



# **Wdrażanie (deployment) projektów Symfony2 przy pomocy Capifony**

***Wydanie 1.0***

**Jacek Siciarek**

April 04 2014

<b>1</b>	<b>Spis treści</b>	<b>2</b>
1.1	Co to jest wdrożenie (deployment) . . . . .	2
1.2	Strategie wdrożeniowe . . . . .	3
1.3	Narzędzie do synchronizacji katalogów <code>rsync</code> . . . . .	3
1.4	Capistrano - Swiss Army Knife . . . . .	7
1.5	Sprawdzanie ustawień na serwerze lokalnym i zdalnym . . . . .	13
1.6	Inicjalizacja środowiska produkcyjnego aplikacji . . . . .	15
1.7	Source Control Management . . . . .	17
1.8	Zakres informacji zwracany przez <code>cap</code> . . . . .	19
1.9	Capifony - Capistrano dla Symfony . . . . .	24
1.10	Ustawienie odpowiednich praw dostępu do katalogów . . . . .	25
1.11	Włączanie i wyłączanie aplikacji . . . . .	26

Dokumentacja dostępna pod adresem <https://github.com/siciarek/capifony-4developers>

## 1.1 Co to jest wdrożenie (deployment)

**Wdrożeniem** nazywamy dowolną modyfikację **środowiska produkcyjnego** aplikacji.

### 1.1.1 Typy wdrożeń

#### Quick Fix

- (\*) zmiana (funkcjonalna/niefunkcjonalna)

#### Release

- (+) dodanie funkcjonalności
- (-) usunięcie funkcjonalności
- (\*) zmiana (funkcjonalna/niefunkcjonalna)

Aby porównać różne strategie wdrożenia posłużymy się poniższym zestawem parametrów.

### 1.1.2 Parametry wdrożenia

1. czas oczekiwania na zmianę
2. czas cofnięcia zmian
3. szacunkowa liczba kroków
4. informacja o zakresie zmian
5. bezpieczeństwo danych (safety)
6. bezpieczeństwo systemu (security)

## 1.2 Strategie wdrożeniowe

### 1.2.1 Strategia `null`

### 1.2.2 Strategia `null + copy`

### 1.2.3 Strategia `scm`

### 1.2.4 Strategia `scm + copy`

### 1.2.5 Strategia `scm + synchronise`

## 1.3 Narzędzie do synchronizacji katalogów `rsync`

Rsync jest aplikacją do synchronizacji katalogów na serwerze lokalnym i zdalnym, w obu kierunkach.

Strona projektu <http://rsync.samba.org>.

Szczegółowa dokumentacja <http://rsync.samba.org/ftp/rsync/rsync.html>.

### 1.3.1 Przykład sparametryzowanej komendy `rsync`

```
SOURCE=.
TARGET=dude@target.server.com:/var/www/target.server.com
```

```
rsync \
--itemize-changes \
--verbose \
--human-readable \
--omit-dir-times \
--times \
--perms \
--progress \
--stats \
--compress \
--recursive \
--links \
--delete \
--exclude-from="config/rsync_exclude.txt" \
--dry-run \
$SOURCE $TARGET
```

### 1.3.2 Przykładowy wynik działania komendy `rsync`

```
sending incremental file list
.L..t..... lib/doctrine/linked_schema.yml -> ../../data/linked_schema.yml
```

```
*deleting    web/apc.php
<f.st..... web/import.php
```

```
Number of files: 10047
Number of files transferred: 1
Total file size: 150.32M bytes
Total transferred file size: 2.76K bytes
Literal data: 0 bytes
Matched data: 0 bytes
File list size: 210.08K
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 211.97K
Total bytes received: 1.49K
```

```
sent 211.97K bytes  received 1.49K bytes  9.08K bytes/sec
total size is 150.32M  speedup is 704.24 (DRY RUN)
```

### 1.3.3 Jak rozumieć format opisu zmiany

```
*deleting    web/apc.php
<f+++++++ newfile.txt
.L..t..... lib/doctrine/linked_schema.yml -> ../../data/linked_schema.yml
<f.st..... web/import.php
```

Format opisu

YXcstpoguax

Gdzie

Y

- < - plik został wysłany na **serwer zdalny**.
- > - plik został pobrany ze **zdalnego serwera**.
- c - element został lokalnie utworzony np. nowy katalog, lub link symboliczny.
- h - oznacza, że dany element jest twardym linkiem (wymaga argumentu `--hard-links`).
- . - plik nie został zmieniony, ale zmieniły się inne parametry pliku.
- \* - prefiks dalszej informacji (np. "deleting").

