

Database Driven Web Development with Joomla

Courseware

By Pia Waugh
19th June 2006

pia@linux.org.au

Table of Contents

Why Database driven Websites?.....	3
Examples of Database driven websites.....	3
Why Open Source for a Web Developer?.....	3
Why Standards Matter.....	4
Understanding CMS's.....	4
CMS Pros.....	4
CMS Cons.....	5
When to use a CMS.....	5
Planning out your CMS.....	5
Goals.....	5
Users.....	6
Content.....	6
Browsability.....	6
Content types.....	7
Joomla Setup.....	7
Installing Joomla.....	7
Personalising the website.....	7
Look and feel.....	8
Components and Modules.....	8
Your User Permissions.....	8
Bling.....	8
Joomla Administration.....	8
Backups.....	8
Disaster recovery.....	9
Upgrading Joomla.....	9
Website monitoring and optimisation.....	9
Content and software licences.....	10
Community and OSS.....	11
Template tweaking.....	11
Changing the CSS.....	11
Changing/modifying templates.....	11
Accessibility.....	11
Real world website development exercise.....	12
An e-commerce site.....	12
An NGO.....	12
Resources.....	13
Open Source CMSs.....	13
Joomla resources.....	13
Joomla Tutorials.....	13
Reference websites.....	13
Licensing.....	13

Why Database driven Websites?

Database driven websites offer a clean and highly functional approach to web development. Information available in a database can easily be accessed by other applications meaning the functionality possibilities are endless. It also makes backups simple and by building on top of existing functionality you can get on with what you do well, web development!

Examples of Database driven websites

There are many CMS' you can choose from. Below is a list you should play with to find what suits your requirements and personal style.

- Joomla
- Drupal
- Typo3
- Plone
- Postnuke

You can also use the Mozilla Firefox web developer tool bar plugin, which will help you with web development.

If you prefer crafting a website by hand and uploading it, Dreamweaver style, you could try **NVU**. The **GIMP** and **Inkscape** are good for graphics, and **Firefox** is a good relatively standards compliant web browser you can test against.

Each of these projects are Open Source and have a community you can participate in to learn, and contribute.

Why Open Source for a Web Developer?

Using Open Source software such as Joomla, Apache and other applications you can be more competitive in an already competitive market. By basing your websites on free products that have large developer and user communities you can keep your prices down, take advantage of the community maintenance, and participate in the software projects to gain valuable experience, contacts and work opportunities as well as the chance to keep up to date with updates to the software you choose to use. There is also a huge amount of web developer resources available freely to you, so you can cherry-pick the best possible tools and code to create your own unique service.

Why Standards Matter

Web standards such as HTML are vital to ensuring your audience can access your website in the way you want it to look, regardless of the web browser or operating system they use. Adhering to W3C (<http://w3c.org>) standards is very important for the following reasons:

- maintaining the look and feel you have worked so hard to create, regardless of the browser or operating environment (Mac, Windows, Linux)
- ensuring vision impaired individuals can browse your website (this is very important, particularly if you ever want to get Government work)
- ensuring the future-proofness of your work. You don't want a website to break just because a new version of IE has come out
- You want your site to work on mobile devices too!

You can check the validity of your code by putting your website into the <http://validator.w3c.org/> website.

Understanding CMS's

A Content Management System (CMS) offers some really compelling benefits to a web developer. There are also some things you have to compromise on when using a CMS so let's explore the pros and cons and when to use a CMS.

The majority of CMS's are not actual applications, just a bunch of scripts talking to a database.

CMS Pros

- Non-technical web contributors – a CMS typically makes it easy to contribute information through a web interface. This gives non-technical users the ability to publish data, ie – you don't end up doing all the publishing!
- Consistency – across a website it can be consistent without too much effort. Most CMS's use a CSS+HTML templating approach.
- Functionality – easily available, and you don't have to create it.
- Available custom look and feel – There are loads of templates available for tweaking or using as is.
- Speedy publication – Very fast to get a website going as you stand on the

shoulders of giants rather than building everything from scratch

- Typically quite easy to back up
- “Clean” code

CMS Cons

- Tricky to “craft”
- You are not a unique snowflake – hard to not look like a CMS
 - drupalsites.net
- Resource cruncher – because it requires some underpinning applications

When to use a CMS

- Need something in a rush
- Need certain functionality
- Need to ensure non-technical users can publish easily
- Want to maintain lots of websites easily
- Want to offer CMS functionality in a web hosting business

Planning out your CMS

Goals

What are the real goals of your website? What is the company/organisation message and existing branding? What will make your employer feel the investment in your skills has paid off? Think about possible goals like:

- Informing the masses
- Selling tickets to something
- Donations
- Participation in forums
- Organisational due diligence

Messaging might include things like:

- “Soccer – the real football”
- “Bringing medicine to you”
- A particular colour scheme or logo/icon set

You'll most likely need to consult with your employers marketing department to get their messaging correct.

Users

Who are the users of your website? First categorise them into web **users** and web **contributors** which will assist us when we set up our security later on.

