

# NextPress

## Documentation Document

NextPress is a new Content Management System which aims to be faster, lighter, and most importantly, MORE SECURE than other available options. To accomplish these lofty goals, NextPress is built on the MVProc platform. For those not familiar... ok, very few are familiar... check out MVProc on SourceForge. Here I'll lay out the basic points.

- Faster – MVProc, an Apache module written in C, processes requests with one procedure call to MySQL, then fills cached templates. The templates are deliberately simple, but surprisingly versatile.
- Lighter – After 400 days of uptime at Free-Blog.net, the apache2 and mysqld processes in total are using less than 100Megs of memory, and while serving over 120,000 pages in that time, the load average is 0.01!
- MORE SECURE – Where to begin? The webserver only needs EXECUTE privileges for a database that doesn't even need to hold the data. Think about that – if the webserver gets hacked, the intruder can still only accomplish the same thing as url hacking with a browser. Also, since EVERY SINGLE PIECE of user input is run through `mysql_real_escape_string`, there is never a concern about SQL injection.

But enough about MVProc, right? Let's install NextPress.

### Installation

In the `etc/` directory, you'll find `install.sh` – if you're running Ubuntu (or a variant that abides by the same directory structure and naming conventions) and are fine with the default install, it will suffice. Otherwise, it's an excellent guide for installing NextPress in whatever way you prefer. The basics:

- MVProc – this is necessary, so you'll need to download, build, and install it. It's free at SourceForge: <https://sourceforge.net/projects/mvproc/files/> Installation is covered more than adequately in `MVProcDoc.pdf` (or the README)
- MySQL is required, and Apache is recommended – running NextPress with MVProc-FCGI has not been tested yet.
- ImageMagick convert binary is necessary for image uploads, but NextPress can be installed and run without it.
- `lib_mysqludf_nextpress` (included) – This is the library of functions that facilitate sending emails, copying files, converting image files, and scanning the directory of page templates.
- `nextSetup.sql` – this file sets up the 3 databases (`nextData`, `nextPublic`, and `nextAdmin`), the account the webserver will use, and the permissions for that account.
- `nextData.sql` – `nextData` is the database where the tables and restricted functions and procedures live. The webserver has NO ACCESS to this database, except through executing procedures in `nextPublic` and `nextAdmin`.
- `nextRestricted.sql` – these are the functions that cannot be executed directly by the webserver.
- `nextPublic.sql` – these are the procedures that can (if you like) be executed over port 80. They do not require session cookies, and only output publicly available data. The public site can, of course, be served over SSL, but you'll likely have to add a subdomain to your admin site.
- `nextAdmin.sql` – these are the admin procedures. They DO require a session cookie, and should NEVER be served without encryption.
- `www/public` – this is the webroot directory. `install.sh` will copy this directory to `/var/nextpress/public` and change the media directory's ownership to the mysql user (to facilitate uploaded media files). Also,

all files are made readable. Set up this way, the webserver HAS NO ABILITY to alter the files.

- `www/templates` – this is the template directory. By default MVProc is directed to cache all the templates (`mvprocCache Y`), so any changes made here should be followed by a reload of apache. Restart will also work, but isn't actually necessary. Within templates are `admin` (templates for the admin interface), `admin/email` (templates for Registration, ForgotPassword, NewArticle – more on this later), `public` (just the layout template for all pages served), `public/pages` (these are the actual pages – more on this later), and `public/dropins` (more on dropins later, too). The structure of the templates directory should not be changed, and should reside OUTSIDE the web root. (`install.sh` puts it in `/var/nextpress/templates`)
- `nextPublic.conf` and `nextAdmin.conf` – these are basic virtual server configurations for NextPress. For more on the mvproc directives, refer to `MVProcDoc.pdf` in the latest MVProc release. These virtual server files should definitely be reviewed prior to activating them, especially if there are other virtual servers running on the server.
- `apparmor` and `SELinux` – everybody's favorite source of problems. The `etc/nextapparmor` file may work on your machine (copied into `/etc/apparmor.d/local/usr.sbin.mysqlld`), but `install.sh` simply disables the `mysqlld` profile, as no matter how much permission I granted the test machines, `apparmor` still denied access to the files and executables I explicitly allowed. `SELinux` will require audit and compilation of rules... or turning it off completely.