## X

- f - plik
- d - katalog
- L - link symboliczny
- D - urządzenie
- S - plik specjalny (np. nazwane gniazdo (named socket) lub kolejka fifo).

Pozostałe znaki są literami opisującymi zmienione atrybuty pliku lub ":", jeżeli dany atrybut nie został zmieniony, poza trzema wyjątkami:

- nowoutworzony plik zamienia każdą z liter na znak "+"
- identyczny element zmienia kropki na spacje
- nieznaną atrybut zmieniany jest na znak "?" (to się może zdarzyć jeżeli na zdalnym serwerze jest starsza wersja `rsync`)

## cstpoguax

A c means either that a regular file has a different checksum (requires `--checksum`) or that a symlink, device, or special file has a changed value. Note that if you are sending files to an `rsync` prior to 3.0.1, this change flag will be present only for checksum-differing regular files.

A s means the size of a regular file is different and will be updated by the file transfer.

A t means the modification time is different and is being updated to the sender's value (requires `--times`). An alternate value of T means that the modification time will be set to the transfer time, which happens when a file/symlink/device is updated without `--times` and when a symlink is changed and the receiver can't set its time. (Note: when using an `rsync` 3.0.0 client, you might see the s flag combined with t instead of the proper T flag for this time-setting failure.)

A p means the permissions are different and are being updated to the sender's value (requires `--perms`).



An o means the owner is different and is being updated to the sender's value (requires --owner and super-user privileges).

A g means the group is different and is being updated to the sender's value (requires --group and the authority to set the group).

The u slot is reserved for future use.

The a means that the ACL information changed.

The x means that the extended attribute information changed.

Można użyć argumentu `--out-format` aby dostosować wyjście do swoich potrzeb.

## 1.4 Capistrano - Swiss Army Knife

Capistrano jest napisanym w języku ruby systemem zdalnego zarządzania zasobami serwerów, przeznaczonym dla dowolnego języka programowania. Może służyć do wdrażania lub zdalnego wykonywania komend na wielu serwerach jednocześnie.

System powstał z myślą o wdrażaniu projektów ruby on rails, jednak jego funkcjonalność jest na tyle uniwersalna, że może służyć właściwie dla każdego języka programowania i każdego typu projektu.

### 1.4.1 Instalacja

Capistrano jest dostępny jako pakiet gem dla ruby.

```
$ gem -v  
1.8.23
```

Najniższa wymagana wersja to 1.3.x, jeżeli nie nie posiadasz takiej, należy przeprowadzić aktualizację a następnie wykonać komendę:

```
$ gem install capistrano
```

Po pomyślnie wykonanej instalacji powinno być możliwe wykonanie poniższej komendy:

```
$ cap --help
Usage: cap [options] action ...
  -d, --debug                Prompts before each remote command execution.
  -e, --explain TASK         Displays help (if available) for the task.
  -F, --default-config       Always use default config, even with -f.
  -f, --file FILE            A recipe file to load. May be given more than once.
  -H, --long-help            Explain these options and environment variables.
  -h, --help                 Display this help message.
  -l [STDERR|STDOUT|file],  Choose logger method. STDERR used by default.
    --logger
  -n, --dry-run              Prints out commands without running them.
  -p, --password             Immediately prompt for the password.
  -q, --quiet                Make the output as quiet as possible.
  -r, --preserve-roles       Preserve task roles
  -S, --set-before NAME=VALUE Set a variable before the recipes are loaded.
  -s, --set NAME=VALUE       Set a variable after the recipes are loaded.
  -T, --tasks [PATTERN]     List all tasks (matching optional PATTERN) in the loaded recipe files.
  -t, --tool                 Abbreviates the output of -T for tool integration.
  -V, --version              Display the Capistrano version, and exit.
  -v, --verbose              Be more verbose. May be given more than once.
  -X, --skip-system-config   Don't load the system config file (capistrano.conf)
  -x, --skip-user-config     Don't load the user config file (.caprc)
```

### 1.4.2 Skrypt cap

Skrypt cap jest narzędziem umożliwiającym połączenie ze zdalnymi serwerami i wykonywanie na nich komend.

```
$ cat `which cap`
#!/usr/bin/ruby1.9.1
#
# This file was generated by RubyGems.
```

```
#
# The application 'capistrano' is installed as part of a gem, and
# this file is here to facilitate running it.
#

require 'rubygems'

version = ">= 0"

if ARGV.first
  str = ARGV.first
  str = str.dup.force_encoding("BINARY") if str.respond_to? :force_encoding
  if str =~ /\A_(.*)_\Z/
    version = $1
    ARGV.shift
  end
end

gem 'capistrano', version
load Gem.bin_path('capistrano', 'cap', version)
```

