Getting started

Last changed 2026-11-12
/exphelp/manual/help-gen-manual-getting-started

Help for editors and users

Where to start: the help pages for people who write, edit and publish content, and for everyone with an account.

These pages explain the everyday work in Exponential: signing in, finding your way around the administration, writing and publishing content, uploading media, translating, and what to do when something goes wrong. They are written for editors and for anyone with an account on a site, and need no technical knowledge.

Every page describes what you see in the administration with the **Admin 4** design, the design most installations use. Other administration designs (Classic, AdminUI) have the same pages and buttons, arranged differently. What you see also depends on your role: an editor sees fewer tabs and buttons than an administrator, and a button you are not allowed to use is shown grayed out or not at all.

Start here

Page	What it helps you do
Velocity is part of Exponential 6.0.15	Know what changed when the Velocity Internet / Web Application Server became part of Exponential: a turning point that makes Exponential one working stack (server, RAD platform, CMS, digital experience and e-commerce)
New in 6.0.15: static publishing	Publish your site free on GitHub Pages: what static publishing brings and where to start
Why Exponential supports PHP 7.4	Know what is usable on PHP 7.4 without support and what needs PHP 8, how Composer picks the right packages for your PHP, how security is kept, and why the project bridges older PHP versions
Getting started	Sign in, take your first steps, and know where to find help
A tour of the administration	Know the header, the tabs, the side columns and the dashboard
Logging into the Editor	Sign in to the Editor, the interface for writing and publishing
Logging into the admin	Sign in to the administration, with lockout, design and safety tips
User manual	Do one task in the administration: one question per page, with who may do it, the steps and what can go wrong

Who makes Exponential

Page	What it helps you do
Who is 7x?	Know the company behind Exponential: nearly thirty years with this code, from the eZ Publish years to Exponential 6, the products, the community and how to take part

Page	What it helps you do
Who is the Exponential Foundation?	Know the nonprofit that operates the open source project, the whole family of Exponential products and what the project commits to
Copyright and copyleft	Know who holds the copyright of Exponential (the Exponential Foundation and 7x together, since the historical shift of 29 October 2026), why the Foundation is named first, what the GPL lets you do, and how to write the notice in your own files and extensions
Our goals	Know why Exponential exists, what is already built in so you do not have to build it, why it is free software, how 7x earns a living, and why we invite you to join, with honest answers to doubts about our intentions
Community	Know the Exponential Community: the people who use, contribute to, promote and develop software using Exponential worldwide
Ecosystem	Know the greater ecosystem beyond the community: every part of the Exponential world within the PHP stack, its users and creators
Partners	Know the Exponential Partners: they contribute financially to the sustained future of the project, contribute code and more back to it, and engage directly with 7x and the Exponential Foundation
The Exponential project	Find the FAQ, the licensing questions, the roadmap, support, the community channels, how to build, contribute and give, as the project website exponential.earth presents them

Everyday work

Page	What it helps you do
Creating and editing content	Create an item, fill in its fields, use the rich text editor, move and copy
Images, media and uploads	Upload one file or many at once, and keep the media library tidy
Bookmarking system pages	Keep Setup, the help, the audit trail or a filtered list in your bookmarks, in folders next to your content
Publishing, versions and the trash	Publish, keep drafts, go back to an older version, hide, remove and restore
Searching	Find content in the administration and on the site
Languages	Translate content, and choose the language of the administration itself
The layouts editor	Arrange the blocks of a page in zones, and publish the layout
Migrating URLs	Keep the old addresses of a site working after a move: collect them, map them, import with a preview, test, undo, export

Accounts, roles and members

Page	What it helps you do
Your account and signing in	Sign in and out, change your password, turn on two-step sign-in
Setting up accounts for editors and administrators	Create an editor's account in the Editors group, check it, and hand it over so that only the editor knows the password; the same for the few administrators
What the Editor role allows	Know what an editor may do, and what stays with administrators
Managing members	As an editor, create, change, disable, move and remove the accounts of the site's members

Page	What it helps you do
The user audit log	Find out who did what to an account, when and from where

Working faster, and getting help

Page	What it helps you do
Keyboard shortcuts	Work with the keyboard alone, in the administration and in these help pages
When something goes wrong	Tell the right people about an error, with what they need to fix it
Getting support	Get free help from the community (forums, Telegram, Discord, Slack) or paid help from 7x
Project files	Read the README, the security policy, the contributors' guidelines and the license of the repository in the help

Grow from user to developer

Page	What it helps you do
Develop with Git	Install Exponential as Git clones with their full history (the recommended way to build websites), keep them up to date, and send fixes back as pull requests
Update with Git	Get updates as they are pushed to GitHub and update Composer to match, hourly, daily, weekly, monthly or only at each release, by hand or automatically
Migrations	Schema and content changes written once as files and applied to every installation: paths for beginners, administrators and teams, 25 recipes, the reference, the FAQ and help when a migration fails
Recent content	The newest content of the site, submissions held for review with who sent them from where, and releasing, hiding or removing a selection at once
Migrations in the administration	See the state of the migrations in Setup > Migrations, run them, follow a run as it goes, and take a run back
Migrations versus migrating a site	Which of the two jobs called "migration" you need: changes of one site, or moving and upgrading a whole installation
Contribute extensions back	Publish your extensions so everyone can use them, and know what the GPL requires when you distribute one
Getting the code	Every way to install the code, with versions, tags and extensions explained

Deeper guides

When a help page is not enough, the guides go into every detail of one area. A few that editors use most: My drafts and my pending items, Versions of an object, Bookmarks, Content languages, Two-step sign-in and The content model and editing content. All guides are listed in the guides' table of contents, and every term is explained in the glossary.

Choosing where Exponential runs

For administrators and developers who choose the machine and the database of a site:

Page	What it helps you do
Database support	Choose a database from the supported ones, in their official order (SQLite 3, PostgreSQL, MongoDB, MariaDB, Oracle, MySQL), and know what each needs
Operating system support	Check whether an operating system, processor and web server can run Exponential, from Linux to Windows and beyond
Page caching with LiteSpeed and OpenLiteSpeed	Make a site on a LiteSpeed or OpenLiteSpeed server fast for anonymous visitors and crawlers: the server answers from stored pages without PHP, with no plugin and no code

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. Start with Partners: start here and How to apply.

Velocity is part of Exponential 6.0.15

As of 6.0.15 the Velocity Internet and web application server is part of Exponential itself, installed with every copy and driven by one console command.

This page is for everyone who wants to know what changed when Exponential 6.0.15 took its own Internet and web application server, **Exponential Velocity**, into the core product: site owners, administrators, hosting companies, developers, decision makers and the press. It needs no technical knowledge to read; the commands are there for those who want to try them.

Contents

1. In one paragraph
2. What changed in 6.0.15
3. Why it is a turning point
4. One stack, not just a framework
5. What Velocity brings out of the box
6. Velocity next to Apache, nginx and FrankenPHP
7. The commands
8. The documented figures
9. What it means for you
10. Questions and answers
11. Read more

In one paragraph

As of Exponential 6.0.15, the Velocity Internet / Web Application Server is part of Exponential. The root `composer.json` **requires** the package `se7enxweb/exponential-velocity` at `~0.0.4.57`; it is no longer only a suggestion. Every installation made with Composer therefore has the server, on every PHP

version Exponential supports (PHP 8.0 and later, including the stock PHP 8.0 of Red Hat Enterprise Linux 9). It is driven by the same console as the rest of Exponential: `./console exp:velocity deploy`, `start`, `restart` and `cache clear`. This marks a turning point for the project, a cornerstone and a capstone at once: a new level of application system integration, in which the core product provides even greater features out of the box, by default.

What changed in 6.0.15

Before 6.0.15	As of 6.0.15
Exponential was a PHP application that needed a web server of someone else's choosing in front of it.	Exponential ships its own Internet and web application server, and still runs behind any other one.
Velocity was a separate package, listed in `composer.json` under `suggest`, to be added by hand.	`composer.json` requires `se7enxweb/exponential-velocity ~0.0.4.57` (through `se7enxweb/exponential-php-floor`, on PHP 8.0 and later); `composer install` installs it.
Velocity needed PHP 8.1, so installations on PHP 8.0 (Red Hat Enterprise Linux 9) had to leave it out.	Velocity 0.0.4.54 runs on PHP 8.0 to 8.5, the same range as the Exponential kernel.
Serving, caching and deploying were separate jobs with separate tools.	`./console exp:velocity deploy` refreshes autoloads, settings, caches, the PHP processes and the server in one ordered run.

Nothing an existing site does stops working. A site served by Apache with PHP-FPM, by nginx or by FrankenPHP keeps being served the way it was; the server is simply there, ready, the day you want it. The shipped `settings/velocity.ini` still starts PHP's built-in server by default (`[ServerSettings] Engine=php`) because that always works; one setting, `Engine=qbix`, switches to Velocity's own server.

Why it is a turning point

For almost all of its history, the software now called Exponential stopped at the edge of PHP. What happened between the visitor and the application (the web server, its process manager, the TLS certificate, the response cache, the reload after a deploy) was assembled around it by each team on its own, and each team learned the same lessons again.

Exponential 6.0.15 closes that gap:

- **A cornerstone.** The server is now part of the foundation the rest of the product is built on. Kernel features are written and tested against it: persistent workers that boot once, a response cache that answers before a worker is chosen, a deploy command that knows which caches and processes a change touches.
- **A capstone.** It completes what the 6.0 line built up: current PHP from 8.0 to 8.5, SQLite as the default database so no database server is needed, six database engines on one kernel, a one-command installer, the console, the audit trail and the help center. With the server included, the whole path from a request to a rendered page belongs to one product, documented in one place.
- **A new level of application system integration.** The application and the server are developed together and tuned against each other rather than merely made compatible. The result is a product that provides even greater features out of the box, in the core, by default.

One stack, not just a framework

Exponential is a whole working unit, not just a framework you have to complete before it does anything. One `composer install` delivers a rapid application development (RAD) stack for Internet applications that is at the same time a content management system, a digital experience platform and an e-commerce system:

Layer	What is in the box	Read more
Server	Exponential Velocity: HTTP/1.1, HTTP/2 over TLS, WebSockets, its own HTTPS certificates, a response cache, static files and images	Velocity web server
Application platform	The Exponential kernel: content classes and objects, versions, roles and policies, multisite, more than 360 locales, REST, cronjobs, the console	README
Rapid development	The RAD tools: a catalogue of extension points, wizards that write a working extension, a survey and a health check	RAD tools
Content management	The administration (Admin 4), the online editor, publishing and versions, the trash, the audit trail	Help for editors and users
Digital experience	Exponential Layouts (a visual page builder with zones, blocks and rules), multisite and multilingual sites, the Exponential Intelligence writing assistant that writes drafts only	The layouts editor
E-commerce	The shop: basket, orders, payment gateways, order receipts and the store dashboard	Store dashboard
Data	SQLite by default; PostgreSQL, MongoDB, MariaDB, Oracle and MySQL on the same kernel	Database support

The Exponential family also includes Exponential Platform DXP, the Symfony-only member, for teams that want the Symfony digital experience platform line; it is a separate product with its own feature set (Who is the Exponential Foundation?).