At this point, NextPress should be running. If you have any issues, please leave a comment on the Installation Woes page at [nextpress.org](http://nextpress.org), and I'll answer as soon as possible, eventually compiling a Frequently Encountered Problems document.

## The Admin Interface

By default (`install.sh`), the Admin Site is the SSL site on your server. I'll break it down page by page here, but first a note on how NextPress processes URIs. The order of precedence is: page, article, category, tag. If nothing matches the uri and the uri is an empty string, the default uri (set in the configuration page) will be tried. Failing all of that, the 404Page will be served. Keep in mind, if a page has the same uri as an article, the article will never be seen. Also of note is that if an article is the sole member of a category or tag selected, it will simply be served as if pulled up directly. If a category or tag matches more than one article, a list of teasers will be shown in date descending order (in other words, newest first). Also, articles are attached to their category's parent, grandparent, and also the “root” category, whose uri is ‘--all--’.

More on what NOT to do later.

## Dashboard

This is just a list of admin pages the user has permission to use, with nice little icons.

## Configuration

These configuration options are, in most cases, critical to the way the site works. Or doesn't. Probably most of these items should be left the way they are unless the installation went a different way than `install.sh`

### Site

- Absolute path of the web directory: This should be obvious, but make sure to change this if you installed to a different directory.
- Absolute path of the template directory: See above.
- Default uri: If a visitor comes to your website with just the domain (eg. <http://myawesomesite.com/>), the default uri will be used, provided there are no pages, articles, categories, or tags with empty strings for uris. The default install has a `landingPage.tpl` (and a mobile version) with an empty string uri, but this config directive can be used to direct a user to any uri when they land at your site. Lastly, if there is no page, article, category, or tag with the default uri, and the request has no uri, NextPress will display a list

of all articles, most recent first.

- Article Layout Page: This one's critical – this is the page that will be served when an article gets requested. If the page doesn't actually exist, article data will simply be served in XML format. You probably don't want that. The templates/public/dropins/ArticleContent.tpl should absolutely be included (via a `<#INCLUDE dropins/ArticleContent #>` tag) to show the actual content of the article, but the page designated here should also include things like a site banner, navigation menu, related articles, etc. The page provided by default is bare bones on purpose: it shows how to put a dynamic page together. The MVProc template tags (`<# LIKE THIS #>`) should be studied if major changes will be made on a functional level.
- Mobile Version of Article Layout: This is simply the mobile version of the above. More on mobile templates in the **Pages** section.
- Article List layout template: This is the page served to display a list of articles matching a category or tag uri. See Article Layout above.
- Mobile version of Article List Layout: see above and above that.
- Limit to how many articles show on the Article List page: This limit exists so that a site visitor doesn't end up waiting for a thousand teasers to be downloaded. Categories and tags can definitely allow this number to remain small (like 20) and visitors can still reach your content, but if you have many posts which are all about the same thing, you may want to avoid the Article List page altogether and use the Archives dropin on your Article Content page with a landing page (using default uri) that has only the Archives dropin.
- 404 error page: This page will be served when the supplied uri just doesn't match anything at all.
- Date Format: Not as critical, but for those who want to show dates on the public site in a different way, please refer to the MySQL documentation for `DATE_FORMAT`.
- Search engine keywords: What to put in the keywords meta tag on the public site.
- Authors may publish their own articles: If no, an Editor must publish.
- Make notifications of new articles available: This will allow registered users of the site to receive notifications of newly published articles.
- Allow Guests to Register: Obviously guests will need to register with an email address to receive notifications. I think this is the only reason to let them register, but it's possible that could change.
- Cached html for Main Menu: There's no reason to touch this. It's there for efficiency in serving up public pages (rather than having a recursive template gyrating multiple times through a list of categories – that is saved for the admin side).
- From Address for site emails: This is just the “From” address of your email blasts and such. A “[no-reply@myawesomesite.com](mailto:no-reply@myawesomesite.com)” address usually works well. That being said, most email servers will not accept email without a legitimate From, so you will absolutely want to set up an SMTP server, or...
- Domain:port for email server: This is set to localhost:25 because it's the most common way to do it. sendmail or postfix are easy enough to get running. However, you may want (or need) to talk to an off-site server, so here's where you do that. If you need credentialed access to a mail server, like Gmail, there is a more involved way to do that. A future iteration of NextPress will include `lib_mysqludf_curl` to handle the trickier setups.