Lista dostępnych komend po wydaniu polecenia

```
$ cap -T
$ cap -vT
```

Skryptu można używać z linii komend podając odpowiednie parametry:

```
$ cap invoke HOSTS="new.ses-control.com,platform.ses-support.com" COMMAND="date"

* 2014-04-03 10:36:44 executing 'invoke'
* executing multiple commands in parallel
  -> "else" :: "date"
```

```
-> "else" :: "date"
servers: ["new.ses-control.com", "platform.ses-support.com"]
[new.ses-control.com] executing command
[platform.ses-support.com] executing command
** [out :: new.ses-control.com] czw, 3 kwi 2014, 10:36:44 CEST
** [out :: platform.ses-support.com] czw, 3 kwi 2014, 10:36:44 CEST
command finished in 44ms
```

Jeżeli do komendy wymagane jest sudo przekazujemy dodatkowy parametr:

```
$ cap invoke HOSTS="new.ses-control.com,platform.ses-support.com" COMMAND="date" SUDO=1
```

Tu niestety może pojawić się błąd:

```
* 2014-04-03 10:51:12 executing 'invoke'
* executing multiple commands in parallel
-> "else" :: "sudo -p 'sudo password: ' date"
-> "else" :: "sudo -p 'sudo password: ' date"
servers: ["new.ses-control.com", "platform.ses-support.com"]
[platform.ses-support.com] executing command
[new.ses-control.com] executing command
*** [err :: platform.ses-support.com] sudo
*** [err :: platform.ses-support.com] :
*** [err :: platform.ses-support.com] no tty present and no askpass program specified
*** [err :: platform.ses-support.com]
*** [err :: new.ses-control.com] sudo
*** [err :: new.ses-control.com] :
*** [err :: new.ses-control.com] no tty present and no askpass program specified
*** [err :: new.ses-control.com]
command finished in 87ms
failed: "sh -c 'sudo -p '\\\\''sudo password: '\\\\'' date'" on new.ses-control.com,platform.ses-support.com
```

Aby umożliwić wykonywanie komend sudo musimy utworzyć plik konfiguracyjny i dodać linijkę:

```
default_run_options[:pty] = true
```

Poniżej domyślne nazwy plików konfiguracyjnych dla cap.

/etc/capistrano.conf - ustawienie globalne serwera (pomijany parametrem --skip-system-config)

~/.caprc file - ustawienie dla konta użytkownika (pomijany parametrem --skip-user-config)

./Capfile - ustawienie dla projektu

deploy.rb - alternatywne ustawienie dla projektu

### 1.4.3 Automatyczne tworzenie pliku konfiguracji

Plik konfiguracyjny dla projektu można utworzyć automatycznie:

```
$ cd /path/to/your/project
$ capify .
```

```
[add] writing '/path/to/your/project/Capfile'
[add] making directory '/path/to/your/project/config'
[add] writing '/path/to/your/project/config/deploy.rb'
[done] capified!
```

Zawartość katalogu /path/to/your/project:

```
$ ls -R /path/to/your/project
/path/to/your/project:
Capfile  config
```

```
/path/to/your/project/config:
deploy.rb
```

Zawartość /path/to/your/project/Capfile

```
load 'deploy'
# Uncomment if you are using Rails' asset pipeline
# load 'deploy/assets'
load 'config/deploy' # remove this line to skip loading any of the default tasks
```

Zawartość /path/to/your/project/deploy.rb

```
set :application, "set your application name here"
set :repository, "set your repository location here"

# set :scm, :git # You can set :scm explicitly or Capistrano will make an intelligent guess based on known version control systems
# Or: 'accurev', 'bazaar', 'cvs', 'darcs', 'git', 'mercurial', 'perforce', 'subversion' or 'none'

role :web, "your web-server here" # Your HTTP server, Apache/etc
role :app, "your app-server here" # This may be the same as your 'Web' server
role :db, "your primary db-server here", :primary => true # This is where Rails migrations will run
role :db, "your slave db-server here"

# if you want to clean up old releases on each deploy uncomment this:
# after "deploy:restart", "deploy:cleanup"

# if you're still using the script/reaper helper you will need
# these http://github.com/rails/irs_process_scripts

# If you are using Passenger mod_rails uncomment this:
# namespace :deploy do
#   task :start do ; end
#   task :stop do ; end
#   task :restart, :roles => :app, :except => { :no_release => true } do
#     run "#{try_sudo} touch #{File.join(current_path, 'tmp', 'restart.txt')}"
#   end
# end
```

