

Drupal Code-Driven Development

The Cheat Sheet



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>

<pre>\$ drush fe feature_cool context:feature_cool_front</pre>	Create a new feature with "feature_cool_front" Context in it.
<pre>\$ drush fu feature_cool</pre>	Update a feature: push settings into code.
<pre>\$ drush fr feature_cool</pre>	Revert a feature: pull settings from code to database.
<pre>\$ drush fd feature_cool</pre>	Check what has changed: diff code and database.
<pre>\$ drush vget process</pre>	Get the list of Drupal variable name/value pairs containing "process".
<pre>\$ drush vset preprocess_css 1</pre>	Set Drupal variable "preprocess_css" value to "1".
<pre>\$ drush fe grep context</pre>	Get all exportable context names.
<pre>\$ drush updatedb -y && drush cc all</pre>	Run Drupal update and clear the cache.
<pre>\$ drush dl views</pre>	Download Views module.
<pre>\$ drush en views</pre>	Enable Views module.
<pre>\$ drush upwd admin --password="secret"</pre>	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.



Speed-up!

Clone https://github.com/nuvoleweb/drush_alias and...

```
$ cdd
```

Jump to Drupal root directory from within the Drupal tree.

```
$ cdd vie[TAB]
```

Autocomplete "cdd views" and jump to Views module root directory.

```
$ dr[fu,fr,fd] feature_c[TAB]
```

Autocomplete "feature_cool" and run "drush [fu,fr,fd] feature_cool".



Extending and Overriding

Override default settings. See more at <http://bit.ly/featuresapi>

```
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

Check out: <http://drupal.org/project/kit>



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.

1

```

/*
 * 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;
}

```

2

```

$ 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):

```