Because every one of those layers comes from one source, with one installer, one console and one set of documentation, adopting Exponential anywhere in the world comes without the usual integration work: there is no server to choose before the first page, no second vendor's documentation to reconcile, and no PHP version where half of the stack is missing. Exponential runs on almost any operating system that has PHP 8.0 or later (PHP 7.4 usable without support) (Operating system support).

What Velocity brings out of the box

Each of these is a documented, shipped feature of Velocity 0.0.4.x:

Feature	What it does	Page
Persistent workers	The application boots once into a parent process; a pool of workers keeps it loaded between requests.	Persistent-worker server
Response cache	Public pages are answered inside the server process before a worker is chosen, with `304`, `ETag` and stale-while-revalidate.	Response cache
HTTPS that looks after itself	A self-signed certificate when none is configured; your own files, a PKCS#12 bundle, Let's Encrypt or another ACME CA when one is; renewed certificates swapped in without a restart.	HTTPS and certificates
Static files and images	Files served without PHP, validated, compressed, sent in ranges, and images converted or resized on request.	Static files and images
Real time	WebSockets, Server-Sent Events and rooms, built into the server.	WebSockets and events

Feature	What it does	Page
Dashboard and control panel	Live requests, workers and memory at `/Q/dashboard`; a password-protected control panel at `/Q/panel`.	Control panel and dashboard
Q shell	A drop-down console on every server view for status, workers, cache, sites, certificates and logs.	Q shell
Scheduler	A small cron inside the server for periodic work, without a second process manager.	Scheduler
Packages and binaries	System packages, Docker images and single binaries for use outside an Exponential installation.	Packages and binaries
Familiar configuration	An on-disk layout in the style of Debian's `/etc/apache2`: `sites-available`, `conf-available`, `mods-available`, enabled with `exp:velocity site|conf|mod enable`.	On-disk layout

Velocity next to Apache, nginx and FrankenPHP

Velocity does not replace the choice of web server; it adds the best one to the box. Exponential is served by any of these, and `exp:velocity` drives the last two:

Engine	Role
Apache with PHP-FPM, nginx with PHP-FPM, LiteSpeed and the others	Traditional servers. Fully supported, and the answer where a host does not allow a long-running process.
Velocity's own server (`Engine=qbix`)	**The recommended engine for every stage**, from development to production, and the fastest. Serves HTTP and HTTPS itself.
FrankenPHP (`Engine=frankenphp`)	**The production-ready alternative**: Caddy with PHP built in, downloaded as one checksum-verified binary by `exp:velocity install`.
PHP's built-in server (`Engine=php`)	Development only; the shipped default because it always works.

Velocity also runs happily behind Apache or nginx as a reverse proxy, so a host with a tuned front server keeps it. Where several customers share one machine and must not read each other's files, run one Velocity instance per site, each as its own operating-system user. The details are in Velocity engines.

The commands

Shell

```
./console exp:velocity deploy --allow-root-user       # after a change: autoloads, settings, cach
es, PHP-FPM, Velocity
./console exp:velocity start --allow-root-user        # start the configured engine
./console exp:velocity restart --allow-root-user      # restart it, so workers load changed class
es
./console exp:velocity cache clear --apply --allow-root-user  # empty Velocity's response cache
```

`deploy --dry-run` lists what it would do and does nothing else; `deploy --kernel` also rebuilds the kernel autoloads after a kernel class was added or renamed. Workers keep the application in memory between requests, so after a class or settings change the server needs `deploy` (or `restart`), not only a cache clear.

The documented figures

These are the figures already documented for this stack, with their method, in the README: **0.36 ms** of server work per cached page (a median over hundreds of requests), **8,941 requests per second** sustained on a 12-core machine serving a real page with zero errors up to 384 connections in flight, and a returning visitor's reload answered with **202 bytes** instead of the whole page. The README is also clear that most of the speed came from fixing Exponential and its cache, which helps on Apache and nginx too; Velocity raises the ceiling further. How the figures are measured: How Velocity figures are measured.

What it means for you

- **Site owners and decision makers:** one product to buy into, one place to read, one supplier relationship. The server is not an extra project.
- **Hosting companies:** Velocity is installed on every Exponential site and runs on the PHP your machines already have, 8.0 included. Offer it where long-running processes are allowed; keep Apache or nginx with PHP-FPM where they are not. Start at Hosting companies start here.
- **Developers:** develop against the same engine production runs, with the RAD tools for extensions and WebSockets, the scheduler and the response cache available without adding anything.
- **Administrators:** one command deploys a change in the right order and stops at the first failure.

Questions and answers

Do I have to use Velocity? No. It is installed with every copy, and it is the recommended engine, but Exponential still runs under Apache, nginx, LiteSpeed and the others exactly as before.

Does it work on Red Hat Enterprise Linux 9 with its stock PHP 8.0? Yes. Velocity 0.0.4.54 runs on PHP 8.0 to 8.5. It needs the `sockets` extension; `pcntl` and `openssl` are recommended for the worker pool and HTTPS.

Is Velocity or FrankenPHP experimental? No. Velocity is the recommended engine for every stage, and FrankenPHP is a production-ready alternative.

Do I need to change anything when I upgrade to 6.0.15? Run `composer install` as usual; it installs the server. Nothing switches engines on its own.

Read more

- Velocity web server and running Exponential in a persistent-worker web server
- Velocity engines: Qbix, FrankenPHP, PHP's built-in server
- Installation chapter 8: Serving the site
- The 6.0.15 changelog
- Operating system support

Who is 7x?

The company behind Exponential.

7x is the North American company that maintains, develops, supports and hosts **Exponential**, the open source content management system you are using right now. This page tells who 7x is, where the software

came from, what 7x builds around it today and how you can take part.

The page is long on purpose: it covers a history of nearly thirty years. Every statement names its source, either a file in this repository (open it in the help or on GitHub) or a public web page. Where two sources disagree, the page says so instead of choosing one. Where something could not be checked, it is left out.

> **Note**
>
> 7x's sister organization, the nonprofit that funds and directs the open source project, has its own page: Who is the Exponential Foundation?

Contents

1. In short
2. Who 7x is
3. The names: Brookins Consulting, 7x, se7enx and Exponential
4. Nearly thirty years: the timeline
5. The eZ Publish years, 1999 to 2021
6. Stewardship: from a dormant code base to Exponential 6
7. The 6.0.x releases
8. The product: what Exponential is
9. The application stack
10. The extension ecosystem
11. The platform line: Exponential Platform, Legacy, Nexus and DXP
12. Exponential Basic
13. Exponential Velocity
14. Services and hosting
15. The community
16. Open source licenses
17. Principles you can see in the code
18. 7x in numbers
19. How to get involved
20. Sources

In short

- **7x is a company.** It describes itself as "a responsive North American web design, support, development and website hosting company" and as "enterprise content management integrators, implementing the Exponential enterprise content management system" (https://se7enx.com/about).
- **7x leads Exponential.** "7x took over leadership of the project and it's development, support, adoption and community growth in 2023" (README.md, section "Who is 7x").
- **7x used to be Brookins Consulting.** Brookins Consulting was an eZ Publish partner in the USA (

https://se7enx.com/about/exponential-hosting-usa). 7x grew from it in three stages: the idea in 2014, the start of the project in 2022 and its launch in December 2023 (From Brookins Consulting to 7x).

- **Nearly thirty years.** The copyright notice of Exponential reads "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved." next to "Copyright (C) 1999 - 2014 eZ Systems AS" (COPYRIGHT.md).
- **One code base, kept alive.** "Exponential has made it beyond it's end of life in 2021 and survived" (README, "Exponential Project Status"). Since December 2023 7x has published the 6.0.x releases, from 6.0.0 on 1 January 2024 to 6.0.14 on 7 June 2026, followed by the current 6.0.15 line (release overview).
- **Not eZ Systems, not Ibexa.** "This project is not associated with the original eZ Publish software or its original developer, eZ Systems or Ibexa" (README, "Exponential Project Notice : 2025.08.12"; https://exponential.earth/about/project).

Who 7x is

In its own words

Statement	Source
"7x is the North American corporation driving The Continued General Use, Support, Development, Hosting, Design of Exponential Enterprise Open Source Content Management System"	README.md, "Who is 7x"
"7x represents a serious company leading the open source community based effort to improve Exponential and it's available community resources"	README.md, "Who is 7x"
"a responsive North American web design, support, development and website hosting company providing user friendly, interactive cloud-based customized deployments of content management systems"	https://se7enx.com/about
"We are enterprise content management integrators, implementing the Exponential enterprise content management system"	https://se7enx.com/about
"North American stewards of Exponential Platform —the PHP CMS that refused to die."	GitHub profile README of se7enxweb
"7x (Corporation) the helpful website development accelerator" (account bio, location given as Seattle, Washington, USA)	https://github.com/se7enxweb

Besides Exponential, the 7x website lists work with other content management systems: "Exponential, WordPress, Concrete, Winter, ... Plone" (https://se7enx.com/about), and its home page offers Exponential CMS (Basic, Platform, Nexus, Legacy), Winter CMS and Concrete CMS (https://se7enx.com/).

7x and the Exponential Foundation

The copyright holder named in every file of the kernel is "Exponential Foundation and 7x", together since 29 October 2026 (Copyright and copyleft): see COPYRIGHT.md, LICENSE and the header of almost every PHP file, for example `autoload.php`. The documentation carries "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved." (The license of this documentation).

The two are not the same organization. 7x is the company; the Exponential Foundation is a nonprofit corporation, founded on 19 March 2026, which names 7x "the current leader in contributions both financial, code, extension development, new features, infrastructure, maintenance and much much more back to the project" (https://foundation.exponential.earth/). The Foundation's own page explains the rest: Who is the

Exponential Foundation?

The people you meet in the repository

In the git history of this repository, the commits of the stewardship period (November 2023 to today) are authored by the account **7x** (the large majority), by **ekke** and **felix** (whose pull requests come from the public GitHub accounts ekked and fwoldt), and by a handful of others. Run `git shortlog -sn --since=2023-11-01` to see the current list. The person who founded the company is named on the public page https://graham.se7enx.com/7x, which is written in the first person: "I founded 7x in 2001 as Brookins Consulting".

The names: Brookins Consulting, 7x, se7enx and Exponential

Name	What it is	Source
Brookins Consulting	The earlier name of the company. Its GitHub account brookinsconsulting sent pull requests to the eZ Publish code base from 2014 (`git log` shows the first merged one on 26 August 2014) and the first outside change of the fork on 11 December 2023 (`2ddbc4c624`).	README.md; `git log --grep="from brookinsconsulting/"`
7x	The company's name today. Pronounced and written out as "se7enx" in its domain and accounts.	https://se7enx.com/
se7enx.com	The company website, with services, hosting and contact.	https://se7enx.com/
se7enxweb	The GitHub account and the Composer vendor name of every package 7x publishes (`se7enxweb/exponential`, `se7enxweb/expai` ...).	composer.json
Exponential	The product, formerly eZ Publish. Renamed in the repository on 12 August 2025 (`837c7b835d` and the commits around it).	August 2025, The product is called Exponential
exponential.earth	The website of the open source project, with documentation, forums and downloads.	https://exponential.earth/
`sevenx`	7x's official public PHP code prefix (`sevenx_dse`, `sevenx_themes_media`). "7x" cannot be a prefix: a class or identifier cannot start with a digit.	Naming prefixes: sevenx and exp
`exp`	The code prefix 7x uses strategically, with intent, to build the Exponential brand and to assign ownership of an extension or solution to the Exponential project and the Exponential Foundation (`explayouts`, `expai`, `expqueue`). Extensions and classes from 7x use `sevenx` or `exp`, as 7x needs.	Naming prefixes: sevenx and exp

From Brookins Consulting to 7x, in three stages

Brookins Consulting was founded in 2001 (https://graham.se7enx.com/7x). 7x did not appear on one day: according to 7x's own account (the owner's statement of 18 October 2026, for this page), it grew in three stages.