### 1.4.4 Przykładowa konfiguracja w pliku Capfile

```
load 'deploy' if respond_to?(:namespace) # cap2 differentiator

# RUN OPTIONS:

default_run_options[:pty] = true

# SSH SETTINGS:

ssh_options[:forward_agent] = true

# SERVERS

role :app,
  'new.ses-control.com',
  'platform.ses-support.com'
```

Powyższa konfiguracja umożliwia wykonanie komend bez podawania listy serwerów.

```
$ cap invoke COMMAND="date"
$ cap invoke COMMAND="date" SUDO=1
```

## 1.5 Sprawdzanie ustawień na serwerze lokalnym i zdalnym

Aby umożliwić sprawdzenie ustawień na lokalnym i zdalnym serwerze Capistrano udostępnia polecenie

```
$ cap deploy:check
```

W wersji podstawowej sprawdza obecność i prawa dostępu do katalogów utworzonych poleceniem

```
$ cap deploy:setup
```

Dostępność komendy `tar` na zdalnym serwerze oraz programu do kontroli wersji na lokalnym.

Wyświetlane są tylko efekty działania komendy sprawdzającej na zdalnym serwerze, chyba, że pojawią się błędy, wtedy wyświetlane są w postaci komunikatu np.:

```
The following dependencies failed. Please check them and try again:
--> 'git' could not be found in the path on the local host
--> 'setfacl' could not be found in the path (ekk.sescom.pl)
```

Jeżeli sprawdzanie zakończy się sukcesem pojawi się komunikat:

```
You appear to have all necessary dependencies installed
```

Aby umożliwić dodawanie sprawdzania zasobów Capistrano udostępnia metodę `depend` Przyjmującą 3 argumenty

**Pierwszy parametr** określa miejsce sprawdzania, może przyjmować poniższe wartości:

- `:local` jeżeli chcemy sprawdzać serwer z którego przeprowadzamy wdrożenie
- `:remote` jeżeli chcemy sprawdzić zdalne serwery.

**Drugi parametr** określa typ sprawdzanego zasobu i może przyjmować poniższe wartości:

- `:directory` sprawdza czy istnieje katalog.
- `:file` sprawdza czy istnieje plik.
- `:writable` sprawdza czy katalog lub plik posiada prawa do zapisu.
- `:command` sprawdza czy komenda jest dostępna.
- `:deb` sprawdza czy pakiet jest zainstalowany.

**Trzeci parametr** to nazwa sprawdzanego zasobu w postaci łańcucha znaków.



### 1.5.1 Przykłady użycia

```
depend :local, :command, "convert"
depend :remote, :command, "setfacl"
```

## 1.6 Inicjalizacja środowiska produkcyjnego aplikacji

Przykładowa konfiguracja w Capfile

```
set :application, "myapp"
set :user, "dude"
set :domain, "target.server.com"
set :repository, "git@myscm.server.net:/home/git/repos/#{application}.git"
set :deploy_to, "/var/www/#{domain}"

role :app, :domain, :primary => true # This may be the same as your ``Web`` server
role :web, :domain # Your HTTP server, Apache/etc
```

### 1.6.1 Przygotowanie środowiska produkcyjnego

```
$ cap deploy:setup
```

```
* 2014-04-04 12:54:57 executing `deploy:setup`
* executing multiple commands in parallel
-> "else" :: "sudo -p 'sudo password: ' mkdir -p /var/www/target.server.com /var/www/target.server.com/releases /var
-> "else" :: "sudo -p 'sudo password: ' mkdir -p /var/www/target.server.com /var/www/target.server.com/releases /var
-> "else" :: "sudo -p 'sudo password: ' mkdir -p /var/www/target.server.com /var/www/target.server.com/releases /var
-> "else" :: "sudo -p 'sudo password: ' mkdir -p /var/www/target.server.com /var/www/target.server.com/releases /var
servers: ["new.ses-control.com", "platform.ses-support.com", "integration.sescom.pl", "hm.ses-control.com"]
```

```
[integration.sescom.pl] executing command
[hm.ses-control.com] executing command
[platform.ses-support.com] executing command
[new.ses-control.com] executing command
command finished in 130ms
* executing multiple commands in parallel
-> "else" :: "sudo -p 'sudo password: ' chmod g+w /var/www/target.server.com /var/www/target.server.com/releases /va
-> "else" :: "sudo -p 'sudo password: ' chmod g+w /var/www/target.server.com /var/www/target.server.com/releases /va
-> "else" :: "sudo -p 'sudo password: ' chmod g+w /var/www/target.server.com /var/www/target.server.com/releases /va
-> "else" :: "sudo -p 'sudo password: ' chmod g+w /var/www/target.server.com /var/www/target.server.com/releases /va
servers: ["new.ses-control.com", "platform.ses-support.com", "integration.sescom.pl", "hm.ses-control.com"]
[integration.sescom.pl] executing command
[hm.ses-control.com] executing command
[platform.ses-support.com] executing command
[new.ses-control.com] executing command
command finished in 98ms
```

