



D8 Module Porting 101

*Porting **Simple FB Connect** Module to D8*

Tanay Sai

Drupal is a registered trademark of Dries Buytaert.

Preface

This is no definitive resource. And is no where close to being completely accurate.

This is just a collection of notes I took while porting the module https://www.drupal.org/project/simple_fb_connect to Drupal 8.

The module is available for you to try on

<http://www.simplefbconnect.com> (D7 version) and at

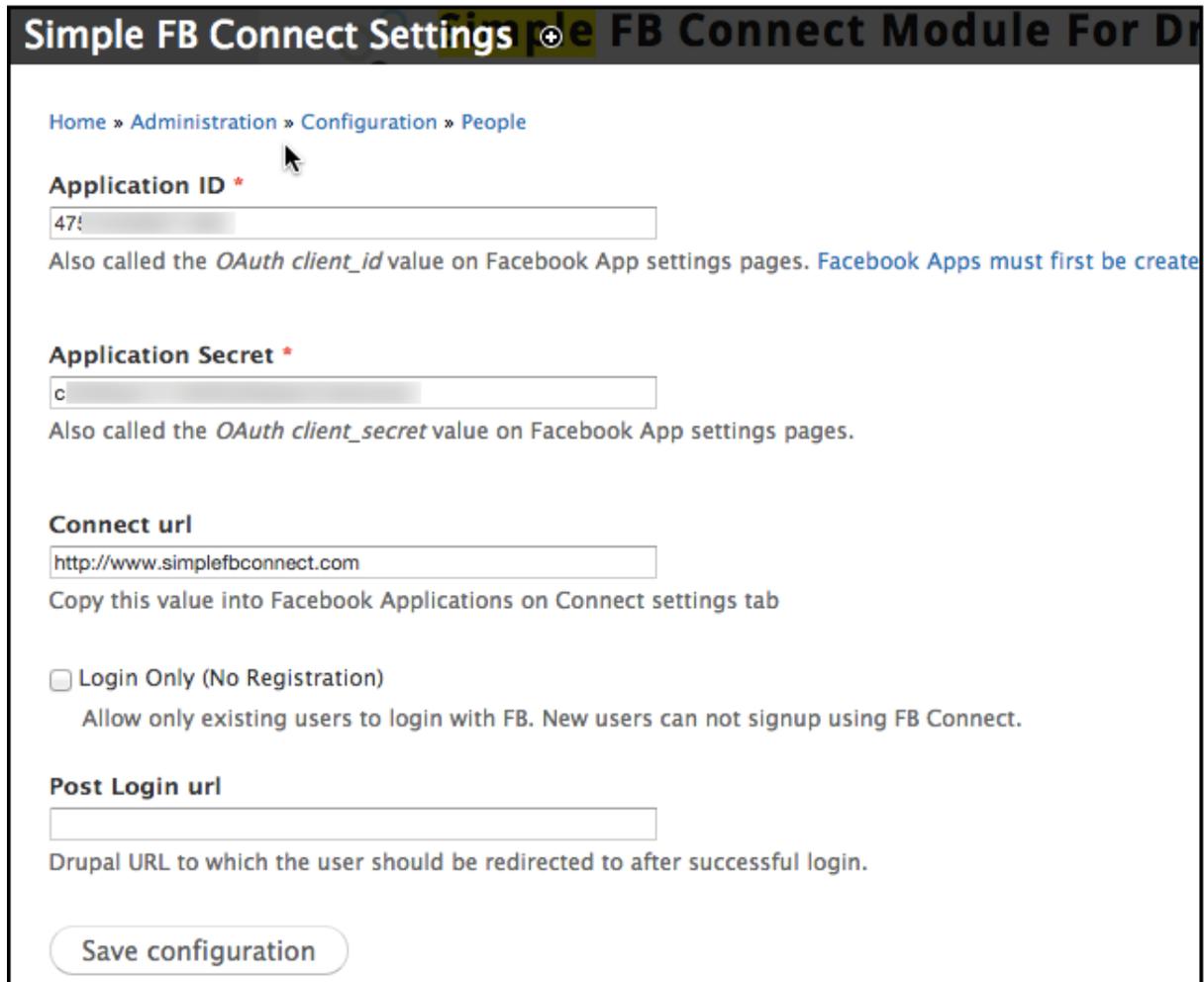
<http://sfcafytbsvqbk.devcloud.acquia-sites.com> (for D8).

