Estrutura de Diretório e CLI

ESTRUTURA DE DIRETÓRIO DEFAULT

```
myproject (symfony root)
  apps
    myapp
      config
        app.yml
        cache.yml
        databases.yml
        factories.yml
        filters.yml
        i18n.yml
        logging.yml
        routing.yml
        security.yml
        settings.yml
        view.yml
      i18n
      lib
      modules
        mymodule
          actions
            actions.class.php
            components.class.php
          i18n
          config
          lib
          templates
            indexSuccess.php
          validate
      templates
  batch
  cache
    myapp
    tmp
  config
    schema.yml (schema.xml)
    propel.ini
    php.yml
  data
    fixtures
    model
    sql
  doc
  lib
    model
      map
      om
  log
    myapp_dev.log
    myapp_prod.log
  test
    myapp
  web
    css
    images
    js
    uploads
      assets
    index.php
    myapp_dev.php
```

COMMAND LINE INTERFACE (CLI)

`$ symfony -T`
Exibe uma lista completa das operações de administração disponíveis

`$ symfony -V`
Exibe a versão do symfony instalada

`$ symfony clear-cache <application_name> [template|config]`
Limpa as informações do cache (atalho: cc)

`$ symfony init-project <project_name>`
Inicializa um projeto e gera os arquivos e diretórios básicos necessários para execução

`$ symfony init-app <application_name>`
Inicializa uma aplicação. Alguns arquivos PHP, correspondentes aos fronts controllers de cada ambiente default, são criados no diretório web (no root do projeto):
index.php (prod) e myapp_dev.php (dev)

`$ symfony init-module <application_name> <module_name>`
Inicializa um módulo. Após o comando, o novo módulo está pronto para ser utilizado:
http://myapp.example.com/index.php/mymodule

`$ symfony propel-build-schema [xml]`
Gera o schema.yml para representação de um banco de dados existente.
Para gerar o schema.xml utilize a opção xml

`$ symfony propel-build-model`
Gera as classes PHP do modelo, de acordo com o modelo de dados descrito no schema.xml.
As classes base do modelo para acesso aos dados serão automaticamente criadas no diretório myproject/lib/model/om/:

BaseArticle.php	BaseComment.php
BaseArticlePeer.php	BaseCommentPeer.php

Também serão criadas as classes do modelo para acesso aos dados em myproject/lib/model:

Article.php	Comment.php
ArticlePeer.php	CommentPeer.php

`$ symfony propel-generate-crud <application_name> <module_name> <ClassName>`
Scaffolding - Gera um novo módulo com as funções CRUD, baseado em uma classe do modelo

`$ symfony propel-build-sql`
Gera o código SQL, para criar as tabelas descritas no schema.yml, no arquivo myproject/data/sql/lib.model.schema.sql

`$ symfony propel-build-db`
Cria um banco de dados vazio

`$ symfony propel-insert-sql`
Insere o código SQL do arquivo myproject/data/sql/lib.model.schema.sql no banco de dados

`$ symfony sync <environment_name> [go]`
Sincroniza o projeto atual com outra máquina

`$ symfony propel-init-admin <application_name> <module_name> <ClassName>`
Inicializa um novo módulo de administração, baseado em uma classe do modelo

`$ symfony test <application_name>`
Inicia o conjunto de teste para uma aplicação

`$ symfony plugin-install [local|global] <channel_name>/<plugin_name>`
Instala um novo plugin

`$ symfony freeze`
Converte o projeto para uma aplicação independente (stand-alone)

`$ symfony unfreeze`
Reverte o projeto para seu estado inicial. Apaga o data/symfony/, lib/symfony/ e web/sf/

`$ symfony disable <application_name> <environment_name>`
Temporariamente desativa sua aplicação. Necessário quando for atualizar uma biblioteca ou uma grande quantidade de dados

`$ symfony enable <application_name> <environment_name>`
Ativa a aplicação e limpa o cache

`$ symfony clear-controllers`
Remove do diretório web/ todos os controllers que não deverão ser executados em um ambiente de produção. Caso não forem incluídos os front controllers de desenvolvimento no arquivo rsync_exclude.txt, este comando garante que um backdoor não revelará informações internas de sua aplicação.

`$ symfony fix-perms`
Modifica as permissões dos diretórios log/ e cache/ para 0777
(estes diretórios necessitam ser graváveis para o framework trabalhar corretamente)

`$ symfony log-purge`
Apaga os arquivos de log do symfony nas aplicações e ambientes onde o arquivo logging.yml especifica a opção purge: on (que é o valor default)