Coming from WordPress

For WordPress developers, how what you know maps to Exponential. It covers post types and custom fields, themes and the template hierarchy, plugins and hooks, WP_Query, shortcodes and blocks, wp-config.php, WP-CLI and...

Source file: `doc/eng-us/help/cms-developers/coming-from-wordpress.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/cms-developers/coming-from-wordpress.md

This page is for developers who know WordPress and start an Exponential project. It translates the WordPress concepts you use every day into their Exponential counterparts and points at the page that explains each. It compares concepts only; it does not rank the two systems, which are built for different starting points.

Contents

1. The biggest difference
2. Concept by concept
3. Templates: from the template hierarchy to overrides
4. Queries: from WP_Query to fetch
5. Plugins and hooks: extensions, events and operators
6. Configuration and the command line
7. What to unlearn
8. Moving a WordPress site

The biggest difference

WordPress starts from posts and pages and lets you add custom post types and custom fields on top, often through plugins. Exponential starts from a **content model you define**: every kind of content is a **content class** with typed **attributes**, and everything, from a page to an image or a user, is an object of a class placed in one **content tree** (The content model). Expect to spend the first day of a project on classes rather than on a theme.

Concept by concept

In WordPress	In Exponential	Read
Post type, custom post type	content class	Content classes and attributes
Custom field, meta box, ACF field	class attribute of a datatype	Choosing datatypes
Post, page	content object with versions	Versions, drafts and publishing
Page parent, menu order	node in the content tree, priority	Objects, nodes and locations
Category, tag	locations, `ezkeyword` or `eztags`	Relations and embeds

In WordPress	In Exponential	Read
Media library attachment	image or file object in the media library	Images and image aliases
Revisions	versions, with compare and restore	Versions, drafts and publishing
Roles and capabilities	roles with policies, limited by section, subtree, class	Sections, roles and policies
Permalinks	URL aliases made from the tree	URL aliases and routing
Theme	design in a design extension	Templates and overrides
Plugin	extension	Extensions you will write
Shortcode	custom tag in rich text	Rich text
Block editor layouts	Exponential Layouts (zones and blocks)	Layouts for CMS developers
Multisite network	siteaccesses on one installation	Siteaccesses and multisite
REST API	the REST interface of `ezprestapi`, and `expservices`	The REST API

Templates: from the template hierarchy to overrides

WordPress picks a PHP template file by a naming hierarchy (`single-{post_type}.php`, `page-{slug}.php` and so on). Exponential's templates are written in its own template language (`.tpl`), not PHP, and a template is chosen by **override rules** in `override.ini`: "for `node/view/full.tpl`, when the class is `exp_event`, use `full/exp_event.tpl`". Rules can match the class, the node, the parent, the section, the depth, the view mode and more. The template language escapes output with `|wash()` and has no direct access to PHP, which keeps templates safe to hand to designers. Start with Templates and overrides for CMS developers; the language itself is in The template language.

Queries: from WP_Query to fetch

Where a WordPress template calls `WP_Query` or `get_posts()`, an Exponential template calls a **fetch function**:

Template

```
{def $events = fetch( 'content', 'list', hash(
        'parent_node_id', $node.node_id,
        'class_filter_type', 'include',
        'class_filter_array', array( 'exp_event' ),
        'sort_by', array( 'attribute', true(), 'exp_event/start_date' ),
        'limit', 10 ) )}
```

Fetches check permissions and hidden nodes for you. Details: Fetching content in templates; in PHP: Content from PHP.

Plugins and hooks: extensions, events and operators

A WordPress plugin hooks into actions and filters. An Exponential **extension** is a folder under `extension/` that can add settings, templates, modules (pages with their own URL), template operators, datatypes, workflow events, cronjobs and console commands. Instead of one hook

system there are several precise extension points:

WordPress habit	Exponential
`add_action` on save or publish	a workflow event on the publish trigger, or an event listener (Workflows and triggers, Events, listeners and filters)
`add_filter` on output	a template operator, or an override of the template (Template operators)
A custom admin page	a module with a view and a policy (Modules and views)
`functions.php` helpers	template operators and PHP classes in your extension

The whole list: Extension points at a glance.

Configuration and the command line

WordPress	Exponential
`wp-config.php`	INI files: shipped defaults in `settings/`, your values in `settings/override/` and `settings/siteaccess/<name>/` (Settings and INI)
Options table	INI settings, read and changed with `./console exp:ini`
WP-CLI	`./console` (`bin/php/console`) and the scripts in `bin/php/`
wp-cron	cronjobs run by the system crontab (Cronjobs)
Page caching plugins	view cache, HTTP cache and the Velocity response cache, built in (Caching for CMS developers)

What to unlearn

- **Do not query the database from templates or plugins.** Use fetch functions and the kernel API; they keep permissions, versions, caches and search right.
- **Do not put markup in content.** Rich text is stored as XML and drawn by templates; editors choose structure, not HTML (Rich text).
- **Do not edit shipped files.** Kernel files and `settings/*.ini` are replaced on update; your work lives in your extension and in override settings.
- **Clear the right cache.** Templates, settings and content views are cached; Caching for CMS developers says which cache follows which change.

Moving a WordPress site

The migration guide covers the WXR export, mapping, HTML to rich text, permalinks and redirects step by step: WordPress and Exponential. For the tools that import the result, see Importing content; the trainer's lesson is From WordPress.

Related pages

- Exponential for CMS developers
- Mapping content to Exponential
- Moving from Smarty: if you know Smarty, the template language will look familiar

- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.