Thanks to Piyuesh(@piyuesh23) and Adam Malone(@adammalone) who helped me in getting started with the porting.

Thanks to my colleague Amit Goyal for his technical review of this writing.

About the Module

This is a simple module that allows Drupal sites to use Facebook Connect for Login and Registration - https://www.drupal.org/project/simple_fb_connect



The screenshot shows the 'Simple FB Connect Settings' page. At the top, there is a breadcrumb trail: 'Home » Administration » Configuration » People'. The page contains several configuration fields:

- Application ID ***: A text input field containing '47!'. Below it, a note states: 'Also called the *OAuth client_id* value on Facebook App settings pages. Facebook Apps must first be create'.
- Application Secret ***: A text input field containing 'c'. Below it, a note states: 'Also called the *OAuth client_secret* value on Facebook App settings pages.'
- Connect url**: A text input field containing 'http://www.simplefbconnect.com'. Below it, a note states: 'Copy this value into Facebook Applications on Connect settings tab'.
- Login Only (No Registration)**: A checkbox with the label 'Login Only (No Registration)'. Below it, a note states: 'Allow only existing users to login with FB. New users can not signup using FB Connect.'
- Post Login url**: A text input field. Below it, a note states: 'Drupal URL to which the user should be redirected to after successful login.'

At the bottom of the form, there is a 'Save configuration' button.

As of Dec 1 2014, around 650 websites report using the module.

The module is simple in its functionality. It provides a universal url for login/registration with Facebook. If the user's email address is found on the Drupal site, the user is simply logged in. Else the user is automatically registered and logged in.

info File

Lets see the module's .info file of the D7 Version.

```
1 name = Simple FB Connect
2 description = "A simple module to allow users to login/singup with their Facebook account"
3 package = Facebook
4 core = 7.x
5 dependencies[] = libraries
6 configure = admin/config/people/simple-fb-cbnnect
```

The .info files in D7 are now .info.yml files on D8.

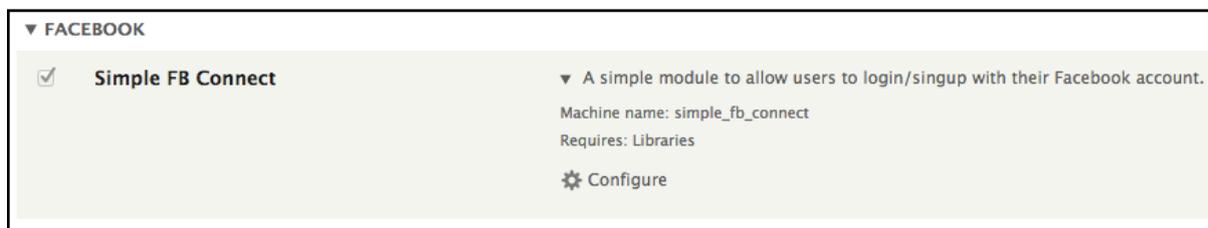
The meta data on the info file is more or less the same. Just a small change to the syntax of the file.

Filename: **simple_fb_connect.info.yml**

```
name: Simple FB Connect
type: module
description: 'A simple module to allow users to login/singup with their Facebook account.'
package: Facebook
core: 8.x
dependencies:
  - libraries
configure: simple_fb_connect_admin
```

One significant difference you can note is that the configure link used to take the url (*admin/config/people/simple-fb-connect*) in D7. In D8, it however takes the routing entry name(*simple_fb_connect_admin*).

Here goes our module on the module Page! (Had to enable the dependency *Libraries* Module as well).