Stage	Year	What happened	Sources that mark it
The idea	2014	The initial idea, started quietly: the se7enx.com domain was registered and the fork of the entire ecosystem, every resource of it, was prepared.	7x's account; "I founded 7x in 2001 as Brookins Consulting (rebranded in 2014)" (https://graham.se7enx.com/7x)

Stage	Year	What happened	Sources that mark it
The start	2022	The project started.	7x's account; "Previously before 2022, 7x was called Brookins Consulting" (README.md)
The launch	December 2023	The project was launched: the company works under the name 7x, and the fork's first outside change is merged on 11 December 2023.	"In 2023/12 Brookins Consulting an eZ Partner in USA ... renamed the company to '7x'" (https://se7enx.com/about/exponential-hosting-usa); "7x took over leadership of the project ... in 2023" (README.md); December 2023

Each of the three public sources names the stage it describes, which is why they give three different years. Through all three, Brookins Consulting and 7x are the same company, and it has worked with this software since its first versions.

Why the product is called Exponential

"At the specific request of Ibexa we are changing this projects name to 'Exponential' or 'Exponential (CMS)' effective as of August, 11th 2025" (https://exponential.earth/about/project). In the repository the rename landed the next day: the README, `composer.json` and `lib/version.php` changed on 12 August 2025, and one commit renamed the product in `bin/`, `doc/`, `design/` and the translations (332 files). Class names such as `eZContentObject`, paths and database tables did not change (August 2025). That is why you still see `eZ` in the code: it is the prefix of real identifiers, not the product name.

Nearly thirty years: the timeline

Dates come from the sources in the last column. "Tag" means the date of a git tag in this repository; "release overview" is the file doc/eng-us/release-overview, which lists every version with its source.

When	What happened	Source
1998	The first year of the copyright notice of the Exponential Foundation and 7x.	COPYRIGHT.md
1999	The first year of the copyright of eZ Systems AS in the code base.	COPYRIGHT.md
early 2000	"The content management system (CMS) eZ Publish was developed by the Norwegian company eZ Systems in early 2000."	https://se7enx.com/about/exponential-hosting-usa
2000	The founder of 7x starts with "the original source code of eZ Publish Version 1.0 and eZ Trade 1.0 building advanced simple to use e-commerce website applications".	https://graham.se7enx.com/7x
2001	Brookins Consulting is founded.	https://graham.se7enx.com/7x
2 September 2002	The oldest commit in this repository's history ("Added trunk dir."), from the Subversion repository of the 3.0 development line.	`git log --reverse`
November 2002 to February 2003	eZ Publish 3.0 betas and release candidates (2.9-3 "AKA 3.0-beta2" on 1 November 2002).	release overview
2 June 2004	eZ Publish 3.4.0.	release overview
22 December 2004	eZ Publish 3.5.0.	release overview
30 May 2005	eZ Publish 3.6.0.	release overview
12 October 2005	eZ Publish 3.7.0.	release overview

When	What happened	Source
2006	eZpedia, "The free eZ Publish encyclopedia that anyone can edit", starts; its introduction article is dated 31 December 2006 and the site was hosted by "USA eZ Publish Community Partner: Brookins Consulting".	https://ezpedia.exponential.earth/
3 December 2007	eZ Publish 4.0.0, the PHP 5 generation.	release overview
14 July 2010	eZ Publish 4.3.0.	release overview, tag 4.3.0
29 September 2010	eZ Publish 4.4.0.	release overview, tag 4.4.0
2011 to 2015	Community Project builds (2011.9 to 2014.11 and later) on GitHub (`ezsystems/ezpublish-legacy`), next to the 5.x line that put the legacy kernel beside a Symfony stack.	release overview
2014	7x, stage one: the initial idea, started quietly by registering the se7enx.com domain and preparing the fork of the entire ecosystem.	7x's account (owner, 18 October 2026); https://graham.se7enx.com/7x
26 August 2014	The first pull request from the GitHub account brookinsconsulting merged into the code base.	`git log`
2017 to 2021	The "5.90 line" releases v2017.08 to v2019.03.x of `ezsystems/ezpublish-legacy`; the last one listed is v2019.03.6.1, dated 3 September 2021.	release overview
2021	"The platform originally known as eZ Publish reached end-of-life in 2021. We kept going."	se7enxweb GitHub profile README
6 June 2019	The GitHub account se7enxweb is created.	GitHub API, `users/se7enxweb`
2022	7x, stage two: the project starts.	7x's account (owner, 18 October 2026); README.md
5 October 2023	The repository `se7enxweb/exponential` is created on GitHub, as a fork of `netgen/ezpublish-legacy`.	GitHub API
December 2023	7x, stage three: the project is launched. The first outside change of the fork is merged on 11 December (pull request #40, `2ddbc4c624`); the stewardship era begins.	December 2023
1 January 2024	6.0.0 is stable.	January 2024
7 January 2024	SQLite arrives as a database that needs no server: driver, schema handler, demonstration database and the setup wizard's choice (first released in 6.0.1, 29 January 2024; `git tag --contains a078ae1729`).	January 2024
4 February 2024	The repository `exponential-platform` (Exponential Platform 3.2.9, Symfony only) is created.	GitHub API
22 August 2024	admin3, a complete responsive administration design.	August 2024
23 February 2025	The repository `exponential-platform-legacy` is created.	GitHub API
April 2025	PHP 8.4 in, PHP 7 out; sites on PHP 7.4 stay on 6.0.7.	April 2025
July 2025	`COPYRIGHT.md` and `LICENSE.md` make licensing and credit explicit.	July 2025

When	What happened	Source
11 and 12 August 2025	The product is renamed Exponential (11 August on exponential.earth, 12 August in the repository).	https://exponential.earth/about/project, August 2025
December 2025	PHP 8.5 is supported.	December 2025
28 January 2026	The repository `exponential-platform-nexus` is created.	GitHub API
12 March 2026	The repository `exponential-platform-dxp` is created.	GitHub API
19 March 2026	The Exponential Foundation is founded in Seattle, Washington.	https://foundation.exponential.earth/
1 June 2026	MongoDB 8 as a database.	June 2026
7 June 2026	6.0.14 is tagged.	release overview, tag v6.0.14
July 2026	`ezpm` in the console, a one-command installer and the Kickstarter; the first commits of the Velocity web server (20 July).	July 2026, Velocity chronicle
31 July 2026	The repository `explayouts` (Exponential Layouts) is created.	GitHub API
22 September 2026	Velocity arrives in Exponential and becomes the recommended engine within two days.	September 2026
October 2026	jQuery 4 and admin4, an audit trail, `exp:ini`, a debug bar, content jobs, TinyMCE 8, Exponential Intelligence (`expai`, repository created 11 October), the LoveStack features of Mugo Web (12 October), and the help system you are reading.	October 2026, 6.0.15 changelog

The eZ Publish years, 1999 to 2021

Exponential did not start in 2023. Its code was first the code of **eZ Publish**, the content management system that the Norwegian company eZ Systems developed from 1999 (COPYRIGHT.md; https://se7enx.com/about/exponential-hosting-usa). eZ Publish was a product that its creator discontinued after almost eighteen years of service to the world (7x's account, owner, 18 October 2026); its last releases are listed until 2021 in the release overview. The name belongs to that past: the software you use today is Exponential. The history of this repository goes back to 2 September 2002, when the 3.0 line was in development, and contains every commit since: 27,242 commits on `main` on 18 October 2026 (`git rev-list --count origin/main`).

What the history shows, generation by generation:

- **eZ Publish 1 and 2** (2000 to 2002): the first generation, with eZ Trade for shops. 7x's founder started with it in 2000 (https://graham.se7enx.com/7x). Brookins Consulting released "eZ Region" as free software for eZ Publish 2.2, sold eZ Publish 2.2 support to businesses and government agencies, and created eZCommunity.net and the eZCommunity2 repositories "for the continued development and improvement of eZ Publish 2.2 + PHP5 support" (https://exponential.earth/about/about-exponential/features-added-by-7x/7x-core-contributions-to-ez-publish-cms-2000-2015). That line lives on as Exponential Basic.
- **eZ Publish 3** (2002 to 2007): introduced the content model you still use in Exponential, with classes, objects, versions, siteaccesses and the template language. The changelogs from 2.9 to 3.10 are in the help.
- **eZ Publish 4** (2007 to 2012): the PHP 5 generation, with ezjscore, the online editor and the 4.x

administration. Changelogs 4.0 to 4.7.

- **eZ Publish 5 and the community builds** (2012 to 2021): the legacy kernel next to the new Symfony stack, the monthly community builds, then the 2017.08 to 2019.03 releases until September 2021 (release overview).

During those years, according to 7x's own list, Brookins Consulting gave "Over 18 Years of steady forum support to eZ Publish Version 3 & 4 & 5 Community Forum Users", released eZ Publish compatible extensions, and mirrored the eZ Publish repositories, the project forge extensions and the documentation of doc.ez.no for preservation (same source as above). Twenty of its pull requests were merged into the code base between 2014 and 2023 (`git log --grep="from brookinsconsulting/"`).

The successors of eZ Publish took other roads. "eZ Platform was eZ Publish's technological successor before being abandoned by Ibexa", and Ibexa DXP OSS is "the rebranded and evolved successor of eZ Platform" (README, "What was eZ Platform?" and "What is Ibexa DXP OSS?").

Stewardship: from a dormant code base to Exponential 6

The history of Exponential begins: "This is where the story of Exponential 6 begins: the month a dormant code base changed hands" (December 2023). In that month the maintainers:

- merged the first outside change (pull request #40, 11 December 2023);
- rewrote the README to say what the software is, which PHP versions it supports and where to ask for help;
- added `SECURITY.md`, `CONTRIBUTING.md` and funding metadata;
- pointed `composer.json` at the se7enxweb packages, so the software installs with Composer under its new name;
- made the setup wizard work on PHP 8 and stop sending users to the servers of the former vendor;
- published the database update scripts from 5.4 to 6.0 for MySQL and PostgreSQL;
- set 6.0.0beta1 (25 December) and beta2 (31 December).

7x's description of the same moment: it "forked the recently no longer functional core of eZ Publish Project Software Stack and quickly released a working stable version as eZ Publish 6.0.0 in 2024/01" (https://se7enx.com/about/exponential-hosting-usa).

The ecosystem overview counts the work since: 6,556 ledger changes in 53 repositories since December 2023, 1,037 of them made by the se7enxweb team, the rest upstream history that the forks carry along. Every one of them is listed in the ledgers, and the coverage page shows that none is missing.