## Media

- Image conversion directive: This is the directive for images included in articles. See the docs for ImageMagick for all options.
- Thumbnail conversion directive: Any image that gets uploaded also gets a thumbnail version.
- Avatar conversion directive: Just a separate conversion for avatars. You could possibly throw some mogrify stuff in there to fish-eye lens all uploads for this, but maybe not...
- Recognized image file extensions: Do make sure to have '|' before AND after each extension. This is only here because there might be some file types I don't know about. This is the config directive used to tell whether an upload should be converted or just copied and linked.

## Disqus

- Disqus shortname: This would be the id you get when you create a publisher account on disqus.com. Mainly this is included to show how easy it is to include off-site comments, which could also be easily included in the ArticleContent template without using a <# CALL #> to get the shortname.

## Pages

NextPress pages are not articles. As mentioned previously, they supersede articles in terms of uri matching, but they can also be whatever you want. For example, let's say you're running a news site. When visitors come to your site without clicking an article link, you might want to present them with a number of article teasers from various standard categories. You might not want to include the main navigation menu. Maybe you want an entirely different header. Or, let's say you run a typical blog site, but want to have a few very different pages where you offer the books you've written. NextPress makes it really easy to fully integrate these other pages.

First, you write the page in your favorite editor. A page is essentially just the inner html of the <body> of an html page. CSS can be added to the site.css file, and Javascript can be added to the site.js file. The nextArticle and nextList pages are easy examples of how to include “dropins” (dynamic content from the database) in a page. If you decide to create your own dropin, it's easily done with a little knowledge of MySQL stored procedures and MVProc. Extending NextPress is its own section later in this document.

Second, you visit the Admin “Pages” page. At that time, NextPress scans the directory of pages to see what's there. It's on the Pages page that you can assign a uri to a page and publish it. You'll likely want the uri to match a category uri so that a link to the page will show up in the MainMenu.

Third, a note on mobile pages. It's possible that a mobile page will have the same content as its non-mobile counterpart, but with different CSS. In this case, it's not necessary to create a separate page. The included nextList and nextList\_mobile pages are an example of reducing the output for mobile devices (in this case, excluding the article archives).

Some advice on site development: Unlike articles, whenever a page is altered the webserver will need a reload. This is because MVProc (by default for NextPress) caches the pages in memory for speed. During development, it's much more efficient to run the site without caching. (the mvprocCache directive in the Virtual Server configuration set to 'N') This way, you won't need to keep reloading the webserver configuration. It's highly recommended, though, to set mvprocCache to 'Y' for production use.

## Articles

Articles are bodies of text, sometimes with inserted pictures... ok, so we know what an article is. In NextPress there are specific data relationships to work with:

- An article can be attached to only one category. Internally, the article is also attached to the full parentage of categories, including the root category, so the article will appear in the list of the parent category. Authors should choose the most specific category possible that applies.
- An article can be attached to many tags, and a tag can be attached to many articles.
- The article's uri should be distinct from any page or tag uris. If a page has the uri, the article will not be reachable. If a tag has the uri, the article will come up any time every time a link for that tag is clicked, which is probably not what you want. Categories are another story – you may want to create an article that links directly from the Main Menu, rather than a page (discussed previously). For this reason, the article author will be warned if the uri matches anything else.
- Articles must be published to be served to visitors. The Editor role (discussed in more detail later) can edit and publish any article. If the Administrator grants the ability for authors to self-publish, they can publish their own articles (but not anyone else's).
- Inserting Images by using the “Insert Image” toolbar link is the preferred method, as this allows the server to resize the image before it gets inserted into the article.

