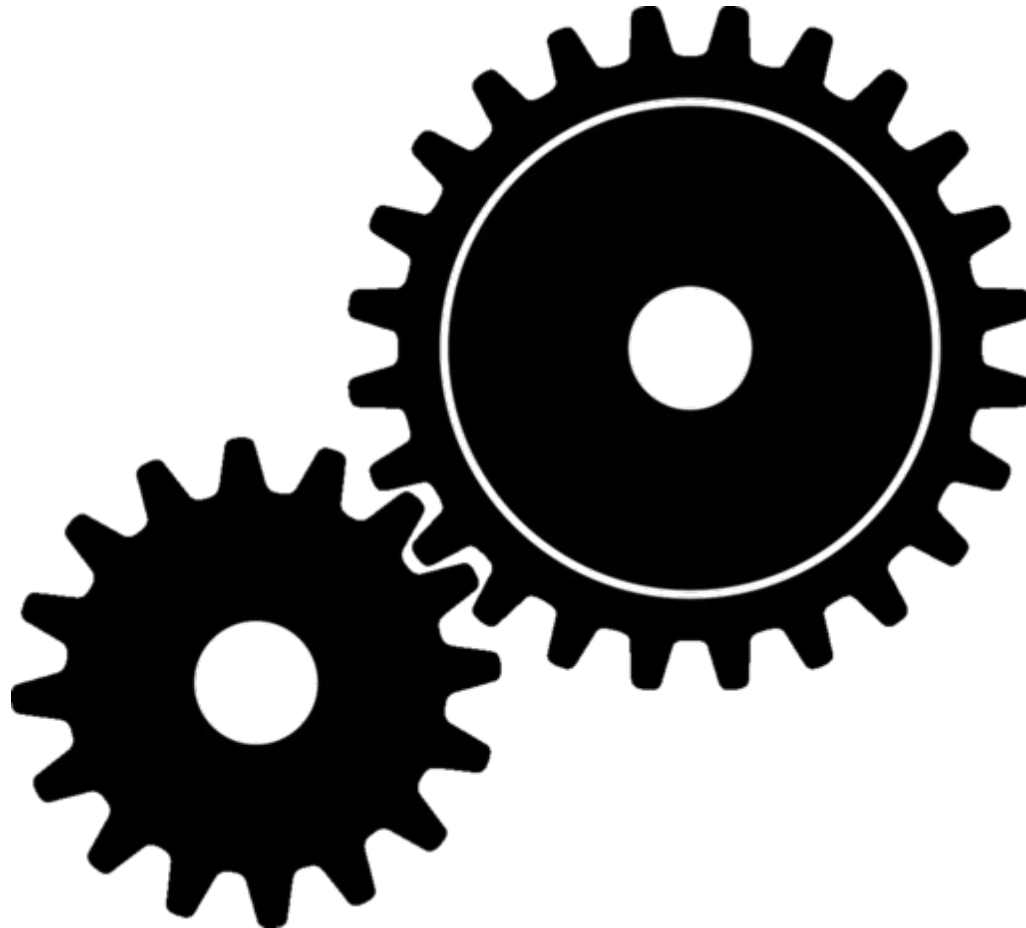


La configuration en Drupal 8 : plusieurs approches



Qui suis-je ?

- Florent Torregrosa (Grimreaper)
- Début d'utilisation avec Drupal 6 en 2011
- Lead developer Drupal chez Smile
- Contributions :
 - ex-membre du bureau de l'ADFF
 - co-mainteneur de drupal.fr et drupalcamp.fr
 - co-organisateur d'événements
 - traductions
 - modules
 - patches



Pourquoi cette présentation ?

- Drupal 8 sorti en novembre 2015
- Les gens rencontrent toujours des problèmes de gestion de la configuration
- La gestion de la configuration du noyau ne suffit pas à répondre à la diversité des besoins sur les projets
- Il existe trop de modules en config_quelque_chose
- Le but de la présentation est d'éviter que cela ressemble à ceci...



Pourquoi cette présentation ?