The 6.0.x releases

From doc/eng-us/release-overview and the git tags (`git tag -l 'v6.0.*' --sort=version:refname`):

Version	Date	Highlights (from the history pages)
6.0.0	1 January 2024	The first stable release of the fork
6.0.1	29 January 2024	SQLite driver and installer path (merged on 7 January 2024)
6.0.2	29 February 2024	

Version	Date	Highlights (from the history pages)
6.0.3	2 April 2024	
6.0.4	5 September 2024	The first release after admin3, the responsive administration design
6.0.5	1 October 2024	
6.0.6	17 December 2024	
6.0.7	1 February 2025	The last release for PHP 7.4 until 6.0.15
6.0.8	2 May 2025	The first release after PHP 8.4 support began (April 2025)
6.0.9	10 June 2025	
6.0.10	28 August 2025	The first release named Exponential
6.0.11	30 December 2025	Released in the month PHP 8.5 support arrived
6.0.12	10 February 2026	
6.0.13	18 April 2026	
6.0.14	7 June 2026	MongoDB 8 as a database
6.0.15	the date of the tag `v6.0.15`	Velocity, admin4, audit trail, Exponential Layouts, Exponential Intelligence, the help system and much more (changelog)

A published tag is never changed: "a fix for a problem in `v6.0.14` or an earlier tag reaches you as the next release, not as a corrected copy of the old one" (SECURITY.md).

The product: what Exponential is

"Exponential is a professional PHP application framework with advanced CMS (content management system) functionality. As a CMS its most notable feature is its fully customizable and extendable content model. It is also suitable as a platform for general PHP development" (README). The project site calls it "an award-winning, enterprise-grade Open Source Content Management System and development framework with functionality for web publishing, media portals, intranets, e-commerce and extranets" (https://exponential.earth/about/project).

A CMS, a CMF and a shop in one

Area	What you get	Read more
Content management	User-defined content classes and objects, versions, drafts, approval and scheduled publication, sections, states, the content tree	The content model and editing content
Framework (CMF)	MVC architecture, modules and views, a template engine, INI settings, a database abstraction layer, CLI scripts, a REST API, stand-alone libraries	Templates and design, REST API v1 and v2
E-commerce	"Advanced e-Commerce Stores (Built in)": products, baskets, orders, VAT, currencies, PayPal	README, "Main Exponential features"; Order receipts, Order list sorting
Multisite	Several sites from one installation, each a siteaccess with its own host, path, language and design	Siteaccess host and language matching, Multi-site settings
Multilingual	368 locale files (361 languages and countries) as content languages; administration translations (the current totals are at the top of the languages page) in `share/translations`, English included	Languages in Exponential

Area	What you get	Read more
Right to left	`dir="rtl"` in content and in the administration for right-to-left languages	Right to left
Page building	Exponential Layouts: layouts, zones, blocks and rules, a port of the concept of the upstream layouts packages	README, "Exponential Layouts"; The layouts editor
Writing assistance	Exponential Intelligence (`expai`): writes drafts, never publishes on its own, hosted providers in the US or the EU or a local model	Exponential Intelligence
Security	A Content Security Policy in the administration, two-step sign-in, a tamper-evident audit trail, hardening guides	Content Security Policy, Audit trail, Security hardening
Help	This help system: the repository's own documentation, searchable, by audience, with context help on every page	Help system
Serving	Apache, Nginx, LiteSpeed, Caddy, IIS and others, or the bundled Exponential Velocity; FrankenPHP as an alternative engine	README, "How Exponential is served"; FrankenPHP

Databases and operating systems

The supported databases, "in order of preference: 1. SQLite 3 (the default) · 2. PostgreSQL · 3. MongoDB · 4. MariaDB · 5. Oracle · 6. MySQL" (README, "Requirements"). Each is described on Database support. The operating systems are sorted into tiers on Operating system support: "almost any that runs PHP 7.4 or later".

The application stack

Layer	What Exponential uses	Source
Language	PHP 8.0 to 8.5 (the `php` requirement in `composer.json` allows up to 8.8); Velocity runs on 8.0 to 8.5 too, from its release 0.0.4.54	README, "What version of PHP is required"; composer.json
Packages	Composer: the kernel `se7enxweb/exponential` and its extensions are Composer packages on Packagist	README, "Important Notice"
Libraries	expcomponents, the fork of the Zeta Components that 7x and the Exponential Foundation maintain since 6.0.15 (archive, cache, database, feed, image conversion, mail, WebDAV and more; Apache 2.0); Zeta Components before	composer.json; expcomponents
Installer	`se7enxweb/exponential-legacy-installer`, a Composer plugin that places extensions in `extension/`	Ecosystem
Web server	Any server that rewrites URLs, or Exponential Velocity	README
Database	SQLite 3, PostgreSQL, MongoDB, MariaDB, Oracle, MySQL	README
Front end	jQuery 4 (through ezjscore) and Bootstrap Icons in admin4; TinyMCE 8 as the online editor	October 2026

The extension ecosystem

`composer.json` of this release requires 66 `se7enxweb/*` packages, so every installation brings them (`grep -c '"se7enxweb/' composer.json` inside `require`). Many are inactive until you switch them on (The default extension distribution). A selection, with what each does:

Package	One line	Read more
`explayouts` and the eleven `explayouts-*` packages	Exponential Layouts, the visual page builder, its editor, API, content browser and queries	Layouts editor
`expai`	Exponential Intelligence, the writing assistant	expai
`expqueue`	The job queue for queued work of extensions	expqueue
`exp_adminui`	AdminUI, an alternative administration: "a feature complete port of the legendary Netgen AdminUI" (repository description)	GitHub
`expui`	"A feature complete replacement for YUI Libraries and Features within an jquery4 based UI Library" (repository description)	GitHub
`ezflow`, `ezwebin`, `ezdemo`	Page flows and the classic website interface and demo site	ezflow, ezwebin
`eztags`	Taxonomy: tags for classifying content	eztags
`cjw_newsletter`	Newsletters, from CJW Network	cjw_newsletter
`sevenx_authentication_2fa`	Two-factor authentication (TOTP and e-mail codes)	Two-step sign-in
`sevenx_dse`	The database source editor, powered by AdminNeo	sevenx_dse
`sevenx_themes_simple`, `sevenx-themes-media`	The default simple theme and the media site theme	sevenx_themes_simple
`git_manager`	Update code checkouts and back up a site from the administration	git_manager
`ezoracle`	The Oracle database driver	Database support
`ezpaypal`	PayPal payments for the shop	ezpaypal
`ezprestapi`	The REST API	ezprestapi
`ezmultiupload`, `ezie`, `ezodf`, `ezstarrating`, `ezgmaplocation`, `ezautosave`	Multi-file upload, image editor, OpenOffice import and export, ratings, map locations, autosave	Extensions
`bcgooglesitemaps`, `bcwebsitestatistics`, `bccie`	Brookins Consulting extensions: sitemaps, statistics, collected information export	Extensions
`xrowmetadata`, `xrowextract`	SEO metadata and data extraction	Extensions
`recaptcha`, `hcaptcha`	Spam protection for forms	Extensions
`expsite-*`, `expquery-translator`	The site API, installer and query translator ported from the Netgen ecosystem	composer.json

Each extension has its own repository under https://github.com/se7enxweb, its own releases and its own changelog (extension changelogs).

The platform line: Exponential Platform, Legacy, Nexus and DXP

Next to Exponential 6, 7x maintains the Symfony based line that descends from eZ Platform and Ibexa OSS. "In short Exponential Platform is the direct continuation of the software stack known as eZ Platform created by eZ Systems and discontinued by Ibexa" (README, "What is Exponential Platform?"). The descriptions

below come from each distribution's own `composer.json` and repository description (read with `gh api repos/se7enxweb/<name>`), not from the upstream feature lists.

Distribution	What it is	Includes Exponential 6?
Exponential Platform 3.2.9 (`exponential-platform`)	"CMS / DXP based on the last free versions of eZ Platform. Symfony / New Stack Only."	No
Exponential Platform Legacy (`exponential-platform-legacy`)	"eZ Platform (+Legacy) distribution": the Symfony platform plus the legacy bridge and `se7enxweb/exponential`	Yes
Exponential Platform Nexus (`exponential-platform-nexus`)	"Platform v2/v3/v4/v5 + Exponential v6.x (Legacy) + Netgen Media Site + NG Layouts", with a demo site; requires `netgen/layouts-ui` and `se7enxweb/exponential`	Yes
Exponential Platform DXP (`exponential-platform-dxp`)	"A meta package for installing 7x Exponential Platform DXP": Symfony only, with the admin UI, content forms, REST, GraphQL and GraphiQL, HTTP cache, Solr and core search, rich text, matrix and query field types, notifications, messenger, cron and JWT; of the layouts packages it requires only `se7enxweb/layouts-core`	No
Nexus starter (`exponential-platform-nexus-starter`)	A ready-to-run Nexus installation built on the DXP package, with Exponential Layouts and the media site data	No

The DXP does not ship a layouts editor, an approval workflow, a scheduler or the legacy kernel: its requirements contain none of them. When an existing Exponential site has to move forward step by step, the dual-kernel distributions Legacy and Nexus run Exponential 6 next to the platform (README, "What Exponential Platform DXP adds"). The platform repositories, their forks of Symfony, Twig and Doctrine pieces and their histories are mapped on The Exponential platform ecosystem. Learn more at https://platform.exponential.earth.

Exponential Basic

Exponential Basic is the line that descends from eZ Publish 2. Its repository is a fork of `brookinsconsulting/ezcommunity2` (GitHub API), the eZCommunity2 repository that Brookins Consulting created for eZ Publish 2.2. Its description: "Exponential Basic v2.4.0.3: PHP 8.4/8.5.9 + Maintained by 7x : Content Management System Powered by Web Server + PHP + Database (MySQL, Postgresql, SQLite, Informix) ... GNU GPLv2+".

Exponential Velocity

Exponential Velocity is the PHP web server that Exponential recommends "for every stage from development to production" (Velocity chronicle). "Velocity is a PHP web server for real production work. It began as a fork of the Qbix web server engine, and the two projects have since drifted far apart" (the package's README, `se7enxweb/exponential-velocity`). The repository is a fork of `Qbix/webserver` (GitHub API) and the package is MIT licensed (its `composer.json`).

Its first commits are from 20 July 2026, its first release `v0.0.1` from September 2026, and it became part of Exponential on 22 September 2026 (September 2026). What it measures, and how to run it, is in the README section "Exponential Velocity —the fast option" and on The persistent-worker server.

Services and hosting

7x sells services around the open source software; the software itself stays free. From https://se7enx.com/services and the GitHub profile README:

- web design and development, e-commerce and payment integration, training;
- "CMS Upgrades: Migration from eZ Publish to Exponential v6", that is, moving a site built while the software was still called eZ Publish onto the current Exponential 6;
- support and maintenance contracts for existing sites;
- managed hosting: "Exponential Website Hosting" and "Exponential Platform SSD Hosting" (https://se7enx.com/).

Prices and terms are on the 7x website, not here.