The screenshot shows the Drupal module page for 'Simple FB Connect'. The page is titled 'FACEBOOK' and features a checkmark next to the module name. The description reads: 'A simple module to allow users to login/singup with their Facebook account.' Below the description, the machine name is listed as 'simple_fb_connect', and the required dependencies are 'Libraries'. A 'Configure' link with a gear icon is also visible.

Additional Resources

Documentation on d.o: <https://www.drupal.org/update/modules/7/8/module-info>

Corresponding Change Record: <https://www.drupal.org/node/1935708>

Routing

So, we have the module up. The module does nothing as of now except that it appears on the Module listing. The next big thing would be getting the urls to work. We have 2 urls defined in the D7 version of the module. One is for the login/registration using the module. The other is the module configuration page.

```
12  /**
13   * Implements hook_menu().
14   */
15  function simple_fb_connect_menu() {
16    $items['user/simple-fb-connect'] = array(
17      'title' => (variable_get('simple_fb_connect_login_only', 0) == 0 ? 'Register/' : '') . 'Login with FB',
18      'page callback' => 'simple_fb_connect_login',
19      'access callback' => 'user_is_anonymous',
20      'type' => MENU_LOCAL_TASK,
21    );
22
23    $items['admin/config/people/simple-fb-connect'] = array(
24      'title' => 'Simple FB Connect Settings',
25      'page callback' => 'drupal_get_form',
26      'page arguments' => array('simple_fb_connect_api_keys_settings'),
27      'access arguments' => array('administer simple fb'),
28      'file' => 'simple_fb_connect.admin.inc',
29    );
30    return $items;
31  }
32
```

The D7 version of the module has the routing defined in hook_menu on the .module file.

The routing information would go into routing.yml file - **simple_fb_connect.routing.yml** file

```
simple_fb_connect_login:
  path: 'user/simple-fb-connect'
  defaults:
    _content: '\Drupal\simple_fb_connect\Controller\SimpleFBConnectController::unified_login_register'
    _title: 'Register / Login with FB'
  requirements:
    _role: 'anonymous'

simple_fb_connect_admin:
  path: '/admin/config/people/simple-fb-connect'
  defaults:
    _title: 'Simple FB Connect Settings'
    _form: '\Drupal\simple_fb_connect\Form\SimpleFBConnectAdmin'
  requirements:
    _permission: 'administer simple fb'
```

UPDATE: Starting from D8 Beta 4, the “*_content*” should be “*_controller*”.

<https://www.drupal.org/node/2378809>

D7

The callbacks defined in routing were usually simple function names defined in the same module file or in a .inc file (in which case the file name is specified in the routing entry).

The key for each routing entry is the url itself.

You could specify an *access callback* (in case of a function to determine access) or *access arguments* in case you wish the url to be accessible to those who have certain permissions.

Ex:

```
'access callback' => 'user_is_anonymous',
```

```
'access arguments' => array('administer simple fb'),
```

D8

The callback is a function specified along with its namespace that looks like “\Drupal\simple_fb_connect \Controller \SimpleFBConnectController::unified_login_register”

The key to each routing entry is a “machine name” for the route (not the url itself)

Almost Similar. You can specify a role or a permission.

Ex:

```
_role : 'anonymous'
```

```
_permission: 'administer simple fb'
```

However you can no more simply specify an access callback function here. Instead if you want to handle access logic on the controller, as you might need some values of the URL use '_access: TRUE' and throw exceptions on the controller function.

Though not covered on this module, remember the % wildcards that you used to have while defining route items in hook_menu? So, how would you handle those on D8? Check out the nice example by Vijay @ <https://www.drupal.org/node/1800686#comment-7089472>

Permissions

Defining permissions on D7 was through `hook_permission()` in the `.module` file.

```
/**
 * Implements hook_permission().
 */
function simple_fb_connect_permission() {
  $items = array();
  $items['administer simple fb'] = array(
    'title' => "Administer Simple Facebook Connect Module settings"
  );
  return $items;
}
```

In D8, permissions would be defined in `permissions.yml` file.

