



Альберт Степанцев @AlexLeonov  
Аварийный менеджер команд разработки

153,0 25,1  
карма рейтинг

вчера в 14:54

## Разработка → О том, как мы на PHP запускали настоящий MS Excel и что из этого вышло

Разработка под Windows\*, Программирование\*, ООП\*, Ненормальное программирование\*, PHP\*

Не секрет, что зачастую PHP-программистам приходится решать задачи, весьма далёкие от бытового представления о «веб-разработке». Развитие языка в последние годы привело к тому, что PHP всё чаще считают языком общего назначения, пригодным не только для сайтов, но и для других задач.

Одну из таких «других» задач мне с коллективом пришлось решать совсем недавно. Мы решили поделиться «картой граблей» с теми, кто, возможно, пойдет по этому же пути.

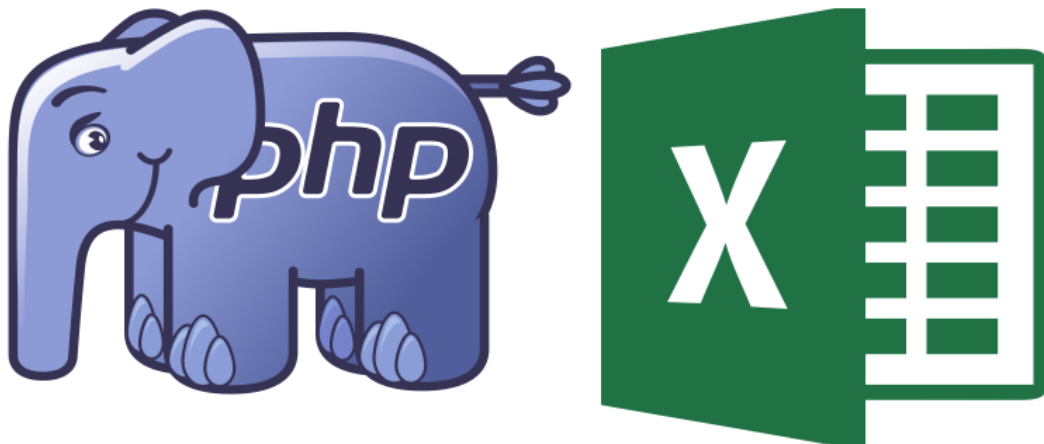
### Дано

- Многие наши партнёры (скажем прямо — это крупные банки) любят считать что-то в Excel. Причем «любят» — это очень нежно сказано. Сложнейшие скоринговые модели могут быть «запрограммированы» в Excel, в файле из сотни листов с десятками макросов
- Перевести «программы», написанные в Excel на какой-либо язык программирования — практически нереально. Это займет уйму времени, а проблема постоянного обновления и проверки корректности делает такую задачу и вовсе нерешаемой

### Требуется

- Основная информационная система нашей компании написана на PHP. Она содержит в себе как веб-интерфейсы, так и множество консольных сервисов и воркеров.
- С этими «программами» в Excel нужно как-то взаимодействовать из консольных приложений на PHP — передавать в них данные, обчислять, получать результаты

Некоторое время нам хватало возможностей популярной библиотеки PHPExcel. Но когда от бизнеса поступило очередное требование «нужно, чтобы работали макросы, и еще бы хорошо всё это сохранять в PDF», стало понятно, что выбранный путь — тупиковый. Нужно не парсить файлы xlsx, не имитировать просчёт, и даже не использовать Open Office, а научиться взаимодействовать с «настоящим» Microsoft Excel.



В результате недолгих изысканий было решено создать внутренний микро-веб-сервис, который умел бы принять данные, открыть указанный файл Excel, вставить в него принятые данные, просчитать результат и выдать его в качестве ответа клиенту. Заинтересованность в таком веб-сервисе выразили несколько внутренних проектов и работа закипела.

### Сервер под Windows? А почему бы нет!

Первым под удар попал отдел dev-ops. Им предстояло подготовить сервер для будущего сервиса. Дело было необычным, поскольку актуального опыта работы с Windows ни у кого нет...

В качестве серверной платформы был выбран Windows Server 2012 R2 standart. Нужно сразу отметить, что «из коробки» Windows совершенно не приспособлена к хостингу приложений на PHP. Требовалось доведение системы до нужного уровня.

Для начала был установлен PowerShellServer. Это позволило нам подключаться к windows-серверу по привычному всем протоколу ssh, не изобретая велосипедов. Поддерживается авторизация по ключам, работает rsync (это важно). Жаль, что в Personal Edition ограничение только на одно одновременное подключение, но для нас это некритично.

Nginx был установлен штатным образом. Взят со страницы [nginx.org/ru/download.html](http://nginx.org/ru/download.html) Имейте в виду — под windows есть существенные ограничения: только один рабочий процесс, который держит не более 1024 соединений. Впрочем, это опять же было некритично для внутреннего микро-сервиса.

PHP 7.0.9 взят с [windows.php.net/download](http://windows.php.net/download), установлен штатным образом.

Для упрощения перезапуска всего этого «добра» был написан несложный cmd-файл:

```
cd C:\nginx
taskkill /f /IM nginx.exe
taskkill /f /IM php-cgi.exe

rm C:\nginx\logs\*
start nginx
start -WindowStyle Hidden php\php-cgi -A "-b 0.0.0.0:9000 -c C:\server\php\php.ini"
```

Первоначальная настройка сервера закончилась успешным выводом страницы с phpinfo(). Однако это было еще только самое начало...