Jeżeli wszystkie prawa dostępu będą ustawione prawidłowo na **serwerze produkcyjnym** zostanie utworzona poniższa struktura katalogów:

```
/var/www/target.server.com
|-- releases
`-- shared
```

### 1.6.2 Sprawdzenie konfiguracji serwera produkcyjnego

```
cap deploy:check
```

### 1.6.3 Przykład zawartości katalogu aplikacji po kolejnym wdrożeniu

```
`-- /var/www/target.server.com
|-- current -> /var/www/target.server.com/releases/20100512131539
|-- releases
|   |-- 20100512131539
|   |-- 20100509150741
|   |-- 20100509145325
`-- shared
    |-- web
    |   |-- uploads
    |-- logs
    |-- vendors
    |-- config
    |   |-- app
    |       |-- parameters.yml
```

## 1.7 Source Control Management

Przykładowa konfiguracja w Capfile

```
set :application, "myapp"
set :user, "dude"
set :domain, "target.server.com"
set :repository, "git@myscm.server.net:/home/git/repos/#{application}.git"
set :deploy_to, "/var/www/#{domain}"

set :scm, :git
set :deploy_via, :copy
```

### 1.7.1 Wartości :scm

**:none** System kontroli wersji nie jest stosowany

**:git** GIT <http://git-scm.com>

**:accurev** AccuRev <http://www.accurev.com>

**:bazaar** Bazaar <http://bazaar.canonical.com>

**:cvs** CVS <http://www.nongnu.org/cvs/>

**:darcs** Darcs <http://darcs.net>

**:subversion** SVN <http://subversion.tigris.org>, <http://subversion.apache.org>

**:mercurial** Mercurial <http://mercurial.selenic.com>

**:perforce** PERFORCE <http://www.perforce.com>

### 1.7.2 Wartości :deploy\_via

**:copy** Na **serwerze źródłowym** wykonuje eksport wersji do katalogu /tmp, następnie kopiuje jego zawartość na **serwer produkcyjny**. Preferowane użycie, nie zadziała jak serwer źródłowy i produkcyjny jest tą samą maszyną.

**:checkout** Wykonuje operację checkout na **serwerze produkcyjnym**, nie zalecany, ze względu na możliwość nieautoryzowanej modyfikacji zawartości repozytorium.

**:export** Wykonuje operację export na **serwerze produkcyjnym**.

**:remote\_cache** Wykonuje git pull (lub odpowiednik tej komendy w innych SCM) w katalogu shared/cached\_copy, zamiast zaciągania całego repozytorium. Ma znaczenie jeżeli zależy nam na przyspieszeniu procesu wdrożenia. Jeżeli w czasie wdrożenia pojawią się problemy należy usunąć katalog shared/cached\_copy.

**:rsync\_with\_remote\_cache** W tym przypadku rsync utworzy katalog na **serwerze produkcyjnym** i będzie go synchronizował ze zmianami w repozytorium. Wymaga instalacji dodatkowego gema.

## 1.8 Zakres informacji zwracany przez cap

Capistrano umożliwia wyświetlanie informacji o wykonywanych komendach w różnych zakresach.

### 1.8.1 Typy zakresu

```
logger.level = Logger::IMPORTANT # -->
logger.level = Logger::INFO      # **
logger.level = Logger::DEBUG     # *
logger.level = Logger::TRACE     #
logger.level = Logger::MAX_LEVEL # Logger::TRACE = Logger::MAX_LEVEL
```

Fragment kodu `logger.rb`

```
module Capistrano
  class Logger #:nodoc:
    attr_accessor :level
    attr_reader   :device

    IMPORTANT = 0
    INFO      = 1
    DEBUG     = 2
    TRACE     = 3

    MAX_LEVEL = 3
  end
end
```