Drupalcamp Lannion, 28/10/2017 :
« La configuration en Drupal 8 : plusieurs approches »

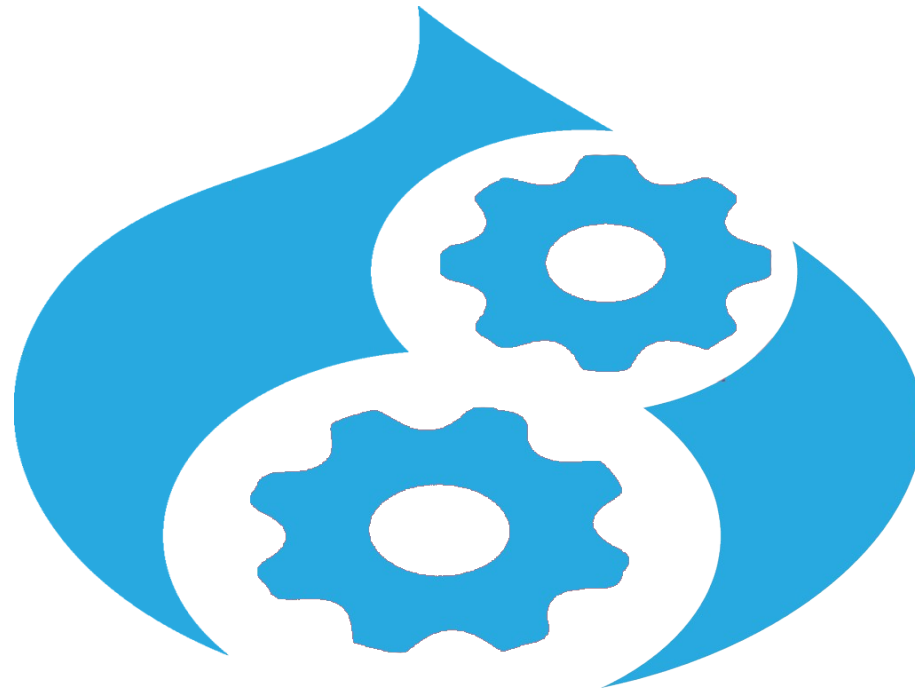


Sommaire

- 1) La gestion de la configuration en Drupal 8
- 2) Présentation de Features
- 3) Présentation de Config Split
- 4) Différents cas d'utilisation



1 : La gestion de la configuration en D8



1 : La gestion de la configuration en D8

Avantages :

- Introduction des entités de configuration
- Stockage uniformisé entre config simple et entité de configuration
- Utilisation du YAML pour le stockage en fichiers
- Traduisible via le noyau
- Surcharge de configuration dans les settings.php



1 : La gestion de la configuration en D8

```
▼ drupalfr_release
  ▼ config
    ► install
    ▼ optional
      views.view.release.yml
    ▼ schema
      drupalfr_releases.schema.yml
  ► src
    drupalfr_release.features.yml
    drupalfr_release.info.yml
    drupalfr_release.links.menu.yml
    drupalfr_release.links.task.yml
    drupalfr_release.module
    drupalfr_release.routing.yml
    drupalfr_release.services.yml
```

- Dans les modules, thèmes, profils d'installation
- config
 - install
 - optional
 - schema
- sous-dossier par langue