`simple_fb_connect.permissions.yml`

```
'administer simple fb':
  title: Administer Simple Facebook Connect
  description: Administer Simple Facebook Connect Module settings
```

Additional Resources

Change Record on d.o: <https://www.drupal.org/node/2311427>

Module File

The following from the module file remained more or less the same on the D8 port:

- Some Global Variable definitions
- Implementation of hook_libraries_info
- A helper function simple_fb_connect_facebook_client_load_include()
- Another helper function facebook_client() (yeah, could ideally be simple_fb_connect_facebook_client)
- Another helper function simple_fb_connect_unique_user_name

Too long for a screen-grab. You can take a look at the code of the file @ http://cgit.drupalcode.org/simple_fb_connect/tree/simple_fb_connect.module?h=8.x-1.x

src/controller folder

Remember the routing we defined for the FB login URL?

We pointed the route to a controller and a method in the controller.

```
simple_fb_connect_login:  
  path: 'user/simple-fb-connect'  
  defaults:  
    _content: '\Drupal\simple_fb_connect\Controller\SimpleFBConnectController::unified_login_register'  
    _title: 'Register / Login with FB'  
  requirements:  
    _role: 'anonymous'
```



We would now add the controller file. Whose name would be the same as defined in the routing Entry, with a “.php” appended to it. - **SimpleFBConnectAdmin.php**

The content of this file would be a class definition that extends *ControllerBase* class. The class contains just one function which handles the whole of login and registration logic.

The code here is very much similar. Except for some differences:

D7	D8
<code>drupal_goto("some url");</code>	<code>return \$this->redirect('drupal route name');</code> <i>OR</i> <code>return new RedirectResponse("any absolute url");</code> Requires the following to be included in the controller to use this: <code>use Symfony\Component\HttpFoundation\RedirectResponse;</code>
<code>variable_get('simple_fb_connect_appid', 0)</code>	<code>\Drupal::config('simple_fb_connect.settings')->get('simple_fb_connect_appid')</code>
<code>check_plain()</code>	<code>String::checkPlain()</code> which is available by including <code>use Drupal\Component\Utility\String;</code>

D7

Querying the DB for the existing User:

```
$query = db_select('users', 'u');

$query->condition('u.mail',
  check_plain($fb_user_profile['email']));

$query->fields('u', array('uid'));
$query->range(0, 1);

$drupal_user_id = 0;
$result = $query->execute();

foreach ($result as $record) {
  $drupal_user_id = $record->uid;
}
```

Creating a User Programatically:

```
$fields = array(
  'name' =>
  $drupal_username_generated,
  'mail' =>
  check_plain($fb_user_profile['email']),
  'pass' => $password,
  'status' => 1,
  'init' => 'email address',
  'roles' => array(
    DRUPAL_AUTHENTICATED_RID
    => 'authenticated user',
  ),
);
//Do more stuff with $fields here
$saccount = user_save("", $fields);
```

D8

```
$query = db_select('users_field_data', 'u');

$query->condition('u.mail',
  String::checkPlain($fb_user_profile['email']));
;

$query->fields('u', array('uid'));
$query->range(0, 1);

$drupal_user_id = 0;
$result = $query->execute()-
>fetchAll(PDO::FETCH_ASSOC);

if (count($result))
  $drupal_user_id = $result[0]['uid'];
```

```
$fields = array(
  'name' =>
  $drupal_username_generated,
  'mail' =>
  String::checkPlain($fb_user_profile['email']),
  'pass' => $password,
  'status' => 1,
  'init' => 'email address',
  'roles' => array(
    DRUPAL_AUTHENTICATED_RID
    => 'authenticated user',
  ),
);
//Do more stuff with $fields here
$saccount = entity_create('user',
  $fields);
$saccount->save();
```

src/form folder

Remember the second route entry for the config form? We had connected the route entry to `'\Drupal\simple_fb_connect\Form\SimpleFBConnectAdmin'`. The form would now have to be defined at `src/Form/SimpleFBConnectAdmin.php`

```
simple_fb_connect_admin:  
  path: '/admin/config/people/simple-fb-connect'  
  defaults:  
    _title: 'Simple FB Connect Settings'  
    _form: '\Drupal\simple_fb_connect\Form\SimpleFBConnectAdmin'  
  requirements:  
    _permission: 'administer simple fb'
```

This is similar to routing an url to `drupal_get_form` in D7.