## Users

Aside from the usual email address, screenname, avatar, etc. NextPress roles are quite specific:

- Administrator – this role has sole access to Configuration, Pages, Users, Categories, Mail Errors, and Import from WordPress pages on the admin site. Aside from User maintenance and possibly Categories, these would be the kinds of edits you'd only make while developing a site.
- Editor – this role has permission to edit, publish, and unpublish any article, regardless of who wrote it.
- Author – this is the role for article authors.
- Subscriber – this is a role specifically designed for site visitors so they can receive email notifications of new articles.

Any user can have multiple roles. Let's say you run your own blog and write all the articles. You would want to assign Administrator and Editor roles to your user account to have full access to what you need to do. You may also want to be a Subscriber so you can see what the other Subscribers are receiving when new article notifications are sent.

On the Users page, the Administrator can prohibit any user. A prohibited user can't log in or receive any notifications, but could still visit the public site.

## Categories

Much has already been said about Categories, but for reference here is a summation:

- Categories drive the MainMenu (the limit of 3 levels is only enforced by configuration of “nestable” in the admin.js file – if you want more than three, you'll need to adjust the CSS in site.css as well).
- If a category shares a uri with a page, that page will be served when the category is selected from the menu.
- If a category shares a uri with ONE article, the article will be served when the category is selected.
- If a category shares a uri with MULTIPLE articles, a list of article teasers will be served when the category is selected.
- If a category shares a uri with NO articles or pages, the article list page supplied serves a “No matching articles found” message.
- Articles attached to a category are also attached to any parent or grandparent of that category.
- The Configuration item “Cached html for Main Menu” should be left alone, unless you REALLY know EXACTLY what you're doing.
- When the categories are changed on the Categories page, the Cached html is updated, and all article hierarchies are updated. If there are many articles, this can take a while.

## Mail Errors

This simple page just shows errors that occurred with email communications. When looking up error codes, it might help to know that NextPress uses libesmtplib to send mail. Empty or missing templates indicate that either the Template root directory is incorrectly configured or the files are simply missing.

## Import from WordPress

Here is a list of what, and in what order, data will be imported from a WordPress XML export file:

- Authors: Display Name, Email Address, URL, and Avatar (you'll have to migrate the avatar images, preserving their location in the web root).
- Categories: Multi-level categories are absolutely maintained.
- Tags
- Attachments: Again, the actual files will have to be migrated just like avatar images.
- Posts and Pages: In NextPress world, these are both simply articles. Pages don't become NextPress

- pages, so there might be a little work to do after the import.
- Comments are NOT imported, as there is no self-hosted comments system in NextPress. Yet?

## Themes

There is no explicit support for themes at this point. However, development of themes is a really simple matter. It just requires making pages (at least two – one for article content and one for article list – mobile versions might be necessary if css won't cover it.) using the available dropins and creating a site.css stylesheet. Other pages are optional. A nice touch might be a SQL script that sets the new pages for use in the config table.

## Extending NextPress

With a working knowledge of MySQL stored routines and MVProc, creating new dropins can be really simple if no admin interface is needed. Studying the [very simple] dropins in the code base should provide sufficient education.

If you do need to create an admin interface, be sure to:

- Add links to any pages to the admin\_links table
- Attach the admin\_link(s) to roles in the role\_admin\_links table
- If you create new roles, be sure understand the data relationships
- Familiarize yourself with the `checkSession` function!
- Pay close attention to the `ord` field in the config table
- Use `getConfig` and `setConfig` functions
- Study how things are done already, with an eye toward non-destructive addition

## The End

I think that might cover it!

Feedback, inquiries, bug reports, and THEMES are all welcome at

NextPress Discussion - <https://sourceforge.net/p/nextpress/discussion/>

or

NextPress Tickets - <https://sourceforge.net/p/nextpress/tickets/>