

Codex

Codex tools: [Log in](#)

Template Hierarchy

Languages: [English](#) • [Français](#) • [Italiano](#) • [日本語](#) • [한국어](#) • [Русский](#) • [Español](#) • [Português do Brasil](#) • [中文\(简体\)](#) • ([Add your language](#))

Introduction

WordPress Templates fit together like the pieces of a puzzle to generate the web pages on your WordPress site. Some templates (the header and footer template files for example) are used on all the web pages, while others are used only under specific conditions.

What this article is about

This article seeks to answer the following question:

Which template file(s) will WordPress use when it displays a certain type of page?

Who might find this useful

Since the introduction of [Themes](#) in WordPress 1.5, [Templates](#) have become more and more configurable. In order to [develop WordPress themes](#), a proper understanding of the way WordPress selects template files to display the various pages on your blog is essential. If you seek to customize an existing WordPress theme, this article aims to help you decide which template file needs editing.

Using Conditional Tags

WordPress provides more than one way to match templates to query types. WordPress Theme developers can also use [Conditional Tags](#) to control which templates will be used to generate a certain page. Some WordPress Themes may not implement all of the template files described here. Some Themes use conditional tags to load other template files. See the [Conditional Tags](#) page and ["Query Based" in Theme Development](#) for more information.

The Template File Hierarchy

The General Idea

WordPress uses the [Query String](#) — information contained within each link on your web site — to decide which template or set of templates will be used to display the page.

First, WordPress matches every Query String to query types — i.e. it decides what type of page (a search page, a category page, the home page etc.) is being requested.

Templates are then chosen — and web page content is generated — in the order suggested by the WordPress Template hierarchy, depending upon what templates are available in a particular WordPress Theme.

WordPress looks for template files with specific names in the current Theme's directory and uses the *first matching* template file

Contents

[hide]

- 1 Introduction
 - 1.1 What this article is about
 - 1.2 Who might find this useful
 - 1.3 Using Conditional Tags
- 2 The Template File Hierarchy
 - 2.1 The General Idea
 - 2.2 Examples
 - 2.3 Visual Overview
 - 2.4 The Template Hierarchy In Detail
 - 2.4.1 Home Page display
 - 2.4.2 Front Page display
 - 2.4.3 Single Post display
 - 2.4.4 Page display
 - 2.4.5 Category display
 - 2.4.6 Tag display
 - 2.4.7 Custom Taxonomies display
 - 2.4.8 Custom Post Types display
 - 2.4.9 Author display
 - 2.4.10 Date display
 - 2.4.11 Search Result display
 - 2.4.12 404 (Not Found) display
 - 2.4.13 Attachment display
- 3 Filter Hierarchy
 - 3.1 Example
- 4 Changelog
- 5 Source File
- 6 Related

[Home Page](#)[WordPress Lessons](#)[Getting Started](#)[Working with
WordPress](#)[Design and Layout](#)[Advanced Topics](#)[Troubleshooting](#)[Developer Docs](#)[About WordPress](#)

Codex Resources

[Community portal](#)[Current events](#)[Recent changes](#)[Random page](#)[Help](#)

listed under the appropriate query section below.

With the exception of the basic `index.php` template file, Theme developers can choose whether they want to implement a particular template file or not. If WordPress cannot find a template file with a matching name, it skips down to the next file name in the hierarchy. If WordPress cannot find any matching template file, `index.php` (the Theme's home page template file) will be used.

Examples

If your blog is at `http://example.com/blog/` and a visitor clicks on a link to a category page like `http://example.com/blog/category/your-cat/`: Here is the progression of how WordPress uses the template hierarchy to find and generate the right file.

WordPress looks for a template file in the current Theme's directory that matches the category's ID.

