMAKE_PREFIX_PATH  
make  
make install  
cd ..
```

```
cd hiphop-php  
cmake .  
make
```

Alternatywy?

- **Roadsend PHP**

PHP kompilowane do C, <http://www.roadsend.com/>

- **PHC compiler**

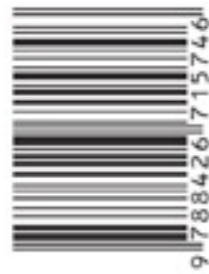
PHP kompilowane do C, <http://www.phpcompiler.org/>

- **Quercus**

Implementacja PHP w Java, <http://www.caucho.com/resin-3.0/quercus/>

- **Phalanger**

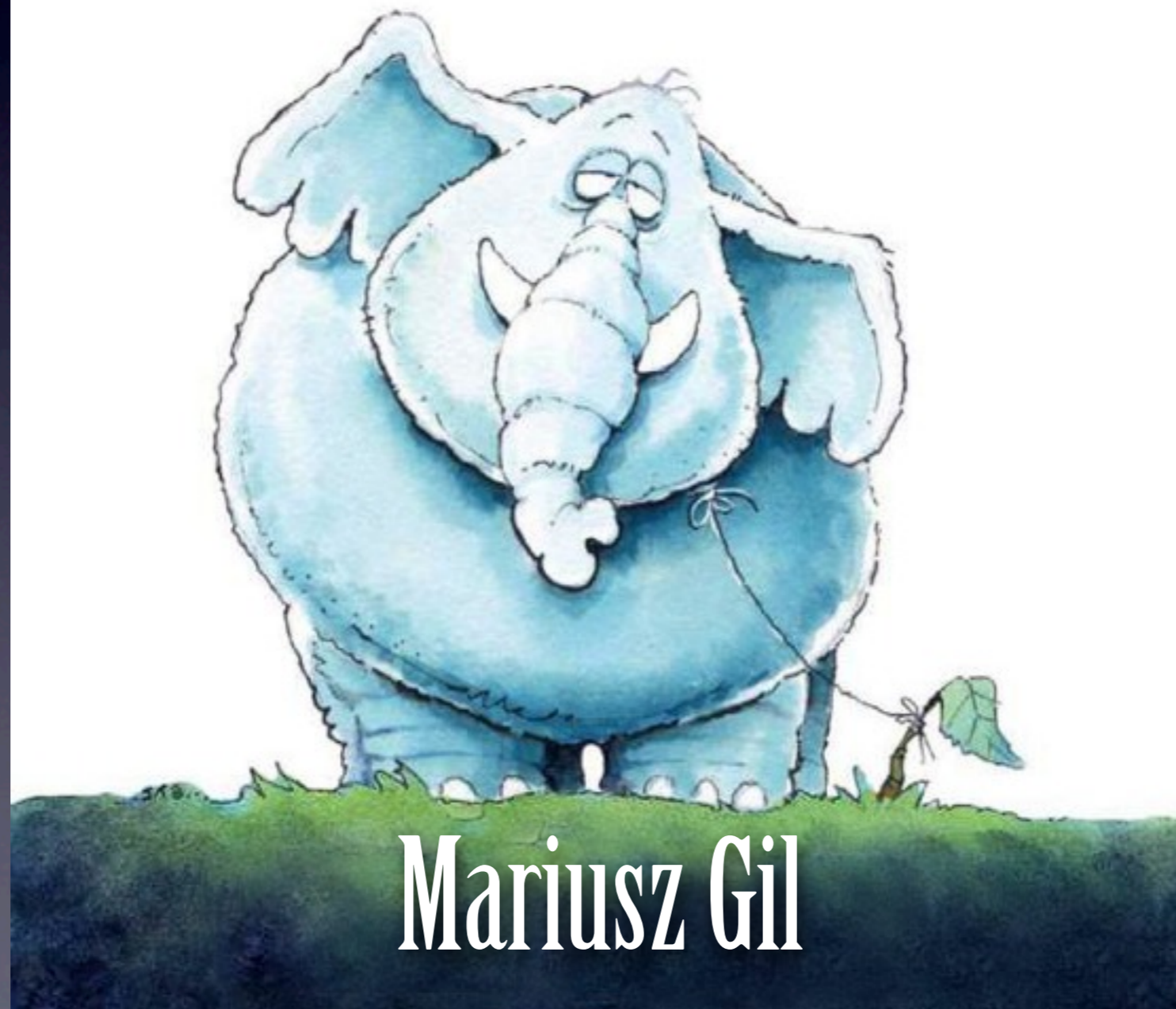
PHP kompilowane do .NET, <http://www.php-compiler.net/>



przedstawia

Dziękuję za uwagę

mariusz.gil@scalability.pl



Mariusz Gil