## 1.8.2 Przykładowy wynik ustawień

```
logger.level = Logger::IMPORTANT
```

```
--> Updating code base with copy strategy
--> Creating cache directory.....V
--> Creating symlinks for shared directories.....V
--> Creating symlinks for shared files.....V
--> Normalizing asset timestamps.....V
--> Downloading Composer.....V
```

```
logger.level = Logger::INFO`
```

```
** sftp upload parameters.yml -> /d0/www/platform-integration/shared/app/config/parameters.yml
** transaction: start
--> Updating code base with copy strategy
** sftp upload /tmp/20140319145926.tar.gz -> /tmp/20140319145926.tar.gz
--> Creating cache directory
--> Creating symlinks for shared directories
--> Creating symlinks for shared files
--> Normalizing asset timestamps
--> Downloading Composer
** [out :: myspec.pl] #!/usr/bin/env php
** [out :: myspec.pl] Some settings on your machine may cause stability issues with Composer.
** [out :: myspec.pl] If you encounter issues, try to change the following:
** [out :: myspec.pl]
** [out :: myspec.pl] Your PHP (5.3.3-7+squeezel4) is quite old, upgrading to PHP 5.3.4 or higher is recommended.
** [out :: myspec.pl] Composer works with 5.3.2+ for most people, but there might be edge case issues.
** [out :: myspec.pl]
** [out :: myspec.pl] Downloading...
** [out :: myspec.pl]
** [out :: myspec.pl] Composer successfully installed to: /d0/www/platform-integration/releases/20140319145926/compos
** [out :: myspec.pl]
```

```
** [out :: myspec.pl] Use it: php composer.phar
--> Updating Composer dependencies
** [out :: myspec.pl] Loading composer repositories with package information
** [out :: myspec.pl] Updating dependencies

logger.level = Logger::DEBUG

* 2014-03-19 16:00:41 executing `deploy`
* 2014-03-19 16:00:41 executing `upload_parameters`
* executing "mkdir -p /d0/www/platform-integration/shared/app/config"
** sftp upload parameters.yml -> /d0/www/platform-integration/shared/app/config/parameters.yml
* sftp upload complete
* 2014-03-19 16:00:41 executing `deploy:update`
** transaction: start
* 2014-03-19 16:00:41 executing `deploy:update_code`
--> Updating code base with copy strategy
* getting (via checkout) revision cc301318ff9b560edd494a57db6c66ea95c878c7 to /tmp/20140319150041
* processing exclusions...
* Compressing /tmp/20140319150041 to /tmp/20140319150041.tar.gz
** sftp upload /tmp/20140319150041.tar.gz -> /tmp/20140319150041.tar.gz
* sftp upload complete
* executing "cd /d0/www/platform-integration/releases && tar xzf /tmp/20140319150041.tar.gz && rm /tmp/20140319150041"
* 2014-03-19 16:00:44 executing `deploy:finalize_update`
* executing "chmod -R g+w /d0/www/platform-integration/releases/20140319150041"
--> Creating cache directory
* executing "sh -c 'if [ -d /d0/www/platform-integration/releases/20140319150041/app/cache ] ; then rm -rf /d0/www/p"
* executing "sh -c 'mkdir -p /d0/www/platform-integration/releases/20140319150041/app/cache && chmod -R 0777 /d0/www"
* executing "chmod -R g+w /d0/www/platform-integration/releases/20140319150041/app/cache"
* 2014-03-19 16:00:44 executing `deploy:share_childs`
--> Creating symlinks for shared directories
* executing "mkdir -p /d0/www/platform-integration/shared/app/logs"
* executing "sh -c 'if [ -d /d0/www/platform-integration/releases/20140319150041/app/logs ] ; then rm -rf /d0/www/pl"
```

```
* executing "ln -nfs /d0/www/platform-integration/shared/app/logs /d0/www/platform-integration/releases/20140319150041/app/logs"
* executing "mkdir -p /d0/www/platform-integration/shared/web/uploads"
* executing "sh -c 'if [ -d /d0/www/platform-integration/releases/20140319150041/web/uploads ] ; then rm -rf /d0/www/platform-integration/releases/20140319150041/web/uploads ; fi'"
* executing "ln -nfs /d0/www/platform-integration/shared/web/uploads /d0/www/platform-integration/releases/20140319150041/web/uploads"
* executing "mkdir -p /d0/www/platform-integration/shared/vendor"
