

Migrer un site Drupal 6 ou Drupal 7 vers Drupal 8

DrupalCamp Nantes 2016

Acquia[®]
THINK AHEAD



Qui suis-je ?

- Senior Technical Solutions Analyst @[Acquia](#)
- Focus : Drupal 8 + 'Next Gen Acquia Cloud'


- Twitter: [@AurelienNavarre](#)
- Site: [anavarre.net](#)

- Twitter: [@DrupalFacile](#)
- Site: [www.drupalfacile.net](#)

Contexte de cette présentation

- Retour d'expérience suite à la migration de la [base de connaissance Acquia](#) de Drupal 6 à Drupal 8
- Quelques challenges :
 - 'End of life' de Drupal 6 (24 février 2016)
 - Contrib, oh my! [SimpleSAMLphp](#), [Search API](#), [Workbench Moderation](#), [Multiversion](#), ...
 - Changement radical du modèle de données - Merci **drush migrate-fields-source** et [Entity Relationship Diagrams](#)

Vous avez dit API de migration ?

 **Experimental modules** are provided for testing purposes only. Use at your own risk.

▼ CORE (EXPERIMENTAL)

- Drupal Upgrade UI** ▶ Provides a user interface for migrating from older Drupal versions.
- Migrate** ▶ Handles migrations
- Migrate Drupal** ▶ Contains migrations from older Drupal versions.

Aperçu rapide des modules de migration

Module	Description
migrate_upgrade	Commande Drush pour exécuter les migrations de Drupal 6 ou 7 vers Drupal 8
migrate_tools	Commandes Drush génériques pour gérer les migrations
migrate_plus	Extension du framework de migration du core avec des fonctionnalités additionnelles (ex : migration depuis des sources XML ou JSON)
migrate_ui	Interface utilisateur construite à partir des entités de Migration
migrate_manifest	Commande Drush pour exécuter une migration SQL à partir de template manifest.yml

Jeu et modèle de données

- Allez-vous migrer 1 millier ou 1 million de données ?
- Souhaitez-vous *vraiment* changer le modèle de données ?
- Migrer via l'UI n'est presque jamais une bonne solution
 - Ne permet pas de migration custom
 - Drush tient (franchement) mieux la charge
 - Contrib a des commandes Drush puissantes

Débuter avec les modules de migration

```
$ drush @site.env en -y migrate_drupal, migrate_upgrade,  
migrate_tools  
The following extensions will be enabled: migrate_drupal,  
migrate_upgrade, migrate_tools, migrate, migrate_plus
```

- Branche **8.x-2.x** pour tous les modules de migration
- Récupérez les identifiants de BDD du site source
- Nombreuses nouvelles commandes de migration !

```
$ drush | grep migrate
```

Configurer la migration initiale

```
$ drush @site.env migrate-upgrade --legacy-db-url=mysql://{USER}:{PASSWORD}@{HOST}:{PORT}/{DB} --legacy-root={URI} --  
configure-only
```

- Une IP vaut mieux qu'un FQDN
- Pensez à **drush -vd** (verbose debug) à la moindre erreur
- **drush config-list** permet d'auditer la configuration

```
$ drush config-list | grep -c migrate  
62
```


La clé de migration est-elle définie ?

```
>>> \Drupal::>>> \Drupal::state()->get('migrate_drupal_6');  
=> [  
  "key" => "upgrade",  
  "database" => [  
    "driver" => "mysql",  
    "username" => "{USER}",  
    "password" => "{PASSWORD}",  
    "host" => "{HOST}",  
    "port" => "{PORT}",  
    "database" => "{DB}",  
    "prefix" => null,  
  ],  
]
```

Exécuter une migration complète

```
$ drush @site.env migrate-upgrade --legacy-db-url=mysql://  
{USER}:{PASSWORD}@{HOST}:{PORT}/{DB} --legacy-root={URI}
```

- **migrate-upgrade** fait le job...mais sans flexibilité
- Nous voulons *contrôler* tous les aspects de la migration
- **migrate-import** répond mieux à nos besoins

Créer un module custom (c'est facile !)

- Créez le fichier ***.info.yml**
- Ajoutez les fichiers de config dans **config/install**
- Installez le module. Voilà.



```
$ drush @site.env config-export -y sync ; cp  
/path/to/config_dir/sync/migrate_plus.migration*  
/path/to/custom/module/config/install/
```

Exécuter une migration custom

```
$ drush @site.env migrate-import --limit=10000 --feedback=1000
```

- Utilisez **--limit** pour restreindre le nombre d'éléments
- Utilisez **--feedback** pour être notifié régulièrement
- Vous pouvez aussi restreindre la migration par Plugin

```
$ drush @site.env migrate-import d6_url_alias,d6_block, d6_file,  
d6_user,{...} --limit=10000 --feedback=1000
```

Statut et messages de Migration

```
$ drush @site.env migrate-status
```

- Utilisez **migrate-status** pour obtenir une vue des éléments importés, ceux restant à importer, etc.
- Utilisez **migrate-messages** pour voir les erreurs

```
$ for i in upgrade_d6_node_type upgrade_d6_user {...} ; do drush @site.env migrate-messages $i ; done
```

Et les migrations *vraiment* custom ?