The community

Place	What it is	Source
share.exponential.earth	The community portal: forums, blogs, tutorials and projects (extensions), "Proudly hosted by se7enx.com"	the site
Forums	Questions and discussions; there is a forum for Exponential Platform too	README, "Where to get more help"
Telegram	"Share Exponential! Telegram Community Support Chat"	README
eZpedia	"The free eZ Publish encyclopedia that anyone can edit", since 2006, under the GNU Free Documentation License	the site
Documentation	"The Official Documentation For Exponential 6 CMS", covering 3.x to 6.x	the site
Training books mirror	"eZ Publish Training Books (100% Compatible With Exponential)" as PDFs, free to everyone	repository description
Learning eZ Publish	"The Book, 'Learning eZ Publish' was the first book published about eZ Publish ... Now Released under the GNU Free Documentation License GFDL"	repository description
Exponential issue tracker	Bugs and improvements of the software	README, "Issue tracker"; When something goes wrong
Community issue tracker	"almost anything Exponential Project / Community / Ecosystem / Website Software related"	README

Contributors and friends of the project

The code base carries the work of hundreds of people. Before 2016, 194 different commit authors appear in the history (`git log --until=2016-01-01 --format=%an | sort -u`), and pull requests were merged from accounts such as ezsystems, jdespatis, patrickallaert, joaoinacio, jeromegamez and many more. Since December 2023 pull requests have come from se7enxweb, ekked, fwoldt, netgen, brookinsconsulting, mugoweb and others (`git log --grep="Merge pull request"`).

Exponential names the companies whose open source work it builds on:

- **Netgen**: Exponential Layouts brings "the concept of the upstream layouts packages to the Exponential kernel" (README); the Nexus distribution is built on Netgen Media Site and the upstream layouts

packages; the `se7enxweb/exponential` repository is a GitHub fork of `netgen/ezpublish-legacy`.

- **Mugo Web**: their LoveStack improvements were rebuilt in 6.0.15 "with full credit to Mugo Web" (README, "Exponential Distribution Integration: LoveStack").
- **CJW Network**: `cjw_newsletter` and the CJW flavor of Nexus; their request-shield is documented as a companion in front of Exponential's HTTP cache (HTTP cache).
- **eZ Systems**, whose copyright from 1999 to 2014 stays in the code (COPYRIGHT.md).

Open source licenses

What	License	Source
The software	GNU General Public License, version 2 or any later version. "Using Exponential under the terms of the GNU GPL is free of charge. The Professional License is no longer available."	README, "License"; LICENSE
Contributions	"any modifications made to the project are also released under the same GPLv2 or later license"	CONTRIBUTING.md
The documentation, including this page	GNU Free Documentation License 1.3 or any later version	The license of this documentation, Free documentation
Exponential Velocity	MIT	its `composer.json`
Inherited eZ Systems documentation	the terms it came with (GPL v2)	The license of this documentation

Principles you can see in the code

These are not slogans; each is something you can check in this repository.

- **Git first.** "Building websites with Exponential? The recommended way is to get the kernel and every se7enxweb extension as Git clones with their full history" (README, "Installation"; Develop with Git). The full history back to 2002 is kept, and every change since December 2023 is explained in the history.
- **Published versions never change.** A fix always arrives as the next version (SECURITY.md).
- **Security by default.** Private reporting to security@se7enx.com with a first answer "within one business day" (SECURITY.md); a Content Security Policy in the administration; a hardening guide written "secure by default in the OpenBSD sense" (Security hardening).
- **Accessible and readable.** Light and dark themes, right-to-left layouts, keyboard shortcuts, and the rule that no administration template carries inline styles (CONTRIBUTING.md; Keyboard shortcuts).
- **Current PHP.** "Current releases are primarily aimed at easing the requirements to support current versions of the PHP language like PHP 8.2, 8.3, 8.4, 8.5, 8.6 and beyond php9" (README, "Exponential Project Status").
- **Open source, completely.** "Everything we do is GPL-licensed and freely available" (se7enxweb GitHub profile README); the documentation is free documentation.
- **Honest numbers.** Performance figures in the README are "reproducible with scripts that ship with the project".

7x in numbers

Counted on 18 October 2026. Numbers change; the command or page in the last column gives today's value.

Figure	Value	How it was counted
Years in the copyright notice	1998 to 2026	COPYRIGHT.md
Commits in this repository's `main`	27,242	`git rev-list --count origin/main`
Oldest commit	2 September 2002	`git log --reverse`
Merged pull requests in the history	1,062 (as of 25 October 2026; the inherited history included)	`git log --oneline --grep="Merge pull request" origin/main | wc -l`
Merged pull requests on GitHub (se7enxweb/exponential)	288 (as of 25 October 2026)	GitHub search `repo:se7enxweb/exponential is:pr is:merged`
6.0.x releases	15 (6.0.0 to 6.0.14), followed by the current 6.0.15 line	`git tag -l 'v6.0.*'`
se7enxweb packages required by `composer.json`	66	composer.json
Packages of the vendor se7enxweb on Packagist	219	https://packagist.org/packages/list.json?vendor=se7enxweb
Public repositories of the GitHub account se7enxweb	1,025, of them 95 not forks	`gh repo list se7enxweb --limit 1100`
Changes in the history since December 2023	6,556 in 53 repositories, 1,037 by the se7enxweb team	Ecosystem
Locale files	368	Languages

How to get involved

- **Use it and tell us.** Report a bug with When something goes wrong; ask in the forums or on Telegram.
- **Grow from user to developer.** Install with Git, keep your work on a branch and send pull requests: Develop with Git, CONTRIBUTING.md.
- **Translate.** The administration has many translations (the current totals are at the top of the languages page); Languages explains how to add or complete one.
- **Improve this help.** Every page has "Edit on GitHub"; the pages are free documentation (Free documentation).
- **Fund the work.** "A wide range of donation options available at sponsor.se7enx.com, paypal.com/paypalme/7xweb and github.com/sponsors/se7enxweb" (README, "Donate and make a support subscription"); the Foundation has its own sponsor page, https://github.com/sponsors/exponentialcms.
- **Hire 7x.** For a new site, the migration of a site built in the eZ Publish years, or support of an existing site: https://se7enx.com/.

Sources

Repository files (all in `se7enxweb/exponential`):

- README.md, COPYRIGHT.md, LICENSE, SECURITY.md, CONTRIBUTING.md, composer.json
- doc/eng-us/release-overview, the history, the ecosystem overview, the Velocity chronicle
- December 2023, July 2025, August 2025
- The 6.0.15 changelog, Languages, The license of this documentation
- The git history: `git log`, `git tag`, `git shortlog`

7x's own account of its three stages (2014, 2022, 2023), given by the owner of 7x for this page on 18 October 2026.

Public web pages (read on 18 October 2026):

- https://se7enx.com/, https://se7enx.com/about, https://se7enx.com/services, https://se7enx.com/about/exponential-hosting-usa, https://graham.se7enx.com/7x
- https://exponential.earth/about/project, https://exponential.earth/about/about-exponential/features-added-by-7x/7x-core-contributions-to-ez-publish-cms-2000-2015
- https://foundation.exponential.earth/
- https://ezpedia.exponential.earth/, https://share.exponential.earth/, https://doc.exponential.earth/
- https://github.com/se7enxweb (profile README and the GitHub API), the repositories named on this page
- https://packagist.org/packages/list.json?vendor=se7enxweb

Related pages

- Who is the Exponential Foundation?
- The history of Exponential
- The Exponential platform ecosystem
- Develop with Git
- Free documentation: this page is free to copy, translate and improve.

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. Start with Partners: start here and How to apply.

Who is the Exponential Foundation?

The nonprofit behind the Exponential open source project, and the family of products it stands for, from the Exponential 6 kernel to the Symfony based Exponential Platform distributions, Velocity, Exponential.

The **Exponential Foundation** is the nonprofit organization behind the Exponential open source project. This page explains what the Foundation is, how it relates to 7x, which products make up the Exponential family and who each one is for, what the project commits to, how it grows, and how you can support it or join it.

Every statement names its source: a file in this repository, a public web page, or a command you can run yourself. The page does not describe what the sources do not say. Where a fact could not be checked, the page says so.

> **Note**
> The company that does most of the work, and its history of nearly thirty years with this code, has its own page: Who is 7x?

Contents

1. In short
2. What the Foundation is
3. The Foundation and 7x
4. What the sources do not say
5. The product family at a glance
6. The products one by one
7. Which product is for you
8. The support commitment
9. How the project grows
10. The PHP stack it stands on and gives back to
11. Timeline
12. How to support or join
13. Glossary
14. Sources

In short

- **A nonprofit for an open source project.** The Foundation describes itself as "a 501(c)(3) Non-Profit Corporation founded in Seattle, Washington, USA", founded on 19 March 2026 (https://foundation.exponential.earth/).
- **Its mission** is "dedicated directly to the continued Funding, Adoption, Support, Direction, Education, Development, Promotion and Documentation for the Exponential Enterprise Open Source Project" (same page).
- **It operates the project.** "This project is operated by the Exponential Foundation a Public Non-Profit Corporation" (https://exponential.earth/about/project).
- **7x is its main contributor.** The Foundation names 7x "the current leader in contributions both financial, code, extension development, new features, infrastructure, maintenance and much much more back to the project" (https://foundation.exponential.earth/).
- **It shares the copyright.** The kernel's notice reads "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved." (COPYRIGHT.md).
- **Everything is free software.** The code is GPL v2 or later, the documentation GFDL 1.3 or later (The license of this documentation, CONTRIBUTING.md).

What the Foundation is

In its own words

Statement	Source
"a 501(c)(3) Non-Profit Corporation founded in Seattle, Washington, USA" on 03/19/2026	https://foundation.exponential.earth/
"a Public Non-Profit Corporation (Tax Exempt 501(3)(c) Organization)"	https://foundation.exponential.earth/, https://exponential.earth/about/project
"The mission of the Exponential Foundation is dedicated directly to the continued Funding, Adoption, Support, Direction, Education, Development, Promotion and Documentation for the Exponential Enterprise Open Source Project."	https://foundation.exponential.earth/
It will "solicit funding and sponsorship" for "Dynamic PHP Based Web Application Publishing Content Management System(s) like Exponential, Exponential Platform and Exponential Basic."	https://foundation.exponential.earth/
It serves "end users, designers, developers, coders, no-code developers, system administrators, programmers and website builders"	https://foundation.exponential.earth/
"This project is operated by the Exponential Foundation"	https://exponential.earth/about/project
"The Exponential Project and the Exponential Foundation support the open source community directly, as part of our daily routine and work ethic, and put the support needs of community users first."	README.md, "An open community"
"The Exponential Foundation funds the open-source project: the core libraries, the documentation, the community tooling, and the long-term survival of the platform as free software."	GitHub profile README of se7enxweb

What the Foundation funds

The se7enxweb GitHub profile README lists what a sponsorship of the Foundation pays for:

- "Core library maintenance —`exponential`, `exponential-platform`, `exponential-platform-legacy`";
- "API documentation at api.exponential.earth" and "Core documentation at doc.core.exponential.earth";
- "GraphQL, REST, and Solr —modern integration layers for headless and API-first builds";
- "Community infrastructure —Slack, mirrors, wikis, and developer resources";
- "zetacomponents security hardening —14 upstream security PRs submitted in April 2026 alone".

The Foundation's site

https://foundation.exponential.earth/ has the sections About, Mission Statement, Support, Goals, News and Supporters. The Goals section says there is "a long detailed list of growth goals for the Exponential Project in 2026"; the list itself is not on the page, so it is not repeated here. The News section announced the Foundation's formation and its tax exemption.

