

**See also the ShortURL ReadMe in the Docs folder detailing this implementation.**

Please backup your site before installing these patches, since a lot of system files are replaced. The PostNuke 0.76rc4 ShortURL package is rather large, consisting of some 400 files. Most of the changes are drop-in changes that doesn't necessitate updating of modules, but there are some exceptions, in particular the Settings and Polls modules, where you need to first go to the Module list, regenerate, and update. Specific patches for popular 3<sup>rd</sup> party templated modules like AutoTheme and PNphpBB2 are included, but only a limited number of modules have been tested with this package. No changes are made to the database, but it is still a good idea to back that up as well. You have been warned.

This is a package of patches for PostNuke 0.76 implementing a new core ShortURL scheme, designed to gradually replace the current Xanthia output filter hack derived from the theme hack posted by Karateka(1) in 2002. Since then no progress seems to have been made to make PostNuke's URLs more user- and Search-Engine friendly. In fact, the current Xanthia filter hack has regressed, becoming bloated with complex and wholly unnecessary Regular Expression rules, many badly written with duplication and a number of bugs, going from the 15 proposed by Karateka to a massive 89. The implementation in this package still retain the filter for backwards compatibility with older themes, modules and blocks, but has been wholly rewritten and pared down to 24 rules, including a rule to fix all links to be root-relative. As PostNuke is in transition to be fully pnAPI(2)-compliant by PostNuke 0.8, the remaining ones can gradually be removed altogether as themes, modules and blocks are updated.

This particular scheme is experimental and may be tweaked or improved upon.

Up until now, PostNuke URLs have consisted of a file in the site's root with a long convoluted query string that is both user- and search-engine unfriendly, meaning the search engines choke on them and thus doesn't index PostNuke sites well, while making the URLs hard to post to people in email or in forums. For instance, a news link looked like this:

```
/modules.php?op=modload&name=News&file=article&sid=123&mode=thread&order=0&thold=0
```

The advent of the ShortURL hack has seen sites hosted on Apache servers with the URL Rewriting module (mod\_rewrite) enabled get URLs like

```
/Article123.html
```

for the above link, where certain assumptions have been made about the default settings for mode, thread and threshold. A big improvement, but not very descriptive, and it comes at the cost of heavy post-processing of the site's content for links. Also, Search Engines use link keyword relevance in their rankings, and **Article123** doesn't say much about the link, except that it's an article with the id 123.

This package seeks to reduce the reliance on the Regular Expression(3) post-processing for links and introduce more user-friendly URLs that have more relevance for people and search engines alike by using virtual directories to visually distinguish sections of the site by module and

function, such as

```
/Example/View.html
```

and for the News articles introduce **Category**, **Topic**, and **Title** information in the link.

```
/Category/Topic/ArticleXXX-title-of-story.html
```

For instance a news story in the category **Computers** and the topic **Postnuke** called "**PostNuke Shorturls**", you'd have the URL

```
/Computers/Postnuke/Article123-PostNuke-Shorturls.html
```

This is a clear, concise and informative link that tells the user and search engine alike something about the link before going there, while retaining backwards compatibility with links of the old ShortURL scheme. It more closely emulates the way we think and organise information, using the folder analogy where we have a clearly-labelled Computer category folder, under which we have the various sub-categories – Topics – with various articles. In this case, we're using a virtual file anchored by the word "**Article**", clearly identifying it as such, followed by the article number and title.

In this instance I've excluded the News keyword altogether for brevity in favour of the Category and Topic keywords which insinuate News anyway, though there is nothing against being consistent with all the other ShortURLs and having the Module appear first, as in

```
/News/Computers/Postnuke/Article123-PostNuke-Shorturls.html
```

This is for the special case of the core News module though, a more generic method is needed overall for URLs with various unknown parameters passed in the query string. This implementation uses the scheme:

```
/Module/Function-Param1:Value1-Param2:Value2... -ParamN:ValueN.(p)htm(l)
```

where the Query string parameters are tagged onto the virtual filename grouped by colons and separated by hyphens, the idea being to use commonly-used characters we might normally use in a list to make it look as natural and readable as possible. It may be a less-commonly used character than the hyphen is needed, like the tilde ~ character. The extension is not necessary, but used for convenience. The 3 types used are either one of **html**, **htm**, or **phtml**, the latter useful to distinguish when you want to link to real HTML files on the site. The extensions as well as the option to use ShortURLs or not is set in the Settings panel, though I've only offered the option of html and phtml, since frankly the MS DOS-holdover extension htm annoys me.

