

Mon expérience d'utilisation d'outils web pour gérer des sites de projets



Albert Rilliard

Note : je ne suis PAS un programmeur web

2 types de sites – 2 types d'outils

- ▶ **Projet PADE (Prosodie Accents Dialecte Expressivité):**
 - ▶ Projet ANR
 - ▶ Site PADE:
 - ▶ Mise en commun de données et communication externe (informations, references, articles, etc)
 - ▶ Gestion d'utilisateurs et de droits
 - ▶ Ajouts / modifications des infos

- ▶ WIKI : <http://pade.limsi.fr>

2 types de sites – 2 types d'outils

- ▶ **Projet AMPER (Atlas Multimedia Prosodique de l'Espace Roman):**
 - ▶ réseau scientifique permettant la collecte de données dialectales comparables à travers l'ensemble de l'espace roman (europe / Amérique latine)
 - ▶ **Site AMPER:**
 - ▶ Centralisation des données (BD avec les résultats des analyses, stockage des fichiers multimédia)
 - ▶ Consultation, modification, comparaison d'enregistrement
 - ▶ Consultation cartographique
 - ▶ Inscription & gestion d'utilisateurs et de droits
 - ▶ Modifications ponctuelles des données dans la base
- ▶ **Framework (symfony) :** <http://amper.limsi.fr>

Symfony

- ▶ <http://www.symfony-project.org/>
- ▶ **Wikipedia :**
 - ▶ Symfony utilise PHP5 (version $\geq 5.2.4$).
 - ▶ **Modèle MVC**
 - ▶ **Templating simple avec des jeux de "helpers"**
 - ▶ Performances optimisées, système de cache
 - ▶ **Gestion des URL indépendamment de l'arborescence fonctionnelle**
 - ▶ **Système de configuration en cascade (en langage YAML)**
 - ▶ **Générateur de back-office et de module (scaffolding)**
 - ▶ **Support de l'internationalisation**
 - ▶ **Abstraction des données de la BD (ORM) – Doctrine ou Propel**
 - ▶ **Support d'AJAX**
 - ▶ Architecture extensible (utilisation / création de plugins)
 - ▶ Utilisation de la ligne de commande (obligatoire)
 - ▶ Système de tests fonctionnels

Quelques plugins

- ▶ **sfDoctrineGuardPlugin** : gestion d'utilisateurs avec système de droits géré par des credentials (configurables facilement)
- ▶ **sfJqueryReloadedPlugin** : fonctionnalités liées à AJAX (mais du diable si je me rappelle lesquelles !)
- ▶ **sfDoctrineAdminGeneratorWithShowPlugin** : thème pour générer le système d'administration

J'ai apprécié ces fonctionnalités

- ▶ Structure MVC oblige à faire les choses proprement (c'est en plus instructif)
- ▶ Didacticiel bien fait et très complet (et en français)
- ▶ Communauté d'utilisateurs suffisamment importante pour obtenir des solutions "google" à toutes mes questions et mes problèmes (messages d'erreurs...)
- ▶ Basé sur des outils standards (et installés au LIMSI 😊)
- ▶ Système de configuration simple et performant

Routing

```
## route d'accueil :
accueil_show:
  url: /accueil.:sf_format
  param: { module:accueil, action: index, sf_format: html }

homepage:
  url: /
  param: { module: accueil, action: index }

# pour les utilisateurs :
new_user:
  url: /user/new
  param: { module: user, action: new, sf_format: html }
  requirements:
    sf_method: post

#### routes pour gerer les consultations dynamiques :
selpts_submit:
  url: /selpts/submit.:sf_format
  param: { module: selpts, action: submit, sf_format: html }
selpts_update_domaines_list:
  url: /selpts/update-domaines-list.:sf_format
  param: { module: selpts, action: updateDomainesList, sf_format: html }
  requirements:
    sf_method: post
```

```
##### Routes generees automatiquement pour les modales
"administrateurs" :
amper_user:
  class: sfDoctrineRouteCollection
  options:
    model:          AmperUser
    module:         user
    prefix_path:    /user
    column:         token
    with_wildcard_routes: true

amper_phrase:
  class: sfDoctrineRouteCollection
  options:
    model:          AmperPhrase
    module:         phrase
    prefix_path:    /phrase
    column:         id
    with_wildcard_routes: true

# generic rules
# please, remove them by adding more specific rules
default_index:
  url: /:module
  param: { action: index }

default:
  url: /:module/:action/*
```

Definition de la structure de données

DEFINITION DE LA STRUCTURE DE LA BD

AmperEspace:

```
columns:
  espace: { type: string(255), notnull: true, unique: true }
  coord_espace: { type: text, notnull: true }
  carte_espace: { type: string(512) }
```

AmperDomaine:

```
columns:
  domaine: { type: string(255), notnull: true, unique: true }
  espace_id: { type: integer, notnull: true }
  coord_domaine: { type: text, notnull: true }
  carte_domaine: { type: string(512) }
  comite_id: { type: integer, notnull: true }
relations:
  AmperEspace: { onDelete: CASCADE, local: espace_id, foreign: id,
foreignAlias: AmperDomaines }
```

