

Llevando configuración a código con *Features*

Beto Arancibia, @betoscopio
Drupal Chile, Octubre 2014

Por qué?

- El código se puede versionar
- Se pueden resolver conflictos
- Separación de contenido y configuración
- Más fácil de actualizar en producción
- Reutilización de código

Features

Conservar la sanidad mental

Features y amigos

- Strongarm
- Diff
- uuid + features_uuid
- Ctools
- features_override

Organicémonos...

- Tipos de contenidos
- Vistas
- Configuraciones de contenido
- Definición de permisos y roles
- Encapsulación de configuraciones

Evitar

- Dependencias circulares
- Configuraciones muy específicas
- Configuraciones generales del sitio
- Campos comunes

Ejemplos?

- Kit : <https://www.drupal.org/project/kit>
- <https://www.drupal.org/project/panopoly>
- https://www.drupal.org/project/project_distribution

Extras

- `my_feature.module`
- `hook_module_alter()`
- `hook_update()`
- Crea tu perfil de instalación

Ojo con...

- CMI
- Módulo configuration

Conclusiones

- Los datos a al BD
- Definición del sitio TODO en código
- Configuraciones, hacer lo posible por que queden en código.

Referencias

- <http://nuvole.org/blog/code-driven-development>
- <https://www.drupal.org/project/features>
- <https://www.drupal.org/documentation/administer/config>