Older URLs are marked with a + before the Function name, as in

```
/PNphpBB2/+profile-mode:editprofile.html
```

so that the server can translate it correctly. If the directory doesn't actually exist, entering

```
/Example/
```

will redirect to the **Example** module main page (Apache only)

```
/Example/main.phtml
```

which in return gets rewritten invisibly to

```
/index.php?module=Example&func=main
```

Otherwise, if it does exist, the index file of the relevant directory will be opened.

Similarly, with

```
/HTML/filename.html
```

if the file exists, it will be opened, else PN will look for

```
/index.php?module=HTML&func=filename
```

It is still possible to tag on query strings like

```
/ModName/main.phtml?theme=seabreeze
```

or

```
/ModName/main-theme:seabreeze.phtml
```

will both be translated to

```
/index.php?module=ModName&func=main&theme=seabreeze
```

There are any number of possible ShortURL systems, the simplest being to simply chop the URL into virtual directories, like **/News/123/** from the above News example as some do. Xaraya uses a variant of this for news, though it doesn't use mod\_rewrite, so appears like

```
/index.php/news/123
```

Again, this is concise, but contains few meaningful keywords other than the module name **News**. You can combine the two methods for News and have

```
/News/Category/Topic/123/title-of-article
```

which works very well, but loses some of the elegance of the above philosophy, since the latter part breaks up the virtual file into 3 with no anchor words, which is not how we organise information.

For generic URLs, there are a number of methods; for instance Mambo, another CMS(4), use generic ShortURLs like

```
/component?option,com_newsfeeds/catid,5/Itemid,7/
```

for a News URL like

```
/index.php?option=com_newsfeeds&catid=5&Itemid=7
```

where the querystring values are grouped by commas and separated by forward slashes (virtual directories). It is a ShortURL, though in this case not shorter, and doesn't have any useful keywords, other than "newsfeed", and is not very human-readable. For a generic URL, this is somewhat unavoidable, but can be better than that.

This implementation also contain a way to customise ShortURLs on a per-module basis through a file called **shorturls.php** placed in the module folder (see the Example module), such as the News URLs, or 3<sup>rd</sup> party modules like PostCalendar, which instead of the full URL like

```
/index.php?module=PostCalendar&func=view&tplview=&viewtype=day&Date=20050405&pc_username=&pc_category=&pc_topic=&print=
```

with the above generic ShortURLs would be rendered as

```
/PostCalendar/view-viewtype:day-Date:20050405.html
```

but with customised URLs become

The beauty is, though, once we've created the groundwork in the core of PostNuke, any implementation will be fairly easy.

As Karateka pointed out at the time in his article, a problem in implementing friendlier URLs with virtual directories is that all paths in PostNuke are relative, ie relative to the site root folder where index.php is located, and fixing it then would have required extensive changes in the core. That is, a URL like **/Example/view.html** would result in the browser looking for all links relative to its present location, ie in the nonexistant subfolder called **Example**, and subsequently it would fail to find the linked stylesheets, images etc, and all links from the page would similarly fail.

Unfortunately this situation has not changed in the intervening years, but as PostNuke modules are becoming API-compliant, they reference the same system function to build their URLs, so fixing this function and other associated functions to use root-relative links(5) will fix all compliant module URLs. But that leaves all other links, like images, Javascript, and stylesheets. The move to templating with Xanthia (for themes) and pnRender (for modules) is also making it easier, since Xanthia templates use a Xanthia variable to reference the theme's image directory path. So fixing Xanthia and pnRender will fix most paths in Xanthia themes. The exception are stylesheet and Javascript link paths and any links in the theme header, for which new path variables need to be introduced, so some updating of Xanthia themes is required. This makes the transition period to PN 0.8 an ideal time to introduced these changes, since few Xanthia themes have been released so far, and core modules are only just being converted to pnRender.

As well as the ShortURL changes to the core, this package contains an overhaul of the templated Admin modules and some User modules, since I implemented a new Javascript-driven Admin panel tab system that necessitated it. And frankly, the current templated modules are drab and unprofessional-looking, having all their theme styling ripped out, including CSS classes, leaving a basic grey and white look with overly large headings and no theme tables for backgrounds. The overhaul of pnRender allows the use of Xanthia-like colour tags, and the opentable functions have been fixed so that proper themed borders can be used. In fact most of the changes are in fixed templates and module files.