1 : La gestion de la configuration en

D8

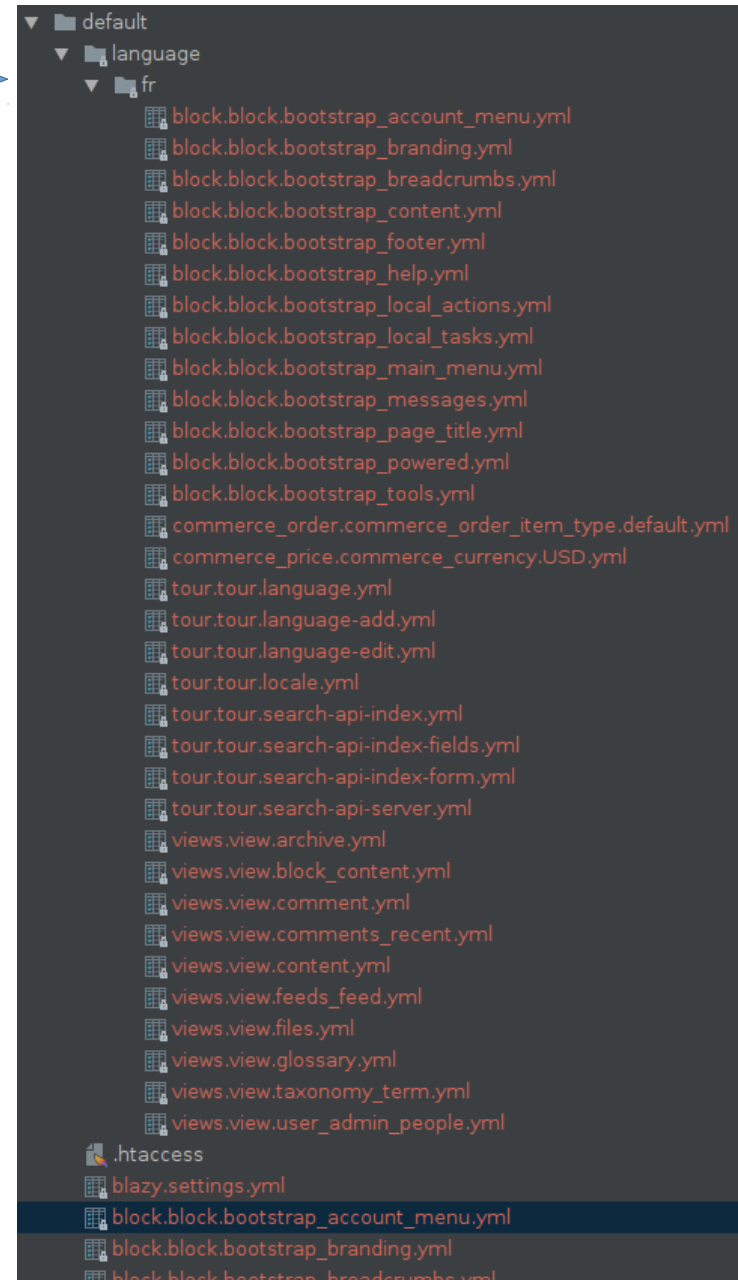
L'export complet avec CMI



Un fichier de configuration YAML



```
uuid: 1f9f2b99-5ed1-4878-bf7f-9e8d5b7118a1
langcode: fr
status: true
dependencies:
  config:
    - system.menu.account
  module:
    - system
  theme:
    - bootstrap
_core:
  default_config_hash: R2K_qX0ZUMbUL6ie_-S9tZVjvqmlWklscikgeRv9vhw
id: bootstrap_account_menu
theme: bootstrap
region: navigation_collapsible
weight: 11
provider: null
plugin: 'system_menu_block:account'
settings:
  id: 'system_menu_block:account'
  label: 'User account menu'
  provider: system
  label_display: '0'
  level: 1
  depth: 2
visibility: { }
```



Drupalcamp Lannion, 28/10/2017 :

« La configuration en Drupal 8 : plusieurs approches »



1 : La gestion de la configuration en D8

Surcharge dans un fichier de settings.php :