The contents of this file `SimpleFBConnectAdmin.php` would be a class that extends `ConfigFormBase`. Here we would be overwriting some of the functions from the parent class - In this case - `getFormID()`, `buildForm()`, `submitForm()`

```
<?php  
2  
3 namespace Drupal\simple_fb_connect\Form;  
4  
5 use Drupal\Core\Form\ConfigFormBase;  
6 use Drupal\Core\Form\FormStateInterface;  
7  
8 class SimpleFBConnectAdmin extends ConfigFormBase {  
9  
10  /**  
11   * {@inheritdoc}  
12  */  
13  public function getFormID() {  
14    return 'simple_fb_connect_api_keys_settings';  
15  }  
16  
17  /**  
18   * {@inheritdoc}  
19  */  
20  public function buildForm(array $form, FormStateInterface $form_state) {  
21    // dpm(\Drupal::config('simple_fb_connect.settings')->get('simple_fb_connect_a  
22    $form['simple_fb_connect_appid'] = array(  
23      '#type' => 'textfield',  
24      '#required' => TRUE,  
25      '#title' => t('Application ID'),  
26      '#default_value' => $this->config('simple_fb_connect.settings')->get('simple_fb  
27      '#description' => t('Also called the <em>OAuth client_id</em> value on Facebook  
28    );  
29  
30    $form['simple_fb_connect_skey'] = array(  
31      '#type' => 'textfield',  
32      '#required' => TRUE,
```

We would include the below to be able to extend *ConfigFormBase*.

```
use Drupal\Core\Form\ConfigFormBase;
use Drupal\Core\Form\FormStateInterface;
```

Building the form array is very similar to how it was done in D7.

D7	D8
drupal_get_form() as callback	Specify <code>_form</code> in the routing entry. Extend <i>ConfigFormBase</i> .
Build the form and return the form array in the callback.	Override the <code>buildForm()</code> function

The submit handler of the form:

```
/**
 * {@inheritdoc}
 */
public function submitForm(array &$form, FormStateInterface $form_state) {
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_appid', $form_state->getValue('simple_fb_connect_appid'));
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_skey', $form_state->getValue('simple_fb_connect_skey'));
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_login_only', $form_state->getValue('simple_fb_connect_login_only'));
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_post_login_url', $form_state->getValue('simple_fb_connect_post_login_url'));
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_user_pictures', $form_state->getValue('simple_fb_connect_user_pictures'));
    $this->config('simple_fb_connect.settings')->set('simple_fb_connect_picture_dimensions', $form_state->getValue('simple_fb_connect_picture_dimensions'));
    $this->config('simple_fb_connect.settings')->save();
    drupal_set_message($this->t('The configuration options have been saved.'));
}
```

A simple override of *submitForm* function in the form class as mentioned earlier.

D7	D8
<code>variable_set('simple_fb_connect_appid', \$value);</code>	<code>\$this->config('simple_fb_connect.settings')->set('simple_fb_connect_appid', \$value);</code>
<code>\$form_state->['simple_fb_connect_appid']</code>	<code>\$form_state->getValue('simple_fb_connect_appid')</code>