Martin Andersen 7/7/2005

- 1) <http://news.postnuke.com/index.php?name=News&file=article&sid=1804>
- 2) API = Application Programming Interface, a set of standard functions in the PostNuke core to perform common tasks.
- 3) Regular Expression (RegEx): A complex pattern-matching language that can look a bit like a mathematical formula, used in the Xanthia ShortURL filter at `/modules/Xanthia/plugins/outputfilter.shorturls.php`.
- 4) CMS = Content Management System
- 5) Root-relative links: Links relative to the server site root (eg `/nuke/filename.html`), which stays static, as opposed to relative to the present file (eg `filename.html`).

Some screenshots of the tabsystem used in the main PostNuke Admin panel, Settings and Xanthia module admin panel with updated versions of the SeaBreeze and PostNukeBlue themes showing their Admin templates, since screenshots of ShortURLs aren't terribly exciting:

Main administration with the Aqua-themed tabs in PostNukeBlue, with the cursor on the Settings panel.



Same as the 1<sup>st</sup> example, but with the tab theme-override on (set from Modify Config link), with a tabs.css file present in the PostNukeBlue theme. These are pure CSS-based tabs, as seen in FireFox. Other browsers won't render the rounded corners:



Main administration with the Luna-themed tabs in SeaBreeze, with the cursor hovering on the 3<sup>rd</sup> Party tab:

Hello Admin You have no unread messages. [Home](#) : [Settings](#) : [Modules](#) : [Blocks](#) : [User Admin](#) : [Permissions](#) : [Xanthia](#) : [Topics](#) : [Add Story](#) : [Downloads](#) : [Forums](#)

### PostNuke administration panel (0.7.6.0-RC4)

[ [View categories](#) | [New category](#) | [Modify configuration](#) ]

Module configuration updated

System **Content** Utility 3rd Party **Crap modules** Resource Pack

#### Content Modules

ADD STORY AddStory	AVANTE GO AvantGo	CATEGORIES Categories	COMMENTS Comments	DOWNLOADS Downloads	EPHEMERIDS Ephemerids	FAQ FAQ	FORMEXPRESS FormExpress
LEGAL Legal	POSTNUKE pnEncyclopedia	PNFORUM pnForum	POLLS Polls	QUOTES Random Quote	REVIEWS Reviews	SECTIONS Sections	SUBMITNEWS Submit_News
TOPICS Topics	TOP LIST Top List	WEB LINKS Web Links					

The Settings panel with the Default CSS-based tab theme in SeaBreeze.

**postnuke**  
Content Management System

Hello Admin You have no unread messages. [Home](#) : [Settings](#) : [Modules](#) : [Blocks](#) : [User Admin](#) : [Permissions](#) : [Xanthia](#) : [Topics](#) : [Add Story](#) : [Downloads](#) : [Forums](#)

### Web site configuration

[Back to Administration Panel](#)  
[ [On-line manual](#) ]

[Site Info](#) [Locale](#) **[Start page](#)** [News](#) [ShortURLs](#) [Themes](#) [Error Report](#) [Footer](#) [Security](#) [Back-end](#) [HTML](#)

Start page:  (The module index.php is pointing to)

Start function type:

Start function:

Arguments for start function:  Comma-separated. Example: name=value,name2=value2

Xanthia administration with Aqua tabs in PostNukeBlue, with the cursor over the Theme Settings tab

Home : Settings : Modules : Blocks : User Admin : Groups : Permissions : Xanthia : Topics : Add Story : Downloads : Forums

### Xanthia theme engine

[Back to Administration Panel](#)  
[ [View themes](#) | [Configure Xanthia](#) ]

### Theme configuration - PostNukeBlue

[ [Generate configuration cache](#) ]

Content Wrappers **Theme templates** Theme colors Theme settings Block control Block Position Tags

#### PostNukeBlue templates

theme templates New theme template			Module templates New module template			Block templates New block template		
File	Action		File	Action		File	Action	
master.htm	[EDIT]	[Reload template]	admin.htm	[EDIT]	[Reload template]	ccblock.htm	[EDIT]	[Reload template]
News-article.htm	[EDIT]	[Reload template]	Downloads.htm	[EDIT]	[Reload template]	dsblock.htm	[EDIT]	[Reload template]
News-index.htm	[EDIT]	[Reload template]	Forum.htm	[EDIT]	[Reload template]	lsblock.htm	[EDIT]	[Reload template]
News-index2.htm	[EDIT]	[Reload template]	home.htm	[EDIT]	[Reload template]	mainmenu.htm	[EDIT]	[Reload template]