- Prise en compte de la valeur à l'exécution
- Non enregistré en base de données

```
$settings['file_private_path'] = '/project/private_files/default';

// Translations.
$config['locale.settings']['translation']['path'] = 'translations/contrib';
$config['locale.settings']['translation']['use_source'] = 'local';

// Redis.
$settings['redis.connection']['host'] = 'redis';

// Varnish.
$config['varnish_purger.settings.varnish']['hostname'] = 'varnish';

// Solr.
$config['search_api.server.solr']['backend_config']['connector_config']['host'] = 'solr';
$config['search_api.server.showcase']['backend_config']['connector_config']['host'] = 'solr-showcase';

// Twitter.
// Go to https://apps.twitter.com to get these tokens.
$config['drupalfr_social.twitter']['consumer_key'] = '';
$config['drupalfr_social.twitter']['consumer_secret'] = '';
$config['drupalfr_social.twitter']['access_token'] = '';
$config['drupalfr_social.twitter']['access_token_secret'] = '';

// Meetup.
$config['drupalfr_meetup.settings']['api_key'] = '';
$config['drupalfr_meetup.settings']['group_urlname'] = '';

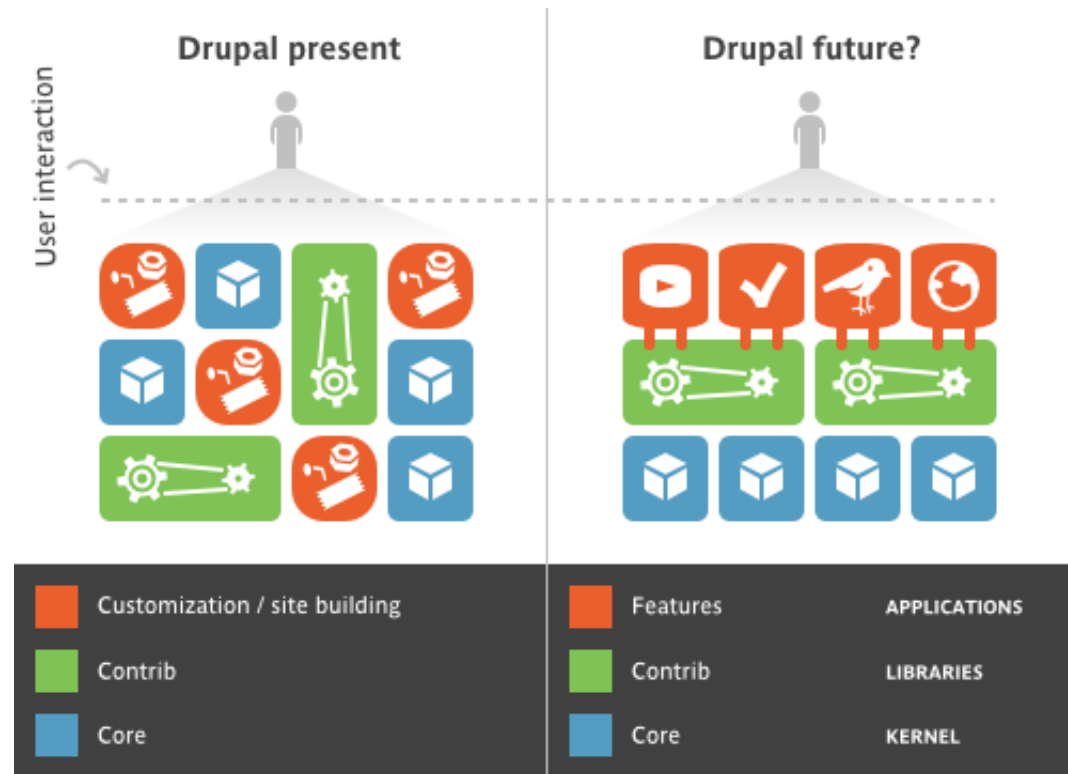
// Mailchimp.
$config['mailchimp.settings']['api_key'] = '';
```

Drupalcamp Lannion, 28/10/2017 :

« La configuration en Drupal 8 : plusieurs approches »



2 : Présentation de Features



2 : Présentation de Features

- À l'origine outil de packaging de la configuration pour réutilisation
- Utiliser pour le déploiement en D6, D7 et D8
- En D8, Features est pensé pour que le module/feature soit activé puis après la conf soit gérée via CMI.
- Ne permet pas la suppression de configuration au déploiement.



2 : Présentation de Features

Bundle assignment

Fonctionnalités Configure Bundles Differences

Accueil » Administration » Configuration » Development » Fonctionnalités

Bundles are used to collect together groups of features. A bundle provides a shared namespace for all features included in it, which prevents conflicts and helps distinguish your features from those produced for other purposes. Common uses of bundles include:

- Custom features for use on a particular site.
- The features of a given distribution.

Use the form below to manage bundles. Each bundle comes with a set of assignment methods. By configuring and ordering the assignment methods, you can set the defaults for what does and doesn't get packaged into features for your bundle. Use the *Bundle* select to choose which bundle to edit, or chose --New-- to create a new bundle. The *Default* bundle does not include a namespace and cannot be deleted.

PAQUET

Drupalfr

Bundle name *

Drupalfr Machine name: drupalfr (Modifier)

A unique human-readable name of this bundle.

Distribution description

Auto-generated bundle from package Drupalfr

A description of the bundle.

Include install profile

Select this option to have your features packaged into an install profile.

Nom du profil *

drupalfr

The machine name (directory name) of your profile.