- C'est toujours Migrate tel que vous le connaissez !
 - **prepareRow()**
 - **query()**
 - ...
- 'Étendez' les Plugins de migration du core
- Écrivez vos propres Plugins

Exemple de 'source Plugin'

```
<?php

namespace Drupal\migrate_webinar\Plugin\migrate\source;

use Drupal\path\Plugin\migrate\source\d6\UrlAlias;

/**
 * @MigrateSource(
 *   id = "webinar_d6_url_alias"
 * )
 */
class WebinarUrlAlias extends UrlAlias {

    public function query() {
        $query = parent::query();
        $query->condition('src', 'user/%', 'NOT LIKE');
        return $query;
    }
}
```

source Plugin du core

Annotation Doctrine

Logique custom via la méthode query()

Migration continue du contenu

- Pouvez-vous vous permettre une coupure du site ?
- Utilisez le tunneling SSH pour sécuriser les transferts

```
$ ssh -i /home/{USERNAME}/{SSH_KEY} -f -L 1111:{DB_SERVER}:3306  
{USER}@{WEB_SERVER} -N
```

- <https://docs.acquia.com/articles/ssh-tunneling-server-side-applications>

Vérifier la cohérence des fichiers

```
mysql> SELECT COUNT(fid) FROM file_managed;
```

```
$ find . ! -path "*styles*" ! -path "*ctools*" ! -path "*css*" !  
-path "*js*" ! -path "*xmlsitemap*" -type f | wc -l
```

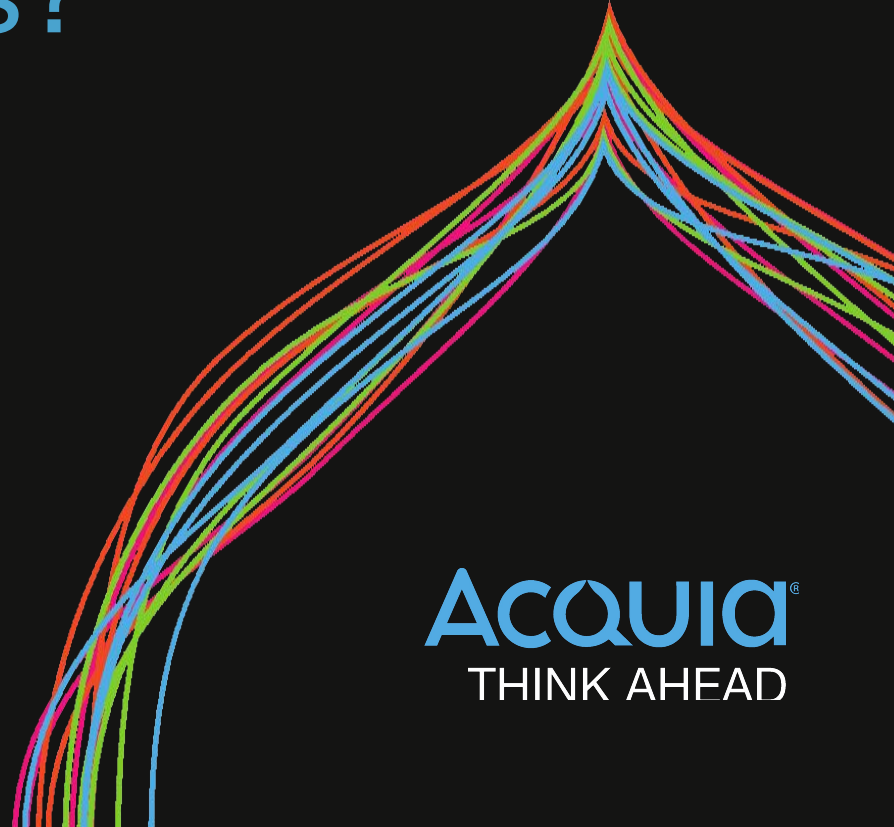
- Les fichiers ne sont pas forcément tous gérés par Drupal
- Utilisez **rsync** quand Migrate API ne peut pas aider

```
$ rsync -avzt --progress --exclude=styles --exclude=ctools --  
exclude=css --exclude=js --exclude=xmlsitemap -e "ssh -i /home/  
{USERNAME}/.ssh/id_rsa" {SOURCE_SERVER}:{SOURCE_PATH} .
```

À garder en mémoire...

- L'API de migration API évolue encore régulièrement
 - Changements majeurs avec 8.1.x - Lisez <http://mikeryan.name/blog/mikeryan/migration-update-for-drupal-8-1>
- Vous voyez **d6** ou **d7** dans un fichier de migration ?
 - Cela signifie un Plugin spécifique à Drupal 6 ou 7
 - Autrement, la migration est agnostique
- Pour les migrations custom, démarrez 'à l'envers'
 - Partez d'un jeu de migration minimaliste
 - Résolvez les dépendances au fur et à mesure

Merci - Questions?



Acquia[®]
THINK AHEAD