The Foundation and 7x

The Foundation and 7x are two organizations with one project:

	7x	Exponential Foundation
What it is	A company: "a responsive North American web design, support, development and website hosting company" (https://se7enx.com/about)	A nonprofit corporation (https://foundation.exponential.earth/)
Role	Leads development, support, hosting and services; "the current leader in contributions" to the project	Operates the project; funds and directs it
Sponsor page	https://github.com/sponsors/se7enxweb	https://github.com/sponsors/exponentialcms
Copyright	"Exponential Foundation and 7x", together, in COPYRIGHT.md, LICENSE and the file headers	
Website	https://se7enx.com/	https://foundation.exponential.earth/, https://exponential.earth/

"The Exponential Open Source Project is funded by donations of time and resources by 7x (and Exponential Sponsors)" (https://exponential.earth/about/project). The Foundation's Supporters section: "The Exponential Foundation supports direct change within the Exponential Project, Sub Projects, New Features, Documentation through the work of 7x" (https://foundation.exponential.earth/).

The profile README explains the two sponsorship streams: sponsoring 7x funds "the company that keeps the lights on" (monthly releases, Zeta Components patches, the LegacyBridge, the Symfony fork, the database translator, the training book mirror); sponsoring the Foundation funds "the open-source project itself".

What the sources do not say

To keep this page honest, these questions are left open because no public source used here answers them: who sits on the Foundation's board, how it is governed, whether it has members besides its supporters, and what its budget is. Ask the Foundation through its website if you need to know. The Foundation's postal address and registration numbers are published on its websites and are not repeated here.

The product family at a glance

The products the Foundation names are "Exponential, Exponential Platform and Exponential Basic" (https://foundation.exponential.earth/). Around them 7x and the project maintain servers, extensions and tools. The "State" column is the state on 18 October 2026, with its source.

Product	What it is	For	State
Exponential 6	The CMS, CMF and shop kernel this installation runs	Everyone who builds or runs a site with it	6.0.14 released 7 June 2026; 6.0.15 is the current line (release overview)
Exponential Basic	The simple CMS of the eZ Publish 2 line	Education, personal use, small sites	v2.4.0.3 (repository description)
Exponential Platform 3.2.9	Symfony-only CMS from the last free eZ Platform	Teams on the eZ Platform 3 stack	3.2.9 (repository description)

Product	What it is	For	State
Exponential Platform Legacy	Symfony platform plus Exponential 6 (dual kernel)	Sites that keep Exponential 6 and add Symfony	2.5.x to 5.0.x lines (README)
Exponential Platform Nexus	Platform plus Exponential 6, Exponential Layouts and Media Site, with demo content	Agencies and teams who want a complete site fast	1.0.0.x to 1.3.0.x lines (README)
Exponential Platform DXP	Symfony-only DXP meta package, the freedom-enabled continuation of Ibexa DXP OSS	New Symfony projects	"Stable; Open Source" (its composer.json)
Exponential Velocity	A web and application server written in PHP	Anyone serving Exponential, and other PHP applications	Required at `~0.0.4.57` (README; it runs on PHP 8.0 since release 0.0.4.54)
Exponential Intelligence	A writing assistant in the administration	Editors and the people who manage them	Required since 6.0.15, `~0.0.1.18` in composer.json
Exponential Layouts	A visual page builder: layouts, zones, blocks and rules	Editors and designers	Required since 6.0.15, `~1.4.43` in composer.json
AdminUI	An alternative administration interface	Editors who prefer it	Required since 6.0.15, `~1.0.0.50` in composer.json
The help system	The help you are reading	Every user of the administration	Phase 1 in 6.0.15 (Help as content)
The extensions	66 se7enxweb packages required by the kernel	Site builders	Each with its own releases

The products one by one

Exponential 6, the kernel

What it is. "Exponential is a professional PHP application framework with advanced CMS (content management system) functionality. As a CMS its most notable feature is its fully customizable and extendable content model" (README). Its code was once the code of eZ Publish, a product that its creator discontinued after almost eighteen years of service to the world (7x's account, owner, 18 October 2026); Exponential carried that code forward and is published as the Composer package `se7enxweb/exponential` (composer.json). The README lists its main features: user-defined content classes, version control, multilingual support, a built-in search engine, role-based permissions, approval and scheduled publication, multisite support, built-in e-commerce, workflows, a template engine, a headless REST API, a database abstraction layer, clustering, and a plugin API.

Who it is for. Organizations that run structured content, several sites or several languages from one installation, and developers who build applications on its framework.

Where it stands. 6.0.15 is the current line, after 6.0.14 (7 June 2026). It adds, among much else, Velocity, the admin4 design, an audit trail, Exponential Layouts, Exponential Intelligence, content jobs, packages end to end, more than 360 locales in the README's count, the LoveStack features of Mugo Web and this help system (6.0.15 changelog, README "Recent improvements"). The story of every change since December 2023 is in the history.

Exponential Basic

What it is. "Exponential Basic is the simple to us to use CMS Framework that is for everyone, educational purposes, personal use and worldwide general" (repository description of se7enxweb/exponentialbasic). Its repository is a fork of `brookinsconsulting/ezcommunity2`, the line of eZ Publish 2.2 that Brookins Consulting kept alive (GitHub API; Who is 7x?).

Who it is for. Learners, personal sites and anyone who wants a small CMS.

Where it stands. "Exponential Basic v2.4.0.3: PHP 8.4/8.5.9 ... (MySQL, Postgresql, SQLite, Informix) ... GNU GPLv2+", website https://basic.exponential.earth (repository description).

Exponential Platform 3.2.9

What it is. "7x Exponential Platform 3.2.9 CMS / DXP based on the last free versions of eZ Platform. Symfony / New Stack Only. No Exponential 6.x Legacy Support" (repository description of `se7enxweb/exponential-platform`). Its `composer.json` requires the se7enxweb forks of the eZ Platform 3 bundles (kernel, admin UI, REST, GraphQL, rich text, Solr, HTTP cache and others).

Who it is for. Teams that run eZ Platform 3 and want to stay on it with current PHP.

Exponential Platform Legacy

What it is. "Exponential Platform Legacy provides a clean distribution of Exponential Platform ... with Exponential (Legacy) Included (now with SQLite support)" (repository description). Its `composer.json` requires `se7enxweb/exponential` and `se7enxweb/legacy-bridge`, so Exponential 6 runs next to the Symfony platform. The README lists the lines 2.5.x, 3.3.x, 4.6.x and 5.0.x as "Dual Kernel Symfony CMS which also includes a full copy of Exponential 6".

Who it is for. Existing Exponential sites, including sites built when the software was still called eZ Publish, that want Symfony beside them and move forward step by step (Legacy bridge).

Exponential Platform Nexus

What it is. "Platform v2/v3/v4/v5 + Exponential v6.x (Legacy) + Netgen Media Site + NG Layouts + Symfony 7.4/5.4/3.4 PHP 8.0-8.5 Compatibility" (repository description of `se7enxweb/exponential-platform-nexus`). Its `composer.json` requires `se7enxweb/exponential`, `netgen/layouts-core`, `netgen/layouts-standard` and `netgen/layouts-ui`, so it ships the layouts editor. The README lists Nexus 1.0.0.x to 1.3.0.x, for Platform v2.5 to v5. A CJW flavor exists (`se7enxweb/cjw-exponential-platform-nexus`) and a ready-to-run starter (`exponential-platform-nexus-starter`, "Currently In Public Beta" in its description) (Nexus starter).

Who it is for. Agencies and teams who want a complete website with demo content, layouts and an administration in minutes.

Exponential Platform DXP

What it is. "Exponential Platform DXP is a free, open-source Digital Experience Platform built on the Symfony framework, the community continuation of the eZ Platform and Ibexa OSS line maintained by 7x" (README). The meta package `se7enxweb/exponential-platform-dxp` describes itself as "Stable; Open Source" with the license "GPL-2.0-or-later or proprietary" (its `composer.json`).

What it ships, read from its requirements: the admin UI, content forms, REST, GraphQL and the GraphiQL console, HTTP cache, core and Solr search, the rich text, matrix and query field types, notifications, messenger, cron, the standard design, Twig components, JWT authentication and

`se7enxweb/layouts-core`. What it does **not** ship: a layouts editor (no layouts UI or standard blocks), an approval workflow, a scheduler, or the legacy kernel. For Exponential 6 beside Symfony, choose Legacy or Nexus.

What the README says it adds over Ibexa DXP OSS: "The SQLite database driver and framework support", current PHP ("PHP 8.3 and later"), "A path from Exponential 6" through the dual-kernel distributions, and "A name of its own".

Who it is for. Organizations with a lot of structured content, teams leaving eZ Platform or Ibexa OSS, and Symfony developers (README, "Who it is for"). Start from the DXP skeleton.

Exponential Velocity

What it is. "A pure PHP web server. No nginx, no Apache, no php-fpm. One process serves static files, PHP scripts, WebSocket connections, and a live dashboard" (README of `se7enxweb/exponential-velocity`). It "began as a fork of the Qbix web server engine" and keeps the application loaded between requests (README of Exponential). It is MIT licensed (its `composer.json`).

Who it is for. Every Exponential site: it is "the recommended engine for every stage, from development to production" (README, "Engines"). FrankenPHP is "the production-ready alternative", PHP's built-in server the development default.

Where it stands. First commits in July 2026, first release `v0.0.1` in September 2026, part of Exponential since 22 September 2026 (Velocity chronicle); the README names it as required at `~0.0.4.57`; it runs on PHP 8.0 since release 0.0.4.54.

Exponential Intelligence

What it is. "A writing assistant built into the administration interface" that "writes drafts, as the signed-in editor and with that editor's own permissions", never publishes on its own, and works with hosted providers in the US or the EU or with a local open-weight model (README, "Exponential Intelligence"). Package `se7enxweb/expai`.

Who it is for. Editors, project managers and content writers ("Great for project managers and content writers!", repository description).

Where it stands. Required in `composer.json` since 6.0.15; "A new installation activates it by default" and it "sends nothing anywhere until you add a provider key" (README). Read Exponential Intelligence before you switch it on.

Exponential Layouts

What it is. "A visual page builder for Exponential. It brings the concept of the upstream layouts packages to the Exponential kernel: a layout is a saved arrangement of blocks in zones, and a rule decides which pages get it" (README). Packages `se7enxweb/explayouts` and the `explayouts-*` family.

Who it is for. Editors who rearrange pages without template work, and designers (The layouts editor).

AdminUI

What it is. "A feature complete port of the legendary Netgen AdminUI Administration UI. This solution depends on Exponential 6.0.15* or greater" (repository description of `se7enxweb/exp_adminui`). It is one of the administration designs; Admin 4 is the one most installations use (Help for editors and users).

The help system

What it is. The help in the administration that shows the git-tracked documentation of `doc/`, rendered safely, searchable and filtered by audience, with a "?" context help button on every page (Help system). Its pages are free documentation (Free documentation). Help pages written as content in the database are planned in phases; phase 1 is in 6.0.15 (Help as content).

The extension ecosystem

`composer.json` requires 66 `se7enxweb/*` packages: themes, the shop's PayPal gateway, tags, newsletters, two-factor authentication, the Oracle driver, the REST API, the job queue, the page builder and more. The list, with one line each, is on Who is 7x? and in The default extension distribution. On Packagist the vendor se7enxweb has 219 packages (https://packagist.org/packages/list.json?vendor=se7enxweb).