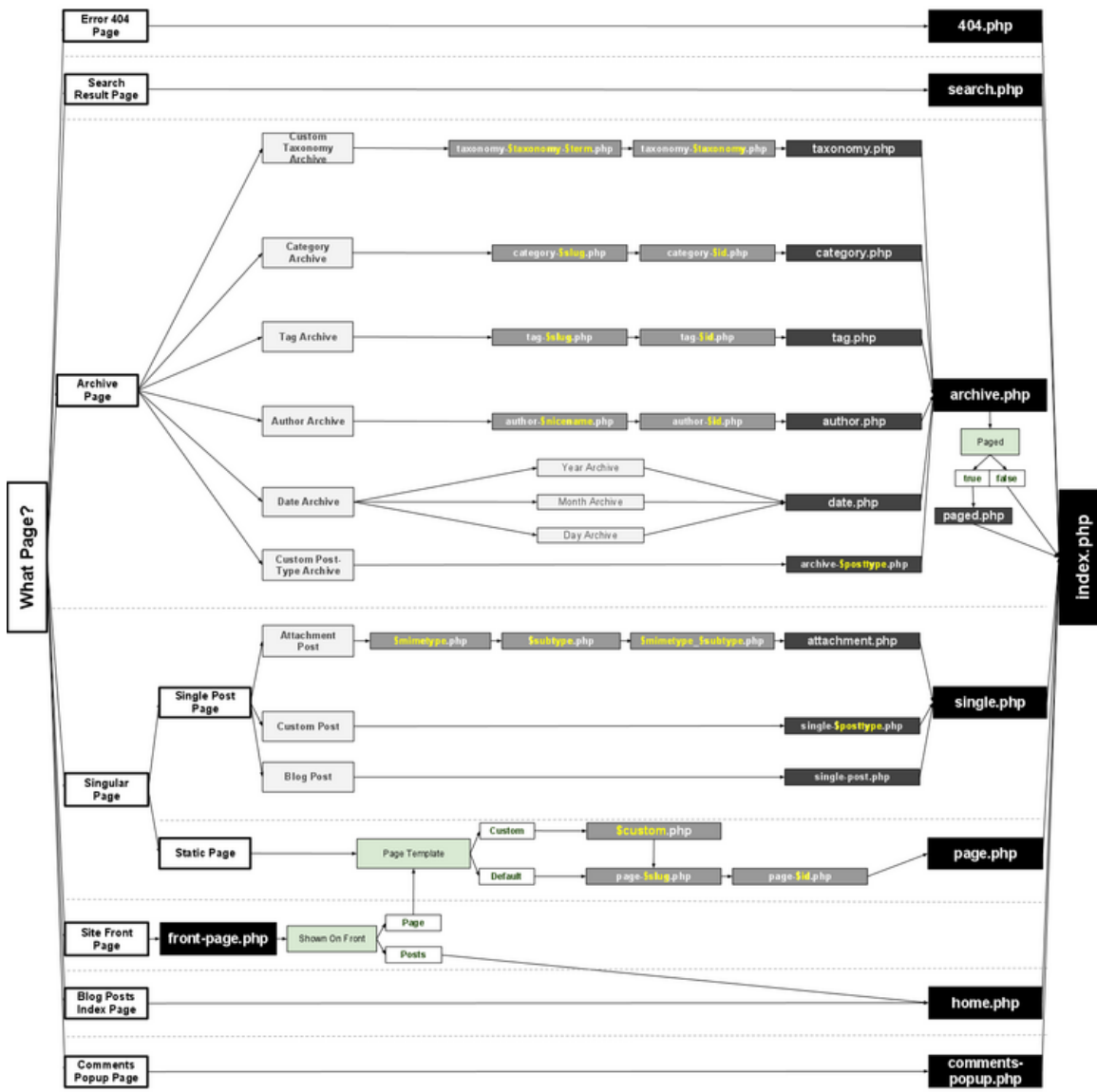
1. If the category's ID is 4, WordPress looks for a template file named `category-4.php`.
2. If it is missing, WordPress next looks for a generic category template file, `category.php`.
3. If this file does not exist either, WordPress looks for a generic archive template, `archive.php`.
4. If it is missing as well, WordPress falls back on the main Theme template file, `index.php`.

If a visitor goes to your home page at `http://example.com/blog/`, the following happens:

1. WordPress first determines whether it has a `static front page`. If a static front page has been set, then WordPress loads that page according to the `page template hierarchy`.
2. If a static front page has not been set, then WordPress looks for a template file called `home.php` and uses it to generate the requested page.
3. If `home.php` is missing, WordPress looks for a file called `index.php` in the active theme's directory, and uses that template to generate the page.

