

What is Drupal?

Drupal is a CMS (Content Management System).

A CMS is a system which helps users to organize different types of information (pages, articles, photos, images, forums, ...).

What is content?

A very simple definition is: Content = Information + Semantics.

Semantics (some time called metadata) is computer processable information which allows the storage, processing and retrieval of Information.

How Drupal manages content?

The basic concepts in Drupal are those of NODE (Information) and TAXONOMY (Semantics)

A NODE is a piece of storable information (i.e. an image, a book, an article, a page,...). Nodes are stored in a database and can be retrieved using different approaches.

A TAXONOMY is a classification of concepts which provides semantics. The following is a very simple taxonomy of ACTIVITIES in a school:

- Courses
- Conferences
- Seminars
- Other activities

Each concept in a taxonomy is called in this document TAXON

Taxonomies are useful in order to classify the nodes and retrieve the nodes of a certain TAXON (i.e Courses, Seminars, ...)

More on nodes

The basic distribution of Drupal provides some types of nodes: pages, articles, books.

A page is the basic node of Drupal, it contains a title, an author, a body of information, etc.

Articles are more complex pages and books are set collaborative and organized pages.

In addition more types of nodes are provided in extensions of Drupal: images, blogs,...

Users and user profiles

Users are identified accounts in Drupal by means of a username and a password.

Each user has one or more profiles.

A profile is a name who identifies a set of permissions in Drupal, for instance, an Anonymous User can read nodes but nodes can be sent and modified by an Administrator. Although there exist some predefined profiles, more profiles can be added depending on your application.

The main utility of profiles is to define who is allowed to send, modify, publish and delete nodes. Also profiles define who administers the Drupal features.

The main lifecycle of nodes

Nodes are created by an Authorized User (AU) and are administered by an Administrator (AD). The normal phases of this process are:

1. Creation: the AU creates the node and send it to the Drupal server.
2. Publication: the AD revises the node and publish it, i.e. make it visible for unauthorized Users.
3. Navigation: unAuthorized Users (internauts) read the content.
4. Obsolescence: the AD deletes or un-publishes the node.

This is a very general life cycle which can be changed by the Administrator depending on the site requirements.

More on semantics

Usually is an Administrator (AD) who creates the taxonomies and decides what taxonomies are associated to each type of node. When a new node is created by an Authorized User (AU), he classifies the node using the associated taxonomies (i.e. the AU decides that a node is a Seminar)

Navigaton in Drupal

Drupal provides different ways of navigating the nodes contained in a site:

1. URL's: information contained in a node could be an hyperlink. Also a menu item can point out to an URL.
2. Taxonomies: using the taxonomies the nodes can be retrieved in groups (i.e. the Courses, the Seminars, etc.). Although this is a particular case of 3.
3. Trough Drupal paths: a Drupal path is a valid URL for Drupal (i.e. node/1 is the Drupal path for the first node, taxonomy/term/1 is the Drupal path for all the nodes which contains the taxon 1).

How Drupal works: Architecture of Drupal

Basically Drupal is a set of PHP modules which stores and retrieves nodes in a relational database (usually MySQL). A module is a set of php functions which have functional relationships, for instance, the module which manipulates nodes is called node.module, the modules for users is the user.module

The modules in Drupal follows an object-oriented like approach, although they are implemented in a classical PHP approach. Each module contains a set of functions which is called a HOOK. A hook is a PHP function that is named foo_bar(), where "foo" is the name of the module (whose filename is thus foo.module) and "bar" is the name of the hook. Each hook has a defined set of parameters and a specified result type.

Examples of hooks are:

- hook_access: define access restrictions
- hook_auth: Verify authentication of a user.
- hook_help: Provide online user help.

I you want to extend Drupal with new functionality, you have to define a new module (i.e. school.module) which implements some hooks. When Drupal wishes to allow intervention from modules, it determines which modules implement a hook and call that hook in all enabled modules that implement it. A complete description of hooks in:

<http://drupaldocs.org/api/head/group/hooks>

From a more abstract and functional point of view the following layers can be distinguished: <http://drupaldocs.org/api/head>

1. The database interface: set of functions which interacts with the database.
2. The menu (path) system: functions which manipulates the Drupal paths.
3. The form generation system: functions which generate and manipulate forms.
4. The File upload system: functions which manipulate files.
5. The Search system: functions for searching.
6. The Node access system: determines who can do what to which nodes.
7. The Theme system: functions that display HTML, and which can be customized by themes. All functions that produce HTML for display should be themeable. This means that they should be named with the `theme_` prefix, and invoked using the function `theme()` rather than being called directly. This allows themes to override the display of any Drupal object.

Installing Drupal

You should have a functional installation of Drupal in order to understand the following paragraphs.

The installation of Drupal is straightforward. You should follow the steps in <http://drupal.org/node/260>.

Although the installation can be made in any directory we recommend that you create an account in your operating system and install there. Let's suppose the directory for this installation is `/home/drupal`.

Do not forget to create a link (i.e. example) in your www directory (`/var/www`) pointing to the drupal installation directory (`/home/drupal`) and a subdirectory of `/sites`. Let's suppose the url for your new installation is: <http://servername/example>

After installing Drupal, login in as administrator and create two nodes (two pages for instance).

More on Drupal navigation and Drupal paths

You can reach the pages created in the previous paragraph using this URL's:

<http://servername/example/?q=node/1>

<http://servername/example/?q=node/2>

The Drupal paths are: `node/1` and `node/2`. What is a Drupal path? A Drupal path is a call to a Drupal function with some additional arguments. In the previous examples the function called is `node_page` with arguments 1 and 2.

The paths in Drupal are defined by each module in the `hook_menu`. For instance the module `node.module` defines its own paths in the `node_menu` function. (Edit `/home/drupal/modules/node.module` and search for `node_menu`).

The paths should be unique and the menu system registers them in order to make them visible. As we know there are no lists of all the paths in Drupal. Sometimes the module documentation or the module help explain these paths by most of the times you have to read the `hook_menu`.

In <http://drupal.org/node/31644> some more information.