AmperInformant:

```
actAs: { Timestampable: ~ }
columns:
  ptenquete_id: { type: integer, notnull: true }
  code_locuteur: { type: string(255), notnull: true }
  initiales: { type: string(10) }
  age: { type: integer }
  niveau_instruction: { type: string(255) }
  autres_infos: { type: string(4000) }
  accord_diffusion: { type: boolean, notnull: true, default: 1 }
  consentement: { type: boolean, notnull: true, default: 1 }
  enqueteur: { type: string(255) }
relations:
  AmperPtenquete: { onDelete: CASCADE, local: ptenquete_id, foreign: id,
foreignAlias: AmperInformants }
  AmperStructmorphs:
    class: AmperStructmorph
    refClass: AmperEnregistrement
    local: informant_id
    foreign: structmorph_id
    foreignAlias: AmperInformants
  AmperModalites:
    class: AmperModalite
    refClass: AmperEnregistrement
    local: informant_id
    foreign: modalite_id
    foreignAlias: AmperInformants
```


Fichiers de configuration

CONFIG GENERALE

```
prod:
  .settings:
    no_script_name: true
    logging_enabled: false
    charset: utf-8

dev:
  .settings:
# error_reporting: <?php echo (E_ALL | E_STRICT)."\n" ?>
  error_reporting: false
  web_debug: true
  cache: false
  no_script_name: true
  etag: false

test:
  .settings:
# error_reporting: <?php echo ((E_ALL | E_STRICT) ^
E_NOTICE)."\n" ?>
  error_reporting: false
  cache: false
  web_debug: false
  no_script_name: true
  etag: false
```

```
all:
  .settings:
    enabled_modules: [default, sfGuardAuth, sfGuardUser, sfGuardGroup,
sfGuardPermission]
    # Form security secret (CSRF protection)
    csrf_secret: 867cccde2c99b79a07d4e2a04fa3244b7955db26
    no_script_name: true
    # Output escaping settings
    escaping_strategy: true
    escaping_method: ESC_SPECIALCHARS
    charset: utf-8

    # Enable the database manager
    use_database: true

  .actions:
    login_module: sfGuardAuth
    login_action: signin
    error_404_module: default
    error_404_action: error404
    secure_module: sfGuardAuth
    secure_action: secure
```

Fichiers de configuration

GESTION DES "VIEW"

default:

http metas:
content-type: text/html

metas:

#title: Projet AMPER
#description: Atlas Multimedia Prosodique de l'Espace Roman
#keywords: prosodie, dialectologie, langues romanes, AMPER
#language: fr
#robots: index, follow
#charset: utf-8

stylesheets: [main.css]

javascripts: []

has_layout: true
layout: layout

Fichiers de configuration

```
# CONFIG D'UN GENERATEUR
```

```
generator:
```

```
class: sfDoctrineGenerator
```

```
param:
```

```
model_class: AmperEnregistrement
```

```
theme: admin
```

```
non_verbose_templates: true
```

```
with_show: false
```

```
singular: ~
```

```
plural: ~
```

```
route_prefix: amper_enregistrement
```

```
with_doctrine_route: true
```

```
actions_base_class: sfActions
```

```
config:
```

```
actions:
```

```
_show:
```

```
label: afficher
```

```
credentials: [[admin]]
```

```
_edit:
```

```
label: modifier
```

```
credentials: [[admin]]
```

```
_new:
```

```
credentials: [[admin]]
```

```
_delete:
```

```
label: supprimer
```

```
credentials: [[admin]]
```

```
_save:
```

```
credentials: [[admin]]
```

```
_filter:
```

```
label: Filtrer
```

```
credentials: [[admin, amper-domaine]]
```

```
_index:
```

```
credentials: [[admin, amper-domaine]]
```

```
fields:
```

```
informant_id: { label: Informant }
```

```
structmorph_id: { label: Structure }
```

```
modalite_id: { label: Modalite }
```

```
list:
```

```
title: Liste des enregistrements
```

```
layout: stacked
```

```
display: [informant_id, structmorph_id, modalite_id]
```

```
params: %%=amper_informant%%_%%amper_structmorph%%_%
```

```
%amper_modalite%%
```

```
max_per_page: 70
```

```
filter:
```

```
display: [informant_id, structmorph_id, modalite_id]
```

```
form: ~
```

```
edit:
```

```
title: Modification de l'enregistrement %%amper_informant%% %
```

```
%amper_structmorph%% %%amper_modalite%%
```

```
new:
```

```
title: Nouvel enregistrement
```



Fichiers de configuration

```
# FICHER DE CONFIG DES OPTINS DE SECURITE
```

```
index:
  is_secure: true
  credentials: [[admin, membre]]
show:
  is_secure: true
  credentials: [[admin, membre]]
filter:
  is_secure: true
  credentials: [[admin, membre]]
new:
  is_secure: true
  credentials: [[admin, amper-domaine]]
edit:
  is_secure: true
  credentials: [[admin, amper-domaine]]
save:
  is_secure: true
  credentials: [[admin, amper-domaine]]

default:
  is_secure: true
  credentials: [admin]
```

Problèmes observés & potentiels

- ▶ Requête « automatiques » TRÈS gourmandes en mémoire
 - ▶ Passage à SQL possible - et recommandé: rapide & économique
- ▶ Certaines fonctions sont complexes (intrinsèquement) et demandent une bonne maîtrise de la programmation web
- ▶ Changements de versions : compatibilité ?