Companion projects

- **request-shield** is a project of CJW Network (https://github.com/cjw-network/request-shield), not of the Foundation. Exponential documents how its HTTP cache works with request-shield in front (HTTP cache) and how to load it under Velocity (`config.php-RECOMMENDED`). Its website https://request-shield.com/ was private when this page was written, so nothing more is said here.
- **The package server** `exponential.packages.exponential.earth` serves the packages of the setup wizard ("Users do not need to use this server outside the setup wizard usage", repository description).
- **AdminNeo**, through `sevenx_dse`, the database source editor (sevenx_dse).

Which product is for you

You want	Choose	Read
A CMS with a shop, multisite and many languages, on PHP only, no database server needed	Exponential 6 with SQLite	Getting started, Database support
The fastest way to serve it	Exponential Velocity	Persistent-worker server
Editors who build pages themselves	Exponential Layouts	The layouts editor
Help writing and translating content	Exponential Intelligence	expai
A Symfony project with REST and GraphQL, no legacy kernel	Exponential Platform DXP	DXP skeleton
A complete demo site with layouts on the Symfony platform	Exponential Platform Nexus	Nexus starter
To keep an Exponential 6 site and add Symfony next to it	Exponential Platform Legacy	Legacy bridge
A small, simple CMS to learn with	Exponential Basic	https://basic.exponential.earth

The support commitment

The project does not publish one "support policy" document. The commitments below are each written down in the place named; together they describe how the project looks after its users.

PHP versions

PHP	Exponential 6	Source
7.4	Up to 6.0.7, and again from 6.0.15 ("PHP 7.4 is usable again from 6.0.15, without support, as a bridge for organizations that cannot change the PHP of their servers yet"); see Why Exponential supports PHP 7.4	README
8.0	Supported, so the stock PHP of Red Hat Enterprise Linux 9 runs it, under Velocity too from its release 0.0.4.54 (until then Velocity needed 8.1)	README, PHP 8.0 support
8.1 to 8.5	Supported, Velocity included	README
8.6 and later	Allowed by the `php` requirement of `composer.json` (up to 8.8); the project aims at "PHP 8.2, 8.3, 8.4, 8.5, 8.6 and beyond php9"	composer.json, README

The continuous integration runs the test suite on PHP 8.1 to 8.5 and checks that PHP 8.0 still boots, on every push to `main` and every pull request (CI).

Security fixes

- "Exponential 6.0 is the maintained line. Security fixes are made on the `main` branch ... and published with the next 6.0.x release" (SECURITY.md).
- Report privately to security@se7enx.com: "You can expect a first answer within one business day" (SECURITY.md).
- 5.x, 4.x, 3.x and older are not supported (SECURITY.md).
- Hardening is documented for administrators: Security hardening and chapter 13 of the installation book.

Releases and version numbers

- **Published versions never change.** "A published tag is never changed: a fix for a problem in `v6.0.14` or an earlier tag reaches you as the next release, not as a corrected copy of the old one" (SECURITY.md).
- **The last position counts on.** "After `0.1.9` the next release is `0.1.10`; only the last position counts up unless you decide otherwise", and versions are sorted with `--sort=version:refname`, not by name (Extensions, "Release it"; the same for Velocity in Getting the code).
- **An extension states its version in its own files.** `ezinfo.php` and `extension.xml` carry the release number (Extensions).
- **Distribution through GitHub and Packagist.** Every package is a GitHub repository under se7enxweb and a Composer package; `composer require` or a Git clone installs it (Getting the code).

Documentation

- The documentation lives in the repository, in `doc/`, and this help shows it in the administration (Help system).
- It is free documentation under the GNU Free Documentation License 1.3 or later (The license of this documentation).
- Every change since December 2023 is told in the history and recorded in the ledgers.

Languages and accessibility

- 368 locale files, 43 of them right to left, and administration translations (the current totals are at the top of Languages in Exponential) (Languages); `dir="rtl"` in content and administration (Right to left).
- The admin4 design follows WCAG rules that tests check: a contrast test of the color tokens in light and dark mode and target sizes of at least 24 by 24 pixels (Styling the administration).

Security headers and the Content Security Policy

Exponential sends a Content Security Policy from `site.ini [ContentSecurityPolicy]`; "By default it is **report-only on the administration** and off on the sites" (Content Security Policy). New administration templates carry no inline styles or scripts so that a strict policy can be enforced (CONTRIBUTING.md).

Git first

"Building websites with Exponential? The recommended way is to get the kernel and every se7enxweb extension as Git clones with their full history" (README, "Installation"; Develop with Git).

How the project grows

Contributors and pull requests

- **Every pull request is merged.** While it is open, its author squashes it and rebases it onto `main` as the reviews ask; once it is merged, `main`'s history is never rewritten, and later fixes come as new pull requests (CI, commit messages; Code reviews).
- **Contributions stay free.** "any modifications made to the project are also released under the same GPLv2 or later license" (CONTRIBUTING.md).
- **Copyright assignment (CLA).** The Exponential Foundation and 7x require a signed Contributor Licensing Agreement for every code contribution to the core kernel, in any form, before it is merged. By signing and submitting, the contributor assigns the copyright in the contribution to the Exponential Foundation and 7x, which become its owners, and gives up control over it: the Exponential Foundation and 7x may use, change, relicense and distribute it in any way, also commercially and under other licenses, without asking or paying the contributor, who cannot withdraw it or set conditions on it. Users' own custom extensions are not affected (Copyright assignment (CLA)).
- **Who contributes.** Since December 2023 pull requests have come from se7enxweb, ekked, fwoldt, netgen, brookinsconsulting, mugoweb and others (`git log --grep="Merge pull request"`). Before that, hundreds of people built eZ Publish (Who is 7x?).
- **Work of other projects, credited.** The LoveStack improvements of Mugo Web were rebuilt "with full credit to Mugo Web" (README); Exponential Layouts ports the concept of the upstream layouts packages (README).

Community places

Place	What it is
share.exponential.earth	Forums, blogs, tutorials and projects (extensions)
Telegram	The community support chat (README)
eZpedia	"The free eZ Publish encyclopedia that anyone can edit", since 2006, GFDL
Documentation	"The Official Documentation For Exponential 6 CMS"

Place	What it is
Training books	"three original developer training books (700+ pages combined)" (profile README)
Community issue tracker	Anything about the project, community, ecosystem or websites (README)
Discord and Slack	Named among the support channels on https://exponential.earth/about/project

No events (conferences, meetups) are documented in the sources used for this page.

The PHP stack it stands on and gives back to

Project	What Exponential uses it for	What the project gives back	Source
PHP	The language	Support for each new PHP version	README
Composer and Packagist	Installation of the kernel and every extension	219 packages published by se7enxweb	composer.json, Packagist
Zeta Components, forked as expcomponents	The kernel's libraries (archive, cache, database, feed, image conversion, mail, WebDAV ...)	"Forked and patched 47+ Zeta Components repos"; "14 security PRs submitted upstream to zetacomponents"; since 6.0.15 the kernel requires the forks `se7enxweb/expcomponents-*`, maintained by 7x and the Foundation under Apache 2.0 (expcomponents)	composer.json, se7enxweb profile README
Symfony, Twig, Doctrine	The platform line	Maintained forks for PHP 8.4 and 8.5	Ecosystem
Upstream layouts packages, Media Site, Site API	Exponential Layouts and Nexus	Ports to the Exponential kernel; a fork of layouts-core	README, Ecosystem
Qbix web server	The engine of Velocity	A production server built on it	Velocity README
AdminNeo	The database source editor		sevenx_dse
jQuery, TinyMCE, Bootstrap Icons	The administration's scripts, editor and icons		October 2026

Timeline

When	What happened	Source
1998	First year of the copyright notice of the Exponential Foundation and 7x	COPYRIGHT.md
1999 to 2014	The copyright years of eZ Systems AS in the code	COPYRIGHT.md
2006	eZpedia starts	https://ezpedia.exponential.earth/
2014	7x's initial idea: the se7enx.com domain is registered quietly and the fork of the entire ecosystem is prepared	7x's account (owner, 18 October 2026); Who is 7x?
2021	"The platform originally known as eZ Publish reached end-of-life in 2021. We kept going."	se7enxweb profile README
2022	7x's project starts	7x's account (owner, 18 October 2026); README.md

When	What happened	Source
December 2023	7x's project is launched and takes over the code base; the first outside change is merged on 11 December	December 2023
1 January 2024	Exponential 6.0.0 (then still named eZ Publish)	release overview
2024 to 2026	Exponential Platform, Legacy, Nexus and DXP repositories are created (February 2024 to March 2026)	GitHub API
July 2025	`COPYRIGHT.md` and `LICENSE.md` are added	July 2025
11 and 12 August 2025	The product becomes Exponential	https://exponential.earth/about/project, August 2025
19 March 2026	The Exponential Foundation is founded in Seattle	https://foundation.exponential.earth/
April 2026	14 security pull requests to Zeta Components	se7enxweb profile README
7 June 2026	Exponential 6.0.14	release overview
September 2026	Exponential Velocity joins Exponential	Velocity chronicle
October 2026	Exponential Intelligence, the LoveStack features and the help system in the 6.0.15 line	6.0.15 changelog
November 2026	The Zeta Components are officially forked as expcomponents (Apache 2.0); the kernel switches to the forks in the 6.0.15 line	expcomponents

How to support or join

- **Sponsor the Foundation**: https://github.com/sponsors/exponentialcms, or "with a sustaining subscription or one time donation" through https://foundation.exponential.earth/. The Foundation says donations are tax deductible in the United States; check with your own tax adviser.
- **Sponsor 7x**: https://github.com/sponsors/se7enxweb, https://sponsor.se7enx.com, https://www.paypal.com/paypalme/7xweb (README).
- **Contribute code**: fork, branch, pull request (Develop with Git, CONTRIBUTING.md).
- **Report what is wrong**: When something goes wrong.
- **Translate and document**: Languages; every help page has "Edit on GitHub".
- **Talk to people**: the forums and Telegram.

Glossary

Term	Meaning
7x	The company that leads Exponential; formerly Brookins Consulting. See Who is 7x?
se7enxweb	7x's GitHub account and Composer vendor name
Exponential Foundation	The nonprofit that operates and funds the Exponential open source project
Exponential	The CMS and framework formerly called eZ Publish; this installation
Exponential 6, 6.0.x	The current major line of Exponential, since 1 January 2024

Term	Meaning
Legacy	The Exponential 6 kernel when it runs beside a Symfony platform ("Referred to as `legacy` in Exponential Platform 5.x and Ibexa OSS", README)
Exponential Platform	The Symfony based line descending from eZ Platform
Dual kernel	A distribution that runs the Symfony platform and Exponential 6 together (Legacy, Nexus)
Nexus	The Exponential Platform distribution with Exponential Layouts, Media Site and Exponential 6
DXP	Digital Experience Platform; here the Symfony-only Exponential Platform DXP
Exponential Basic	The simple CMS of the eZ Publish 2 line
Velocity	Exponential's own PHP web and application server (`vc`)
FrankenPHP	Caddy with PHP built in, the alternative engine
Exponential Intelligence (expai)	The writing assistant that writes drafts
Exponential Layouts (explayouts)	The visual page builder
Extension	A package that adds modules, classes, templates or settings to Exponential
Siteaccess	One way into an installation: a site, a language or the administration
GPL	GNU General Public License; the software's license, version 2 or later
GFDL	GNU Free Documentation License; the documentation's license, version 1.3 or later
CSP	Content Security Policy; tells the browser which scripts and styles a page may use
Packagist	The Composer package directory
eZ Publish	The name the software had while eZ Systems developed it; a product its creator discontinued after almost eighteen years of service to the world. Used only for that history; the product today is Exponential
eZ Platform, Ibexa DXP OSS	The Symfony based successors of eZ Publish by eZ Systems and Ibexa
LoveStack	Mugo Web's distribution of the legacy kernel, whose features 6.0.15 rebuilt

Sources

Repository files (`se7enxweb/exponential`): README.md, COPYRIGHT.md, LICENSE, SECURITY.md, CONTRIBUTING.md, composer.json, release overview, history, ecosystem, CI, Extensions, Languages, Content Security Policy, Styling the administration, 6.0.15 changelog.