[Afficher le poids des lignes](#)

ASSIGNMENT METHOD	DESCRIPTION	ACTIVÉ	OPERATIONS
+ Colis	Detect and add existing package modules.	<input checked="" type="checkbox"/>	
+ Exclude	Exclude configuration items from packaging by various methods including by configuration type. When configuration is excluded, it won't be automatically reassigned to other packages.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Base type	Use designated types of configuration as the base for configuration package modules. For example, if content types are selected as a base type, a package will be generated for each content type and will include all configuration dependent on that content type.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Namespace	Add config to packages that contain that package's machine name.	<input checked="" type="checkbox"/>	
+ Optional type	Assign designated types of configuration to the 'config/optional' install directory. For example, if views are selected as optional, views assigned to any feature will be exported to the 'config/optional' directory and will not create a dependency on the Views module.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Alter	Alter configuration items before they are exported. Altering includes options such as removing permissions from roles.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Forward dependency	Add to packages configuration on which items in the package depend.	<input checked="" type="checkbox"/>	
+ Core type	Assign designated types of configuration to a core configuration package module. For example, if image styles are selected as a core type, a core package will be generated and image styles will be assigned to it.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Site type	Assign designated types of configuration to a site configuration package module. For example, if image styles are selected as a site type, a site package will be generated and image styles will be assigned to it.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Profil	Add configuration and other files to the optional install profile from the Drupal core Standard install profile. Without these additions, a generated install profile will be missing some important initial setup.	<input checked="" type="checkbox"/>	<input type="button" value="Configurer"/>
+ Existing	Add exported config to existing packages.	<input checked="" type="checkbox"/>	
+ Dépendance	Add to packages configuration dependent on items already in that package.	<input checked="" type="checkbox"/>	

Drupalcamp Lannion, 28/10/2017 :
« La configuration en Drupal 8 : plusieurs approches »



2 : Présentation de Features

Fonctionnalités

Fonctionnalités

Configure Bundles

Differences

Accueil » Administration » Configuration » Development

Export packages of configuration into modules.

+ Create new feature

Paquet

Drupalfr

<input type="checkbox"/>	FUNCTIONNALITÉ		DESCRIPTION	VERSION	STATUS	ÉTAT
<input type="checkbox"/>	Commerce	drupalfr_commerce	Included configuration		Installé	
<input type="checkbox"/>	Société	drupalfr_company	Provides Société type de contenu and related configuration. Included configuration		Installé	
<input type="checkbox"/>	Contact	drupalfr_contact	Included configuration		Installé	
<input type="checkbox"/>	Core	drupalfr_core	Provide core components required by other features. Included configuration		Installé	Modifié
<input type="checkbox"/>	Événement	drupalfr_event	Provides Événements type de contenu and related configuration. Événement majeur organisé par l'association		Installé	
<input type="checkbox"/>	User	drupalfr_user	Provide User related configuration. Included configuration		Installé	
<input type="checkbox"/>	Drupalfr	drupalfr	drupal.fr site. Included configuration		Installé	Modifié

Block

User account menu Site branding Main page content Footer : Contact Connexion Garder le contact Plan du site Qui sommes nous ?
Help Primary admin actions Tabs Main navigation Status messages Menu du compte de l'utilisateur Navigation principale

Configuration simple

drupalfr_theme.settings

Dépendances

Block Newsletter Help Migrate Plus System User Views

Menu

User account menu Main navigation

Migration Group

Drupalfr

Voir

Glossary

The following packages are not exported.

Unpackaged

unpackaged

Configuration that has not been added to any package.
Included configuration

Not exported

Use an export method button below to generate the selected features.