Visual Overview

The following diagram shows which template files are called to generate a WordPress page based on the WordPress Template hierarchy.



[Full-Size Image \(png\)](#)

A more in depth hierarchy diagram, including template-related conditional tags and body CSS classes, can be found [here](#).

WordCamp Minneapolis developed a poster with a more whimsical take on the template hierarchy. [Download the PDF](#)

The Template Hierarchy In Detail

The following sections describe the order in which template files are being called by WordPress for each query type.

Home Page display

Template file used to render the Blog Posts Index, whether on the site front page or on a static page. Note: on the Site Front Page, the Front Page template takes precedence over the Blog Posts Index (Home) template.

1. **[home.php](#)**
2. **[index.php](#)**

Front Page display

Template file used to render the Site Front Page, whether the front page displays the Blog Posts Index or a static page. The Front Page template takes precedence over the Blog Posts Index (Home) template.

1. **front-page.php** - Used for both **Your latest posts** or **A static page** as set in the **Front page displays** section of [Settings](#) -> [Reading](#)
2. [Page display rules](#) - When **Front page** is set in the **Front page displays** section of [Settings](#) -> [Reading](#)
3. [Home Page display rules](#) - When **Posts page** is set in the **Front page displays** section of [Settings](#) -> [Reading](#)

Single Post display

Template file used to render a single post page.

1. **single-{post_type}.php** - If the [post type](#) were **product**, WordPress would look for **single-product.php**.
2. **single.php**
3. **index.php**

Page display

Template file used to render a static page (page post-type)

1. **custom template file** - The [Page Template](#) assigned to the Page. See [get_page_templates\(\)](#).
2. **page-{slug}.php** - If the page slug is **recent-news**, WordPress will look to use **page-recent-news.php**
3. **page-{id}.php** - If the page ID is **6**, WordPress will look to use **page-6.php**
4. **page.php**
5. **index.php**

Category display

Template file used to render a Category Archive Index page

1. **category-{slug}.php** - If the category's slug were **news**, WordPress would look for **category-news.php**
2. **category-{id}.php** - If the category's ID were **6**, WordPress would look for **category-6.php**
3. **category.php**
4. **archive.php**
5. **index.php**

Tag display

Template file used to render a Tag Archive Index page

1. **tag-{slug}.php** - If the tag's slug were **sometag**, WordPress would look for **tag-sometag.php**
2. **tag-{id}.php** - If the tag's ID were **6**, WordPress would look for **tag-6.php**
3. **tag.php**
4. **archive.php**
5. **index.php**

Custom Taxonomies display

Template file used to render the Archive Index page for a [Custom Taxonomy](#)

1. **taxonomy-{taxonomy}-{term}.php** - If the taxonomy were **sometax**, and taxonomy's term were **someterm** WordPress would look for **taxonomy-sometax-someterm.php**. In the case of Post Formats, the taxonomy is 'post_format' and the terms are 'post-format-{format}'. i.e. **taxonomy-post_format-post-format-link.php**
2. **taxonomy-{taxonomy}.php** - If the taxonomy were **sometax**, WordPress would look for **taxonomy-sometax.php**
3. **taxonomy.php**
4. **archive.php**
5. **index.php**

Custom Post Types display

Template file used to render the Archive Index page for a [Custom Post Type](#)

1. **`archive-{post_type}.php`** - If the [post type](#) were **product**, WordPress would look for **`archive-product.php`**.
2. **`archive.php`**
3. **`index.php`**

(For rendering a single custom post type, refer to the Single Post display section above.)