Categorise your **users** into types of users. Eg – for a community organisation website, you might have members, newcomers, sponsors, sister organisations, press. Try and figure out the sorts of questions your users might have in coming to your website. We'll use this to then figure out what information needs to be directly linked, and how we can group the information.

Categorise your **contributors** into groups, depending on what you want them to be able to do. Administrators are a given, but these are not necessarily your content creators. You may need different groups for different parts of the business (marketing, sales), or you may need just a group of content creators, who have certain rights but not all. This ensures all of your contributors have the correct permissions to achieve what they need without permissions to parts of the system they don't need.

In some companies you need to setup **workflow** systems, which basically means a structure of approval and content collaboration through multiple people/groups before going live to the web.

Content

First brainstorm on the information you need on the website, eg - information about the organisation, latest projects, case studies, contact details.

Then make a mind map of the information you need. Large pieces of butcher paper are highly recommended!

See if you can create two categories of links that will appear from the main page. Functional information (eg – contact us, about us, aggregator, blog) and contextual information (specific to the website, eg – safety around the home, About OSS).

So there are four steps here:

1. Brainstorm
2. Prioritise (according to your goals and users)
3. Cluster
4. Split into functional vs informative information (website stuff vs informative)

Browsability

Do a little role playing game where you put yourself in the mindset of your

different **users** and see whether the website would answer their queries quickly and easily. You don't want them to have to click through **any more than 2 clicks**, otherwise you lose them.

Test your mind map on other people, see if it makes sense to them.

Content types

What **type** of content do you need to put on the website. This will help you determine the functionality you require.

- Newsfeeds?
- Newsletters?
- Text?
- Photo Gallery?
- Online Store?
- Languages?

Joomla Setup

Installing Joomla

There are a few ways to set up Joomla. Joomla requires a few things, mainly MySQL (a database), PHP (the language it is written in), and Apache (a web server). Joomla itself is basically just a bunch of PHP scripts that interfaces with the database and webserver to give you your site. Your install options include:

- Joomla standard – simply install the requirements on your Linux server, and then unpack Joomla (which can be downloaded from joomla.org) into where you want the website to sit. Create a database for Joomla in MySQL. Setup Apache to point to your joomla folder and then browse to the site. You'll need to put in a load of information to the web browser including your MySQL database name, username and password, then it will set itself up. There are great install guides available at <http://help.joomla.org/content/category/15/99/132/>
- Joomla Standalone server – this bundles Joomla and all its dependencies into a single install suitable for Microsoft Windows

Personalising the website

You can change the settings of your Joomla to ensure the defaults are aligned with how you want your website to work. Check out Global Configuration and Contacts to start.

Look and feel

To start, find a template that reasonable suits your mind-map. We'll play with this later so don't worry about getting something too perfect. Check out the links in the references at the end of the document.

Components and Modules

Find the components and modules you need according to your "Content Types". Install the ones you think would work, and uninstall them if they don't suit.

- **Components** – functionality you need for the website, eg - newsletter
- **Modules** – front page interfaces to your component functionalities

Please note – do not uninstall anything that was already there! You can always unpublish which is better than unintentionally removing something needed.

Your User Permissions

According to your **contributors** information, have a look at the default user groups and see whether they meet your requirements.

Also, you'll find that by default registered users can edit some webpage content when they are logged in to the main website. They'll see an "edit" icon that gives them permission. Play with setting up usergroups with these permissions to give your users easy access to updating webpages without having to use the administrative backend.

Bling

How are you going to attract people to your site? Differentiate yourself? You could try some of the following tactics:

- A free and useful information pack download
- Games
- Interactive sessions with peoples' heroes
- Competitions

Joomla Administration

Backups

You can backup Joomla a variety of ways:

- Joomla backup components
- Directory backups (for web structure)

- MySQL database backups (for web content – you'll do this more with Paul next week)

Because the content is in the database you can simply backup the database.

Disaster recovery

There are a few ways to ensure fast recovery or failover

- A staging site/server
- Recent backups kept online
- Offsite data backups and a DR plan

Upgrading Joomla

Upgrading Joomla is actually quite simple. You simply extract the latest version in the same place as the existing file. Basically you are not upgrading an application, just a bunch of PHP scripts.

Follow these basic steps:

1. Backup your entire website (database and site directory)
2. Download the latest version of Joomla
3. Upload this to your server and extract in the Joomla directory
4. Run the website
5. Done

Website monitoring and optimisation

Monitoring tools are useful to see what is popular on your website, what needs improving, and what already works well. I would suggest both resource and webpage monitoring.

Resource monitoring

Resource monitoring means monitoring the components of your server to make sure it is coping with the load. You can monitor CPU usage, RAM, Hard disk, network bandwidth and more. NAGIOS is a great tool for this.