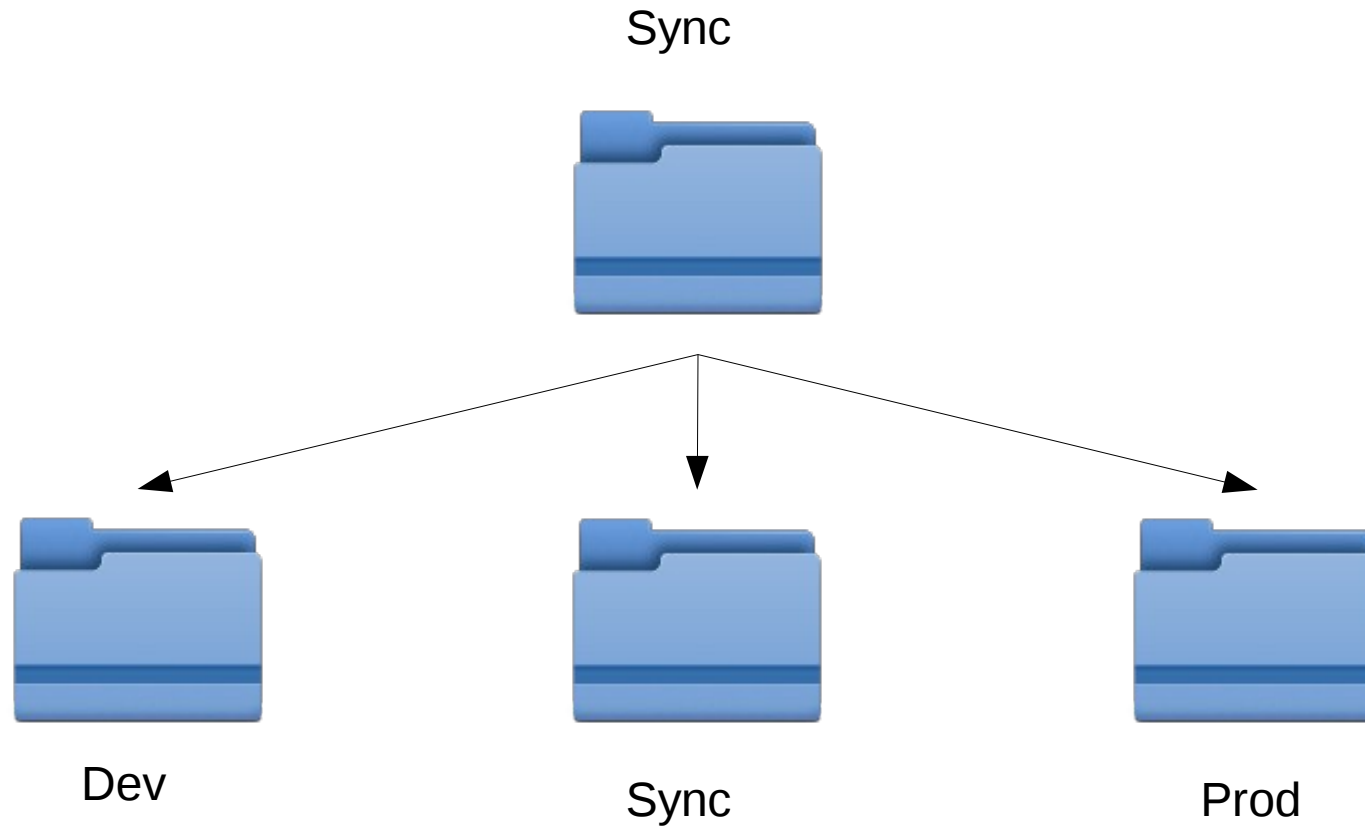
Download Archive

Write

Drupalcamp Lannion, 28/10/2017 :
« La configuration en Drupal 8 : plusieurs approches »



3 : Présentation de Config Split



3 : Présentation de Config Split

- S'appuie sur Config Filter
- Permet de définir des dossiers, en plus du répertoire sync natif, dans lesquels :
 - la configuration va être exportée ou non
 - altère également les modules actifs ou non sur le site
- 2 modes de split



3 : Présentation de Config Split

- Blacklist :
 - une fois sélectionné :
 - un module n'est activé que si le split est actif
 - la configuration qui dépend du module n'est présente que dans ce dossier de split
 - la configuration supplémentaire sélectionnée n'est présente que dans ce dossier de split



3 : Présentation de Config Split

- Graylist :
 - pour la configuration, pas de choix de module directement
 - une fois sélectionnée, la configuration est exportée dans ce dossier de split si elle diffère de la configuration présente dans le dossier sync
 - Permet de fournir une configuration initiale via le dossier de sync puis si modification il y a, elle est exportable dans le split



4 : Différents cas d'utilisation

- Installer un site depuis de la configuration existante
- Faire un profil d'installation
- Ignorer ou modifier certaines configuration par environnement



4 : Différents cas d'utilisation

- Installer un site depuis de la configuration existante
 - => Config installer
 - Part du profil d'installation initial pour le jouer à l'installation puis importer la configuration exportée via CMI.