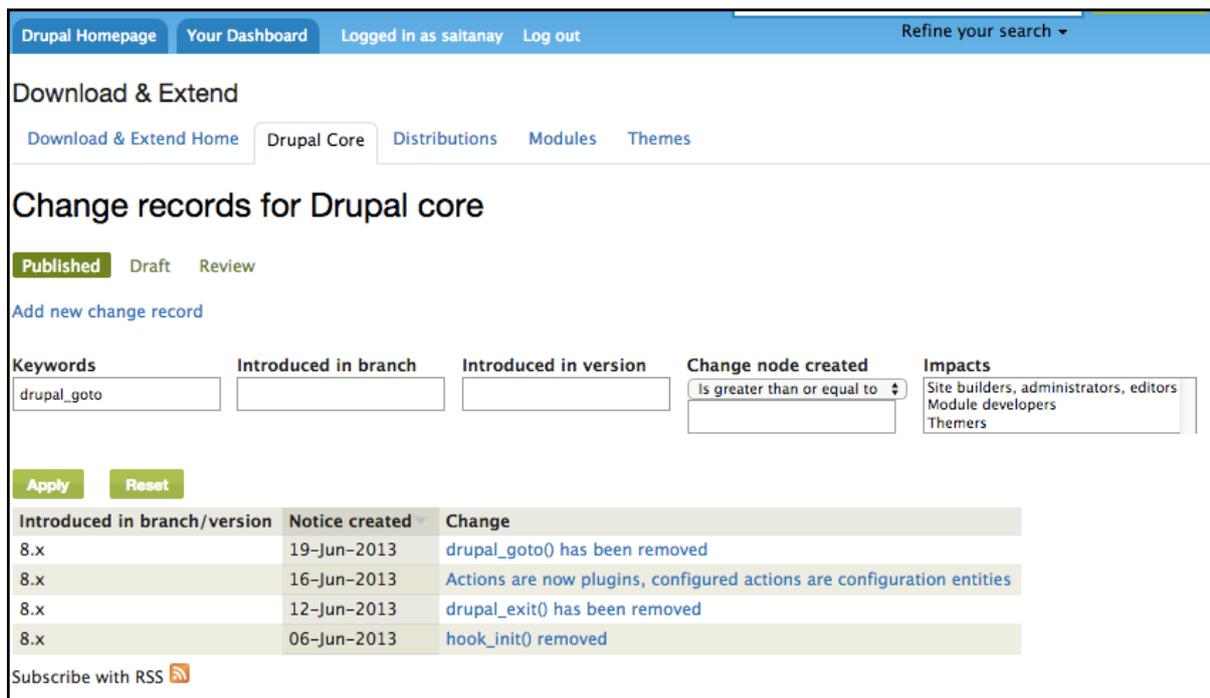
Well that is pretty much it! The above resulted in a fully functional port of the module. Try the module @ <http://sfcafytbsvqbk.devcloud.acquia-sites.com> (Don't be surprised if the site is broken when you try it).

The module is to be improved a lot. Still much of the code is procedural and is not Object Oriented in spirit. This should be worked on. But this was only a small start in a first attempt at porting a module.

Tools Used

Had trouble getting `dsm()` to work. Used `dpm()` with devel's Kint module enabled.

<https://www.drupal.org/list-changes> is your one-stop guide to finding what happened to finding the equivalent of anything from D7 on D8. Example, if you wanted to search for what happened to `drupal_goto` on D8, you could search for the same on the above link!



Drupal Homepage Your Dashboard Logged in as saitanay Log out Refine your search ▾

Download & Extend
Download & Extend Home Drupal Core Distributions Modules Themes

Change records for Drupal core

Published Draft Review

Add new change record

Keywords Introduced in branch Introduced in version Change node created Impacts

drupal_goto [] [] Is greater than or equal to [] Site builders, administrators, editors
Module developers
Themers

Apply Reset

Introduced in branch/version	Notice created ▾	Change
8.x	19-Jun-2013	drupal_goto() has been removed
8.x	16-Jun-2013	Actions are now plugins, configured actions are configuration entities
8.x	12-Jun-2013	drupal_exit() has been removed
8.x	06-Jun-2013	hook_init() removed

Subscribe with RSS 

Heard of another tool - <https://www.drupal.org/project/drupalmoduleupgrader> - I hadn't used it when I worked on this port. But it is one that you should definitely try out.