Author display

Template file used to render an Author Archive Index page

1. **`author-{nicename}.php`** - If the author's nice name were **rami**, WordPress would look for **`author-rami.php`**.
2. **`author-{id}.php`** - If the author's ID were **6**, WordPress would look for **`author-6.php`**.
3. **`author.php`**
4. **`archive.php`**
5. **`index.php`**

Date display

Template file used to render a Date-Based Archive Index page

1. **`date.php`**
2. **`archive.php`**
3. **`index.php`**

Search Result display

Template file used to render a Search Results Index page

1. **`search.php`**
2. **`index.php`**

404 (Not Found) display

Template file used to render a Server 404 error page

1. **`404.php`**
2. **`index.php`**

Attachment display

Template file used to render a single attachment (attachment post-type) page

1. **`MIME_type.php`** - it can be any [MIME type](#) (**`image.php`**, **`video.php`**, **`application.php`**). For `text/plain`, in order:
 1. **`text.php`**
 2. **`plain.php`**
 3. **`text_plain.php`**
2. **`attachment.php`**
3. **`single-attachment.php`**
4. **`single.php`**
5. **`index.php`**

Filter Hierarchy

The WordPress templates system allow you to filter the hierarchy. The filter (located in the [get_query_template\(\)](#) function)

uses this filter name: "**`{ $type }_template`**" where **`$type`** is the a file name in the hierarchy without the **`.php`** extension.

Full list:

- `index_template`
- `404_template`
- `archive_template`
- `author_template`
- `category_template`
- `tag_template`
- `taxonomy_template`
- `date_template`
- `home_template`
- `front_page_template`
- `page_template`
- `paged_template`
- `search_template`
- `single_template`
- `text_template`, `plain_template`, `text_plain_template` (all mime types)
- `attachment_template`
- `comments_popup`

Example

For example, let's take the default author hierarchy:

- **`author-{nicename}.php`**
- **`author-{id}.php`**
- **`author.php`**

To add **`author-{role}.php`** before **`author.php`** we can manipulate the actual hierarchy using the 'author_template' hook. This allows a request for `/author/username` where username has the role of editor to display using `author-editor.php` if present in the current themes directory.

```
function author_role_template( $templates='' )
{
    $author = get_queried_object();
    $role=$author->roles[0];

    if(!is_array($templates) && !empty($templates)) {
        $templates=locate_template(array("author-$role.php",$templates),false);
    }
    elseif(empty($templates)) {
        $templates=locate_template("author-$role.php",false);
    }
    else {
        $new_template=locate_template(array("author-$role.php"));
        if(!empty($new_template)) array_unshift($templates,$new_template);
    }

    return $templates;
}
add_filter( 'author_template', 'author_role_template' );
```

Changelog

- **3.1 :**
 - **`archive-{post_type}.php`** added
- **3.0 :**
 - **`front-page.php`** added

- [single-{post_type}.php](#) added
- [taxonomy.php](#) added
- [taxonomy-{taxonomy}.php](#) added
- [taxonomy-{taxonomy}-{term}.php](#) added
- [author-{id}.php](#) added
- [author-{nicename}.php](#) added
- 2.9 :
 - [page-{id}.php](#) added
 - [page-{slug}.php](#) added
 - [tag-{id}.php](#) added
 - [category-{slug}.php](#) added
- 2.3 :
 - [tag.php](#) added
 - [tag-{slug}.php](#) added
- 2.0 :
 - [attachment.php](#) added
- 1.5 :
 - Introduction of the theme system.

Source File

- The template hierarchy functions are located in [wp-includes/theme.php](#).
- The template-load hierarchy is located in [wp-includes/template-loader.php](#).

Related

- Theme Development
- The Loop
- Conditional Tags
- Template Tags
- Using Permalinks

Template Hierarchy: [Category Templates](#), [Tag Templates](#), [Taxonomy Templates](#), [Page Templates](#), [Post Type Templates](#), [Author Templates](#), [Date Templates](#), [Search Templates](#), [404 Templates](#), [Attachment Templates](#), [Loop Templates](#)

[Categories: Design and Layout](#) | [Advanced Topics](#) | [Templates](#) | [Template Tags](#) | [UI Link](#)

[Privacy](#) | [License / GPLv2](#) | See also: [Hosted WordPress.com](#) | [WordPress.TV Videos](#) | [WordCamp Events](#) | [BuddyPress Social Layer](#) | [bbPress Forums](#) | [WP Jobs](#) | [Matt](#)

Like

727k

Follow @WordPress