4 : Différents cas d'utilisation

- Astuce : créer un profil d'installation custom quasi vide, uniquement pour les modules installés initialement

Contenu du fichier
standard.install



```
function standard_install() {
  // Set front page to "node".
  \Drupal::configFactory()->getEditable( name: 'system.site')->set('page.front', '/node')->save( has trusted data: TRUE);

  // Allow visitor account creation with administrative approval.
  $user_settings = \Drupal::configFactory()->getEditable( name: 'user.settings');
  $user_settings->set('register', USER_REGISTER_VISITORS_ADMINISTRATIVE_APPROVAL)->save( has trusted data: TRUE);

  // Enable default permissions for system roles.
  user_role_grant_permissions( rid: RoleInterface::ANONYMOUS_ID, ['access comments']);
  user_role_grant_permissions( rid: RoleInterface::AUTHENTICATED_ID, ['access comments', 'post comments', 'skip comment approval']);

  // Assign user 1 the "administrator" role.
  $user = User::load( id: 1);
  $user->roles[] = 'administrator';
  $user->save();

  // We install some menu links, so we have to rebuild the router, to ensure the
  // menu links are valid.
  \Drupal::service( id: 'router.builder')->rebuildIfNeeded();

  // Enable the Contact link in the footer menu.
  /** @var \Drupal\Core\Menu\MenuLinkManagerInterface $menu_link_manager */
  $menu_link_manager = \Drupal::service( id: 'plugin.manager.menu.link');
  $menu_link_manager->updateDefinition( id: 'contact.site_page', ['enabled' => TRUE]);

  user_role_grant_permissions( rid: RoleInterface::ANONYMOUS_ID, ['access site-wide contact form']);
  user_role_grant_permissions( rid: RoleInterface::AUTHENTICATED_ID, ['access site-wide contact form']);

  // Allow authenticated users to use shortcuts.
  user_role_grant_permissions( rid: RoleInterface::AUTHENTICATED_ID, ['access shortcuts']);

  // Populate the default shortcut set.
  $shortcut = Shortcut::create([
    'shortcut_set' => 'default',
    'title' => t('Add content'),
    'weight' => -20,
    'link' => ['uri' => 'internal:/node/add'],
  ]);
  $shortcut->save();

  $shortcut = Shortcut::create([
    'shortcut_set' => 'default',
    'title' => t('All content'),
    'weight' => -19,
    'link' => ['uri' => 'internal:/admin/content'],
  ]);
  $shortcut->save();

  // Allow all users to use search.
  user_role_grant_permissions( rid: RoleInterface::ANONYMOUS_ID, ['search content']);
  user_role_grant_permissions( rid: RoleInterface::AUTHENTICATED_ID, ['search content']);

  // Enable the admin theme.
  \Drupal::configFactory()->getEditable( name: 'node.settings')->set('use_admin_theme', TRUE)->save( has trusted data: TRUE);
}
```



4 : Différents cas d'utilisation

- Faire un profil d'installation (usine à sites)
 - => Features
 - Gestion des mises à jour potentiellement à l'aide de `hook_update`
 - Pour l'automatisation des traductions, utilisation de `drush_language`, attention cependant, car cela modifie la langue de la configuration.



4 : Différents cas d'utilisation

- Ignorer ou modifier certaines configuration par environnement
 - Features
 - Config split :
 - environnement dev
 - environnement prod
 - + Config ignore si besoin



Conclusion

- Dès le début d'un projet :
 - planifier comment la configuration sera gérée
 - définir ce qui pourra être modifié ou non par le client



Merci pour votre attention

Avez-vous des questions ?



Liens

- https://www.drupal.org/project/config_filter
- https://www.drupal.org/project/config_split
- https://www.drupal.org/project/config_ignore
- https://www.drupal.org/project/config_installer
- <https://www.drupal.org/project/features>
- <https://assos.centrale-marseille.fr/ftorregrosa/blog/la-configuration-en-drupal-8%C2%A0-plusieurs-besoins-et-solutions>
- <https://events.drupal.org/vienna2017/sessions/advanced-configuration-management-config-split-et-al>
- <https://events.drupal.org/vienna2017/sessions/drupal-8-configuration-management-teams>
- <https://www.phase2technology.com/blog/stop-using-features-guide-drupal-8-configuration-management>



Crédits

- <https://www.wikipedia.org> et <https://commons.wikimedia.org> pour les images
- <https://www.zivtech.com> pour le logo Drupal 8 avec engrenages (modifié)
- <https://www.drupal.org/docs/7/modules/features/features-for-drupal-6-and-7> pour l'image de Features