* executing "sh -c 'if [ -d /d0/www/platform-integration/releases/20140319150041/vendor ] ; then rm -rf /d0/www/platform-integration/releases/20140319150041/vendor ; fi'"
* executing "ln -nfs /d0/www/platform-integration/shared/vendor /d0/www/platform-integration/releases/20140319150041/vendor"
--> Creating symlinks for shared files
* executing "mkdir -p /d0/www/platform-integration/shared/app/config"
* executing "touch /d0/www/platform-integration/shared/app/config/parameters.yml"
* executing "ln -nfs /d0/www/platform-integration/shared/app/config/parameters.yml /d0/www/platform-integration/releases/20140319150041/app/config/parameters.yml"
--> Normalizing asset timestamps
* executing "find /d0/www/platform-integration/releases/20140319150041/web/css /d0/www/platform-integration/releases/20140319150041/web/js -type f -exec touch {} \;"
* 2014-03-19 16:00:45 executing `symfony:composer:update`
* 2014-03-19 16:00:45 executing `symfony:composer:get`
* executing "if [ -e /d0/www/platform-integration/releases/20140319150041/composer.phar ] ; then echo 'true'; fi"
--> Downloading Composer
* executing "sh -c 'cd /d0/www/platform-integration/releases/20140319150041 && curl -s http://getcomposer.org/installer.php | php'"
** [out :: myspec.pl] #!/usr/bin/env php
** [out :: myspec.pl] Some settings on your machine may cause stability issues with Composer.
** [out :: myspec.pl] If you encounter issues, try to change the following:
** [out :: myspec.pl]
** [out :: myspec.pl] Your PHP (5.3.3-7+squeeze14) is quite old, upgrading to PHP 5.3.4 or higher is recommended.
** [out :: myspec.pl] Composer works with 5.3.2+ for most people, but there might be edge case issues.
** [out :: myspec.pl]
** [out :: myspec.pl] Downloading...
** [out :: myspec.pl]
** [out :: myspec.pl] Composer successfully installed to: /d0/www/platform-integration/releases/20140319150041/composer.phar

logger.level = Logger::TRACE
logger.level = Logger::MAX_LEVEL # Logger::TRACE jest równy Logger::MAX_LEVEL
```



```
* 2014-03-19 16:02:46 executing `deploy`
  triggering before callbacks for `deploy`
* 2014-03-19 16:02:46 executing `upload_parameters`
* executing "mkdir -p /d0/www/platform-integration/shared/app/config"
  servers: ["myspec.pl"]
  [myspec.pl] executing command
  command finished in 24ms
  servers: ["myspec.pl"]
** sftp upload parameters.yml -> /d0/www/platform-integration/shared/app/config/parameters.yml
  [myspec.pl] /d0/www/platform-integration/shared/app/config/parameters.yml
  [myspec.pl] done
* sftp upload complete
* 2014-03-19 16:02:46 executing `deploy:update`
** transaction: start
* 2014-03-19 16:02:46 executing `deploy:update_code`
  triggering before callbacks for `deploy:update_code`
--> Updating code base with copy strategy
  executing locally: "git ls-remote git@ses-support.net:/home/git/repos/platform-integration.git HEAD"
  command finished in 91ms
* getting (via checkout) revision cc301318ff9b560edd494a57db6c66ea95c878c7 to /tmp/20140319150246
  executing locally: git clone -q git@ses-support.net:/home/git/repos/platform-integration.git /tmp/20140319150246
  command finished in 2742ms
* processing exclusions...
* Compressing /tmp/20140319150246 to /tmp/20140319150246.tar.gz
  executing locally: tar czf 20140319150246.tar.gz 20140319150246
  command finished in 38ms
  servers: ["myspec.pl"]
** sftp upload /tmp/20140319150246.tar.gz -> /tmp/20140319150246.tar.gz
  [myspec.pl] /tmp/20140319150246.tar.gz
  [myspec.pl] done
* sftp upload complete
* executing "cd /d0/www/platform-integration/releases && tar xzf /tmp/20140319150246.tar.gz && rm /tmp/20140319150246"
```

```
servers: ["myspec.pl"]
[myspec.pl] executing command
command finished in 52ms
* 2014-03-19 16:02:49 executing `deploy:finalize_update`
* executing "chmod -R g+w /d0/www/platform-integration/releases/20140319150246"
servers: ["myspec.pl"]
[myspec.pl] executing command
command finished in 23ms
```