## Настраиваем сборку на Windows или Как наступить на все подводные камни?

Мы внутри компании используем Continuous Integration. Всегда. Для любого, сколь угодно малого проекта. Примерный план развертывания выглядит так:

- Сервер Teamcity следит за изменениями в нужных ветках репозитория (в данном конкретном случае workflow был упрощен до предела и ветка была фактически одна — master)
- Он же запускает сборку проекта при появлении изменений:
  - Содержимое репозитория с сервера TeamCity с помощью rsync доставляется на целевой сервер, во временную папку (это даёт нам возможность сэкономить на агентах TeamCity)
  - Там же, с помощью ssh, запускается билд-скрипт на phing, который и делает основную работу:
    - Переносит код в постоянное место
    - Устанавливает зависимости через composer
    - Раскладывает конфиги
    - Применяет миграции и так далее...
    - И, наконец, переключает симлинк current (это у нас wwwroot), на новую папку

Что потребовалось далее? ssh-сервер уже установлен, rsync выполняется корректно. Установим phing:

- Исполняемый файл (phing.phar) берем с [www.phing.info/trac/wiki/Users/Installation](http://www.phing.info/trac/wiki/Users/Installation)
- Аккуратный и красивый bat-ник можно взять на [www.phing.info/trac/browser/bin/phing.bat](http://www.phing.info/trac/browser/bin/phing.bat)
- Не забываем добавить путь до phing в PATH, чтобы получить system-wide команду

Git for Windows берем с [git-scm.com](http://git-scm.com), устанавливаем, проверяем корректную работу.

Точно по такой же схеме поступаем с composer, только bat-файл пишем сами и он будет значительно проще:

```
@echo off
if "%PHPBIN%" == "" set PHPBIN=C:\server\php\php.exe
"%PHPBIN%" "C:\nginx\php\composer.phar" %*
```

Вроде бы всё готово. Запускаем сборку... fail!

Причина 1. Нужно установить расширение php\_openssl.dll, иначе Phing не сможет работать с репозиториями через

SSL. Проблем не доставило.

Причина 2. Более серьезная. В нашем сценарии сборки используется техника переключения симлинка на папку со свежей сборкой на последнем шаге. Примерно так:

```
<symlink target="$(current.dir)" link="$(home.dir)/$(build.branch)/current" overwrite="true" />
```

В результате получается что-то вроде

```
symlink: "c:\server\domains\this.service\master\current" =>
"c:\server\domains\this.service\master\2016-04-01-12-34-56"
```

Оказалось, что создать символическую ссылку на NTFS — не проблема. Проблема ее удалить... Отчего-то операция удаления симлинка требует прав администратора, которых у обычного PHP нет и быть не может.

Нам помогла утилита junction ( [technet.microsoft.com/en-us/sysinternals/bb896768](http://technet.microsoft.com/en-us/sysinternals/bb896768) ). С ней вышеуказанный кусок сценария стал выглядеть примерно так:

```
<exec command="junction -d $(home.dir)/$(build.branch)/current" checkreturn="true" passthru="true" />
<symlink target="$(current.dir)" link="$(home.dir)/$(build.branch)/current" overwrite="true" />
```

Итак, всё встало на свои места, сборка заработала, как ей и положено. Настала пора писать код!

## COM-объекты в PHP

Надо отметить, что собственно код сервиса не доставил никаких проблем.

Как запустить приложение Microsoft Excel и загрузить в приложение существующий файл?

```
namespace App\Components;

class Excel
{
    protected $xls;

    public function __construct($filename = null)
    {
        $this->xls = new \COM("Excel.Application");
        // @todo: выключить, если не требуется видеть работу приложения
        $this->xls->Application->Visible = 1;
        $this->xls->DisplayAlerts = 0;
        if (empty($filename)) {
            $this->xls->Workbooks->Add();
        } else {
            $this->xls->Workbooks->Open($filename);
        }
        $this->xls->Workbooks[1]->Activate();
    }
}
```

Как закрыть приложение после окончания работы скрипта?

```
public function __destruct()
{
    $this->xls->Workbooks[1]->Close(false);
    $this->xls->Quit();
}
```

Установить значение ячейки или диапазона?

```
public function setValue($range, $value)
{

```

```
$this->xls->Range($range)->Value = iconv('UTF-8', 'Windows-1251', $value);  
}
```

Прочеть значение из ячейки или диапазона?

```
public function getValue($range)  
{  
    return iconv('Windows-1251', 'UTF-8', $this->xls->Range($range)->Value);  
}
```

Экспортировать книгу в PDF?

```
const FORMATS = [  
    'PDF' => 0  
];  
  
public function saveAs($filename, $format = self::FORMATS['PDF'])  
{  
    // Будь проклят тот день, когда разработчики MS-DOS придумали обратные слэши!  
    $this->xls->Workbooks[1]->ExportAsFixedFormat($format, str_replace('/', '\\', $filename));  
}
```

Что надо сделать, чтобы вся эта безумная магия заработала?

Добавить расширение `php_com_dotnet.dll`

## Вместо заключения

Довольно феерично наблюдать за сервером: при приходе запроса мгновенно открывается Excel, запускается экспорт и потом также мгновенно всё это хозяйство закрывается.

Удачи и не наступайте на те же грабли!

## Литература

1. [php.net/manual/en/book.com.php](http://php.net/manual/en/book.com.php)
2. [msdn.microsoft.com/ru-ru/library/wss56bz7.aspx](http://msdn.microsoft.com/ru-ru/library/wss56bz7.aspx)
3. [geektimes.ru/post/50878](http://geektimes.ru/post/50878)

◆ php, phpxcel, excel, ms excel, ms office, composer, phing