Webpage Monitoring

Monitoring what is popular on your website, what breaks, and determining how people are coming across your site and where they are leaving it can all help you to optimise the website to achieve whatever it is supposed to (sales, information, contact, whatever). You can also monitor the types of users you

get, how long they stay on your website, where they come from and what browsers they use to optimise the users' experience. Joomla has some components for doing statistics, and some of this monitoring, however some great third party tools include **AWStats** which you can install on your server, or **Google Analytics** which you can register for using your Google logon. There are many other tools out there but these are pretty good.

Content and software licences

Not many people think about content licensing, but it will be important to your employers that you can understand the different types of content licences, particularly when using free material. Understanding free software licences (like the GPL) is also important so that you can articulate what it means to your work if you are ever asked.

With both sorts of licences you can put them somewhere on a continuum between Open and Proprietary.

Content licences

There are many ways to licence data such as content, photos, music or video. By understanding your options you give yourself and your employers the choice to take advantage of more open licences than the traditional copyright. The main one to know about is Creative Commons (<http://creativecommons.org/>) which gives a range of licences from completely open to completely closed.

Software licences

Open Source Software licences generally have a few basic rules you need to remember:

- You have the right to use, copy, redistribute, modify Open Source Software
- If you make modifications to the Open Source project (code) and redistribute the final product, you must contribute your changes back to the community it came from (not always the case)
- You can't discriminate against people or fields of endeavour, either technologically or socially

So practically, If you write code for Joomla and onsell that product, you have to contribute your changes to the Joomla project. If you make code changes but only use them inhouse, you don't have to send your changes back upstream.

Open Source and Open Content licences generally have a positive impact in creating a larger and better quality pool of publicly accessible information and code. Certainly worth thinking about.

Community and OSS

Community is a word I've thrown around a bit, and it is worth mentioning that the best part of Open Source Software (OSS) is that there is a huge global and non-discriminating (mostly) community that you can participate in. You can benefit from the many contacts, learning opportunities and work opportunities that come from being involved in a technical community as well as have the opportunity to contribute yourself and thus have a long term positive impact. Ways to participate are:

- Find a usergroup that aligns with your interest/s
- Join mailing lists
- Read blog or news aggregators
- Blog about your interest and get your blog added to a community aggregator
- Attend meetings and conferences (conf.linux.org.au and osdc.com.au)
- Participate in forums

Template tweaking

Changing the CSS

The CSS is the backbone of the templating for Joomla and many other CMS's. Make a copy of what is in the CSS by default, and then try tweaking it to make it useful for your purposes.

Changing/modifying templates

Try uploading a new template and playing with it. See how it changes the entire website. Try making your own template. Follow this URL for some in depth instructions. <http://dev.joomla.org/content/view/1136/79/>

Accessibility

Why is accessibility important?

- You don't want to limit your audience
- You want to be in line with Government standards for many contracts
- By adhering to standards you are making a more consistent web experience
- You want to ensure it works on mobile devices

Real world website development exercise

Pick one of the following real world examples and create a basic website with the requirements below and some dummy data of your choice.

An e-commerce site

Choose from one of the following:

- a childrens party company
- a pet store
- an online computer store

Provide a way to purchase goods online, a look and feel appropriate to the audiences you expect, and some information about the company you make up. Be imaginative! You'll probably want customer references.

An NGO

Choose from one of the following:

- A wildlife conservation organisation
- A parenting resource
- A human rights organisation

Provide information about the organisation, recent projects, mailing lists, forums and other tools that are useful for creating an active community. Try and make this resource as accessible as you can.

Resources

Open Source CMSs

<http://joomla.org/> - the software and links to other Joomla information

<http://typo3.org/> - TypO3

<http://drupal.org/> - Drupal

<http://plone.org/> - Plone

Joomla resources

<http://joomla.org/> - the software and links to other Joomla information

<http://help.joomla.org/content/category/15/99/132/> - install guide

<http://www.joomlatribune.com/> - news from the Joomla world

<http://www.joomlahacks.com/> - hacks for joomla

<http://www.joomplates.com/> - paid for templates

<http://www.joomla-templates.com/> - more templates, paid and free

<http://joomlateam.com/> - free templates and more

<http://www.joomlastyle.com/> - free templates

<http://www.joomlashack.com/> - templates, components, services and tutorial

<http://onestopjoomla.com/> - great components with no compulsory registration!

Joomla Tutorials

<http://dev.joomla.org/content/view/full/1136/79/1/4/> - Excellent templating tutorial

<http://www.teachmejoomla.com/>

<http://www.joomlashack.com/articles/joomla-tutorials/>

<http://www.netshinesoftware.com/joomla-tutorial.html>

<http://www.siteground.com/tutorials/joomla/>

Reference websites

<http://ask-oss.mq.edu.au/> - the Australian Special Knowledge for Open Source Software. Loads of information about Open Source and licences

<http://www.paintedfoot.com/> - a beautifully done website

<http://linux.org.au> – a reference site for the Australian Open Source community

Licensing

http://www.fsf.org/licensing/licenses/index_html – The Free Software Foundation text

<http://opensource.org/docs/definition.php> – The Open Source definition