Navigation

- Aide mémoire Symfony2 (/fr/cms/22/aide-memoire-symfony2)
- Aide mémoire twig (/fr/cms/96/aide-memoire-twig--symfony2)
- Créer ses events (/fr/cms/178/creer-des-evenements-dans-symfony--listener--subscriber--dispatcher)
- Gedmo Sortable (/fr/cms/192/configurer-le-gedmo-sortable-des-doctrine-extensions-dans-symfony2)
- XHprof et Symfony (/fr/cms/198/installer-le-profiler-xhprof-avec-symfony-et-debian--analyse-de-performances-php)
- Aide mémoire composer (/fr/cms/160/aide-memoire-composer---symfony-2)
- Aide mémoire form (/fr/cms/131/aide-memoire-form---symfony2)
- Bundle sf2 extensible (/fr/cms/103/extensibilite-d_un-bundle-symfony2)
- Behat et Mink (/fr/cms/187/des-tests-fonctionnels-dans-symfony-2-grace-a-behat-et-mink)
  - Behat code coverage (/fr/cms/204/behat-php-code-coverage)
- phpunit et bundle sf2 (/fr/cms/106/tests-unitaires-d_un-bundle-symfony2)
  - Mise en place phpunit (/fr/cms/136/mise-en-place-de-phpunit-pour-un-bundle-symfony2)
  - Mise en place travis-ci (/fr/cms/137/mettre-en-place-travis-ci-pour-tester-un-bundle-symfony2)

Par Philippe Le Van (plv) (https://twitter.com/#!/plv) Dernière mise à jour : 18 septembre 2014

Installer le profiler Xhprof avec Symfony et Debian. Analyse de performances PHP

Introduction

Xhprof est un profiler codé par Facebook et donné à la communauté opensource en 2009. Il permet d'analyser assez finement l'exécution du PHP.

On peut notamment surveiller la mémoire prise par chaque fonction, le temps pris par chaque fonction, l'ensemble des appels de fonction (avec des jolis graphs).

Analyser les résultats peut prendre du temps, mais pour diagnostiquer des problèmes de performances, c'est un outil qui peut être précieux.

Ce tutoriel concerne l'environnement suivant (ça fonctionne probablement avec des versions antérieures) :

- PHP 5.5
- Debian Wheezy (dotdeb)
- Symfony 2.5

Principe de fonctionnement

Ce tutoriel décrit l'installation de 3 composants :

- une extension PHP sous PECL (package debian php5-xhprof)
- un bundle Symfony 2 qui permet d'activer xhprof sur un site symfony2 et qui enregistre les infos en base pour XH GUI (https://github.com/jonaswouters/XhprofBundle (https://github.com/jonaswouters/XhprofBundle))
- L'installation de l'outil XH GUI (https://github.com/preinheimer/xhprof (https://github.com/preinheimer/xhprof)) qui permet d'afficher les résultats de façon lisible.

Module PHP Pecl Xhprof

C'est une installation très classique

```
apt-get update
apt-get install php5-xhprof
```

Bundle symfony jns/xhprof-bundle

Ajouter le bundle dans composer.json

```
// ajouter les lignes suivantes dans votre composer.json

      "jns/xhprof-bundle": "1.0.*@dev",
      "facebook/xhprof": "dev-master@dev"
```

Activer le bundle dans AppKernel (en dev)

Contactez-nous

Kitpages
17 rue de la Frise
38000 Grenoble
tel : 04 76 69 26 65

Emai : contact@kitpages.fr (mailto:contact@kitpages.fr)


```
// initialiser le bundle dans app/Kernel
public function registerBundles()
{
    // ...

    if (in_array($this->getEnvironment(), array('dev', 'test'))) {
        // ...
        $bundles[] = new Jns\Bundle\XhprofBundle\JnsXhprofBundle();
    }
)
```

Ajouter les confs dans parameters.yml et config_dev.yml

```
# dans parameters.yml

    xhprof_enable: true
    xhprof_gui_url: http://xhgui-url.mondomain.fr/



# dans config_dev.yml

jns_xhprof:
    location_web:    %xhprof_gui_url%
    enabled:         %xhprof_enable%
    entity_class:    Kitpages\XhprofBundle\Entity\XhprofDetail
    enable_xhgui:    %xhprof_enable%
    command: "option"
    command_option_name: xhprof
```

Créer une entité doctrine qui enregistre les données

ATTENTION ! le nom de la table est "details". Il ne faut pas mettre autre chose. Ce nom est écrit en dur dans le code de XH GUI

```
<?php

namespace Kitpages\XhprofBundle\Entity;

use Jns\Bundle\XhprofBundle\Entity\XhprofDetail as BaseXhprofDetail;
use Doctrine\ORM\Mapping as ORM;

/**
 * @ORM\Entity
 * @ORM\Table(name="details")
 */
class XhprofDetail extends BaseXhprofDetail
{
    /**
     * @var integer $id
     *
     * @ORM\Column(name="id", type="string", unique=true, length=17,
nullable=false)
     * @ORM\Id
     */
    protected $id;
}
```

Mettre à jour la base, composer,...