### 1.8.3 Zapis loga do pliku

```
log_file = "deployment.log"
self.logger = Logger.new(:output => log_file)

logger.level = Logger::DEBUG
```

## 1.9 Capifony - Capistrano dla Symfony

Capifony jest napisanym w języku ruby systemem zdalnego zarządzania zasobami serwerów, dedykowanego do wdrożeń projektów Symfony.

### 1.9.1 Instalacja

Capifony jest dostępny jako pakiet gem dla ruby.

```
$ gem -v
1.8.23
```

Najniższa wymagana wersja to 1.3.x, jeżeli nie nie posiadasz takiej, należy przeprowadzić aktualizację a następnie wykonać komendę:

```
$ gem install capifony
```

Po pomyślnie wykonanej instalacji powinno być możliwe wykonanie poniższej komendy:

```
$ capifony --help
Usage: capifony [path]
    -h, --help                Displays this help info
    -v, --version              Capify specific symfony version (1|2)
    -s, --symfony VERSION     Specify app name (folder) to capify
    -p, --app NAME
```

### 1.10 Ustawienie odpowiednich praw dostępu do katalogów

```
set :writable_dirs,      [ "app/cache", "app/logs" ]
set :webserver_user,     "www-data"
set :permission_method,  :acl
set :use_set_permissions, true
```

Ustawienie ścieżek do których inny użytkownik ma mieć prawo dostępu typu rw.

```
set :writable_dirs,      [ "app/cache", "app/logs", "web/uploads" ]
```

Podanie użytkownika, który ma mieć prawa dostępu typu rw do powyższych katalogów.

```
set :webserver_user,     "www-data"
```

Podanie sposobu w jaki mają być ustawiane prawa dostępu.

```
set :permission_method,  :acl
```

Dostępne wartości parametru:

- :acl - korzysta z setfacl

- `:chown` - wymaga ustawienia `set use_sudo:, true`
- `:chmod` - korzysta z `chmod +a`

Preferowana jest wartość `:acl`

Flaga, czy należy ustawiać prawa dostępu, może być czasowo wyłączane bez potrzeby usuwania całej powyższej konfiguracji.

```
set :use_set_permissions, true
```

## 1.11 Włączanie i wyłączanie aplikacji

### 1.11.1 Podstawowe komendy

```
cap deploy:web:disable  
cap deploy:web:enable
```

Zmiana wyświetlanego powodu wyłączenia `REASON` i czasu ponownego uruchomienia `UNTIL`

```
cap deploy:web:disable REASON="database migration" UNTIL="2014-04-08"
```

### 1.11.2 Szablon strony z powiadomieniem

Szablon domyślny `/usr/lib/ruby/vendor_ruby/capistrano/recipes/deploy/templates/maintenance.rhtml`

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
  
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">  
  
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
<title>System down for maintenance</title>

<style type="text/css">
  div.outer {
    position: absolute;
    left: 50%;
    top: 50%;
    width: 500px;
    height: 300px;
    margin-left: -260px;
    margin-top: -150px;
  }

  .DialogBody {
    margin: 0;
    padding: 10px;
    text-align: left;
    border: 1px solid #ccc;
    border-right: 1px solid #999;
    border-bottom: 1px solid #999;
    background-color: #fff;
  }

  body { background-color: #fff; }
</style>
</head>

<body>

  <div class="outer">
    <div class="DialogBody" style="text-align: center;">
```

```
<div style="text-align: center; width: 200px; margin: 0 auto;">
  <p style="color: red; font-size: 16px; line-height: 20px;">
    The system is down for <%= reason ? reason : "maintenance" %>
    as of <%= Time.now.strftime("%H:%M %Z") %>.
  </p>
  <p style="color: #666;">
    It'll be back <%= deadline ? deadline : "shortly" %>.
  </p>
</div>
</div>
</div>
```

</body>

</html>

Szablon zmodyfikowany ./maintenance.pl.rhtml

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl">

<head>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
  <title>Aplikacja jest czasowo niedostępna</title>

  <style type="text/css">
    div.outer {
      position: absolute;
      left: 50%;
      top: 50%;
      width: 500px;
      height: 300px;
```

```
    margin-left: -260px;
    margin-top: -150px;
  }

  .DialogBody {
    margin: 0;
    padding: 10px;
    text-align: left;
    border: 1px solid #ccc;
    border-right: 1px solid #999;
    border-bottom: 1px solid #999;
    background-color: #fff;
  }

  body { background-color: #fff; }
</style>
</head>

<body>

  <div class="outer">
    <div class="DialogBody" style="text-align: center;">
      <div style="text-align: center; width: 200px; margin: 0 auto;">
        <p style="color: red; font-size: 16px; line-height: 20px;">
          Od <%= Time.now.strftime("%Y-%m-%d %H:%M") %>
          aplikacja jest czasowo niedostępna z powodu <%= reason ? reason : "konserwacji" %>.
        </p>
        <p style="color: #666;">
          Zostanie ponownie uruchomiona <%= deadline ? deadline : "niebawem" %>.
        </p>
      </div>
    </div>
  </div>
```

```
</div>

</body>
</html>
```

Aby capistrano korzystał ze zmodyfikowanego szablonu należy dodać do Capfile

```
set :maintenance_template_path, "maintenance.pl.rhtml"
```

```
cap deploy:web:disable REASON="migracji baz danych na nowy serwer" UNTIL="20 lutego 2014 9:30"
```

### 1.11.3 Konfiguracja serwerów www

Apache

```
RewriteCond %{DOCUMENT_ROOT}/maintenance.html -f
RewriteRule .? maintenance.html [L]
```

NginX

```
if (-f /maintenance.html)
{
    rewrite .? maintenance.html last;
}
```