

A **quick reference** to concepts and practices in Drupal Code-Driven Development.*

* Hungry for more? Check out http://nuvole.org/blog

Working with Drush

Rule Drupal from the command line. For more: http://drush.ws

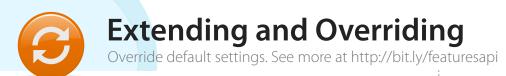


\$ drush fe feature_cool context:feature_cool_front	Create a new feature with "feature_cool_front" Context in it.
\$ drush fu feature_cool	Update a feature: push settings into code.
\$ drush fr feature_cool	Revert a feature: pull settings from code to database.
\$ drush fd feature_cool	Check what has changed: diff code and database.
\$ drush vget process	Get the list of Drupal variable name/value pairs containing "process".
\$ drush vset preprocess_css 1	Set Drupal variable "preprocess_css" value to "1".
\$ drush fe grep context	Get all exportable context names.
\$ drush updatedb - y && drush cc all	Run Drupal update and clear the cache.
\$ drush dl <i>views</i>	Download Views module.
\$ drush en <i>views</i>	Enable Views module.
\$ drush upwd adminpassword="secret"	Change admin password.
<pre>\$ drush sqlq "UPDATE {system} SET schema_version='6001' WHERE filename LIKE '%feature_cool.module'"</pre>	Run a query: roll-back a feature schema number.

4	Speed-up! Clone https://github.com/nuvoleweb/drush_alias and

5 COO	Jump to Drupal root directory from within the Drupal tree.
S Caa Vie liabi	Autocomplete "cdd views" and jump to Views module root directory.
y ulilu, ll, lui lealule c iiabi	Autocomplete "feature_cool" and run "drush [fu,fr,fd] feature_cool".





function hook_user_default_roles_alter(&\$roles)

Alter the default roles right before they are cached into the database.

function hook_user_default_permissions_alter(&\$permissions)

Alter the default permissions right before they are cached into the database.

function hook_content_default_fields_alter(&\fields)

Alter the default cck fields right before they are cached into the database.

CTools also has a variety of hook COMPONENT alter().



Naming Conventions





Component		Naming Convention	Example	
	View	[machine-name]_[view]	blog_list	
	Context	[machine-name]_[context]	blog_front	
	Content Type	[content-type]	blog	
	Imagecache	[content-type]-[descriptor]	blog-xl	



Controller Feature

One feature to rule them all.

```
* Enabling features: feature_rocks
```

```
function feature_controller_update_6001() {
  $return = array();
  $modules = array('feature_rocks');
  features_install_modules($modules);
  $return[] = array('success' => TRUE, 'query' =>
        'Enabling features: '. implode(', ', $modules));
 return $return;
```

```
Add all your project features as dependencies:
   dependencies[] = "feature_cool"
   dependencies[] = "feature_awesome"
  dependencies[] = "feature_rocks"
 name = "Controller Feature"
Project = "feature_controller"
version = "6.x-1.0-dev"
```

\$ drush updatedb -y && drush cc all The following updates are pending: feature example module 6001 - Enabling features: feature rocks

Do you wish to run all pending updates? (y/n):