Les dernières tâches classiques et le bundle est activé. Vous pouvez voir dans la table "details" les infos s'enregistrer à chaque page visitée.

```
# mettre à jour les vendors
composer update

# mettre à jour la base de données
app/console doctrine:schema:update

# vider le cache
app/console cache:clear
```

Installer XHGUI

Installer le site site xhgui

XHGUI est un site à part qui lit les données dans la table "details" créée par le projet symfony2 qu'on veut profiler.

Le code est disponible sur github : https://github.com/preinheimer/xhprof (https://github.com/preinheimer/xhprof)

Voilà quelques étapes pour installer le service :

```
# cloner le projet
cd /var/www
git clone https://github.com/preinheimer/xhprof.git
```

Installer graphviz

Pour voir des jolis graphiques il faut installer graphviz

```
# en root
apt-get update
apt-get install graphviz
```

Changer le config.php

copier le xhprof_lib/config.sample.php en xhprof_lib/config.php

et modifier les configurations suivantes suivant votre environnement :

```
<?php
$_xhprof['dbtype'] = 'mysql'; // Only relevant for PDO
$_xhprof['dbhost'] = '127.0.0.1';
$_xhprof['dbuser'] = 'root';
$_xhprof['dbpass'] = 'xxx';
$_xhprof['dbname'] = 'db_name';
$_xhprof['dbadapter'] = 'Pdo';
$_xhprof['servername'] = 'server-profiled.dev.com';
$_xhprof['namespace'] = 'http://xhgui-url.mondomain.fr/';
$_xhprof['url'] = 'http://xhgui-url.mondomain.fr/';

// [...]
$_xhprof['dot_binary']  = '/usr/bin/dot';
$_xhprof['dot_tempdir'] = '/tmp';
$_xhprof['dot_errfile'] = '/tmp/xh_dot.err';

// [...]
$_xhprof['display'] = true;
$_xhprof['doprofile'] = true;

//Control IPs allow you to specify which IPs will be permitted to control when
profiling is on or off within your application, and view the results via the
UI.
$controlIPs = array();
$controlIPs[] = "127.0.0.1";   // localhost, you'll want to add your own ip
here
$controlIPs[] = "::1";         // localhost IP v6
```

Créer le virtual host apache

Créer le fichier xhprof (ci-dessous) dans /etc/apache2/sites-available/

Créer un lien dans /etc/apache2/sites-enabled

relancer apache (service apache2 restart)

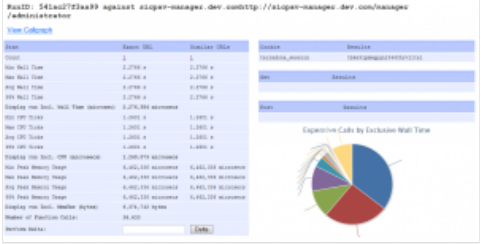
```
<VirtualHost *:80>
    ServerName xhgui-url.mondomain.fr
    ServerAdmin toto@hotmail.com
    DocumentRoot /var/www/xhprof
    addDefaultCharset UTF-8
</VirtualHost>
```

Utilisation

Consulter l'interface

Allez sur l'URL : http://xhgui-url.mondomain.fr/

Vous voyez normalement apparaitre une interface qui vous permet de voir le détail des consommations en resources (CPU, mémoire,...) de chaque fonction et de chaque page de votre site.



Callgraph

L'interface graphviz vous permet de consulter des graphs d'appel

Profiler une commande symfony2

Un petit point intéressant du bundle : on peut aussi profiler une commande symfony

```
# ajout de --xhprof pour profiler une commande
app/console acme:my:command --xhprof
```

Conclusion

Pour aller plus loin, je vous invite à aller voir la documentation des composants mentionnés dans ce tutoriel.

N'hésitez pas à ajouter dans les commentaires vos astuces / remarques / corrections,...

Commentaires

Ajouter un commentaire