Other repositories, read through the GitHub API on 18 October 2026: the `composer.json` and description of `exponential-platform`, `exponential-platform-legacy`, `exponential-platform-nexus`, `exponential-platform-dxp`, `exponential-platform-nexus-starter`, `exponentialbasic`, `exponential-velocity`, `expai`, `exp_adminui`, and the se7enxweb profile README (`se7enxweb/.github`).

7x's own account of its three stages (2014, 2022, 2023), given by the owner of 7x for this page on 18 October 2026.

Public web pages, read on 18 October 2026: https://foundation.exponential.earth/, https://exponential.earth/about/project, https://exponential.earth/, https://se7enx.com/about,

https://ezpedia.exponential.earth/, https://share.exponential.earth/, https://doc.exponential.earth/, https://packagist.org/packages/list.json?vendor=se7enxweb.

Related pages

- Who is 7x?
- The history of Exponential
- Develop with Git
- Free documentation: this page is free to copy, translate and improve.

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. Start with Partners: start here and How to apply.

Copyright and copyleft

Since 29 October 2026 the copyright of Exponential is held by the Exponential Foundation and 7x together, the Foundation named first: "Exponential Copyright ©1998-2026 Exponential Foundation and 7x", in files "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved." Who holds it and why, what the GPL lets you do, the third-party notices that stay and how to write the notice in new files and extensions: Copyright and copyleft.

Why Exponential supports PHP 7.4

Supported - PHP 8.0 to 8.5; usable without support - PHP 7.4 (Exponential 6.0.15 and later).

This page is for site owners, administrators, hosting companies and developers who run, or must run, Exponential on PHP 7.4, and for anyone who wonders why a project that supports PHP 8.5 on the day it appears also goes back to a PHP version that its makers stopped supporting in 2022.

> **Supported: PHP 8.0 to 8.5. Usable without support: PHP 7.4 (Exponential 6.0.15 and later).**
>
> "Without support" means: the software is designed to be usable on PHP 7.4, but the project gives no bug-fix or security commitment specific to PHP 7.4, you use it at your own risk, and the features that need PHP 8 are not available there.

The supported versions stay PHP 8.0 to 8.5, and the newest PHP stays fully supported; usability on PHP 7.4 is added below them, not instead of anything.

Contents

1. In short
2. Who it is for
3. What works on PHP 7.4 and what needs PHP 8
4. How it works
5. A short history of PHP support in Exponential
6. Why the project does this

7. How security is kept
8. The honest trade-offs
9. The end-of-life stance
10. Installing on PHP 7.4
11. Related pages

In short

Question	Answer
Which PHP versions?	Supported: 8.0 to 8.5. Usable without support: 7.4. `composer.json` declares `"php": "^7.4 || ^8.0 || ... || ^8.8"`; versions after 8.5 do not exist yet and are not tested.
What is usable on 7.4, without support?	The whole kernel: content, the admin and editor interfaces, templates, the REST API, SOAP, WebDAV, cronjobs, search, the setup wizard and `exp:install`, and the extensions bundled with the kernel.
What needs PHP 8?	Exponential Velocity, Exponential Layouts and the media site (8.0), and a handful of extensions whose releases need 8.0. The table lists each one.
Do I have to choose packages by hand?	No. The kernel requires the metapackage `se7enxweb/exponential-php-floor`; Composer installs the version that fits your PHP and with it the right set of packages.
Is PHP 7.4 safe?	PHP 7.4 has had no security fixes from the PHP project since 28 November 2022. It is safe only where your operating system vendor or host still patches it. See How security is kept.
What does "without support" mean?	No bug-fix or security commitment specific to PHP 7.4; a problem that occurs only on 7.4 may stay unfixed. You use it at your own risk.
How long?	Usability on PHP 7.4 is a bridge, not a destination. See The end-of-life stance.

Who it is for

Using Exponential on PHP 7.4, without support, is for organizations that want a current Exponential but **cannot change the PHP version of the machine it runs on yet**. In our experience they are rarely careless; they are usually bound by something outside their control:

- **Hosting contracts and managed platforms.** A shared or managed host decides the PHP versions it offers, and a contract may run for years.
- **Long-term operating system releases.** Distributions with long support periods shipped PHP 7.4 as their standard PHP: Debian 11, Ubuntu 20.04 LTS, and the `php:7.4` stream of Red Hat Enterprise Linux 8 and its rebuilds. Organizations with extended support contracts for these systems receive patched PHP packages from their vendor.
- **Certified and audited environments.** In the public sector, health care, finance and industry, a server image may be certified as a whole. Changing PHP means a new certification, planned months ahead.
- **Remote, offline and low-bandwidth sites.** An installation in a remote office, on a ship, in a research station or on a network without Internet access is updated when someone can get there.
- **Other applications on the same server.** A server that also runs software which only works on PHP 7.4 cannot move until that software can.

For all of them the choice used to be: stay on an old Exponential with known bugs, or wait. Now they can run the current release, with its fixes and features, and move to PHP 8 when their circumstances allow, without

moving Exponential again at that moment.

What works on PHP 7.4 and what needs PHP 8

The **kernel** (everything in `kernel/` and `lib/`) and the **extensions shipped with it** (`ezjscore`, `ezoe`, `ezformtoken`, `expservices`, `expimageeditor`, `exphelpsite`) run on PHP 7.4. For the extension packages and servers, the table shows the oldest PHP each one needs. "7.4 (older release)" means Composer installs, on PHP 7.4, the last release of the same line that still runs there; on PHP 8 you get the newest one.

Feature or package	PHP 7.4 (without support)	PHP 8.0	PHP 8.1 to 8.3	PHP 8.4 and 8.5
Kernel: content, admin, editor, templates, REST v1 and v2, SOAP, WebDAV, cronjobs, search	yes	yes	yes	yes
Setup wizard, kickstarter, `exp:install`	yes	yes	yes	yes
Apache or nginx with PHP-FPM or mod_php	yes	yes	yes	yes
PHP's built-in server (`exp:velocity start --engine=php`)	yes	yes	yes	yes
Exponential Velocity, the application server (`--engine=qbix`)	no: `exp:velocity start` refuses with a message	yes	yes	yes
FrankenPHP engine	brings its own PHP 8 binary	yes	yes	yes
Exponential Layouts (`explayouts` and its packages), the expsite packages, the media site theme	no	no	yes	yes
Two-factor authentication (`sevenx_authentication_2fa`)	no	yes	yes	yes
`exp_adminui`, `expai`, `expqueue`, `exp_enhanced_link`, `expchangeclass`, `cjw_themes_jumper`, `cjw_tmv`	no	yes	yes	yes
`sqliimport` 2.x, `ezodf` 7.x, `ezownerchange` 1.4	no	yes	yes	yes
Tags (`eztags`), CJW Newsletter, Syndication, `bccie`, `git_manager`	7.4 (older release)	yes	yes	yes
`xrowextract`, `sevenx_dse`	no (their code needs PHP 8)	yes	yes	yes
reCAPTCHA library (`google/recaptcha`)	1.2.4	1.3	1.3	2.x
All other extensions the kernel installs (ezflow, ezwebin, ezdemo, recaptcha, hcaptcha, ezmultiupload, ...)	yes	yes	yes	yes
AVIF image output	no (WebP is the best format)	no	yes	yes
File times inside the log viewer's ZIP download	the time of the download	the original times	the original times	the original times

Nothing in this table fails silently with a white page. A package that cannot run on your PHP is simply not installed by Composer, and the one server that would need PHP 8, Velocity, says so:

```
$ php bin/php/console exp:velocity start
the Velocity engine (qbix) needs PHP 8.0 or later, this is PHP 7.4. Exponential runs on PHP 7.4 w
ith Apache or
nginx and PHP-FPM, or with the php engine (exp:velocity start --engine=php)
```

How it works

Three pieces make one code base run from PHP 7.4 to PHP 8.5.

1. The kernel is written for the oldest supported PHP. Every PHP file must be valid PHP 7.4: no union types, no nullsafe operator, no `match`, no named arguments, no constructor property promotion, no enums, no `readonly`. Typed properties and arrow functions, which PHP 7.4 introduced, are fine. A newer function is used only behind `function_exists()` or through a polyfill. The rule is checked by `make lint74-all` and by a PHP 7.4 job in continuous integration, which lints every file, installs the production dependencies with Composer on PHP 7.4 and loads every kernel and extension class.

For 6.0.15 this meant: the namespace `Exponential\View\Kernel\Class` became `Exponential\View\Kernel\ContentClass` (`class` is a reserved word before PHP 8.0; the old name still works through `class_alias`), a few `Type|null` parameters became `?Type`, three nullsafe operators became plain checks, `mixed` disappeared from two magic methods, and a few runtime differences got a fallback: `WeakMap` (8.0) falls back to `SplObjectStorage`, a GD image is a resource on 7.4 and an object on 8, a cURL handle must stay open until its last use, and `ZipArchive::setMtimeName()` exists only from 8.0.

2. Polyfills fill in the functions. `symfony/polyfill-php80` and `symfony/polyfill-php81` define `str_contains()`, `str_starts_with()`, `get_debug_type()`, `array_is_list()` and the other small functions added in PHP 8.0 and 8.1, on the PHP versions that lack them; on newer PHP they do nothing. `lib/phpcompat.php` covers the few the kernel needs beyond that.

3. One metapackage per PHP range. The kernel's `composer.json` requires `se7enxweb/exponential-php-floor` at `^1.0`. Each version of that metapackage belongs to one PHP range and requires the packages that run there:

Metapackage version	PHP	What it adds
1.0.1	7.4 (usable without support)	the last PHP 7.4 releases of the lines that moved to PHP 8 (eztags ~2.4.23, cjw_newsletter ~4.2.14, syndication ~1.3.10, bccie ~1.1.18, git_manager ~2.0.22) and google/recaptcha ~1.2.4; 1.0.0 also required xrowextract and sevenx_dse, whose code does not run on 7.4
1.1.0	8.0	the extensions that need PHP 8.0, Velocity included, and google/recaptcha ~1.3.1
1.2.0	8.1 to 8.3	all of 1.1.0, plus Exponential Layouts, the expsite packages and the media site theme; google/recaptcha ~1.3.1
1.3.0	8.4 and later	the same set with the current google/recaptcha 2.x
1.1.4	8.0	all of 1.1.0 plus Exponential Layouts, the expsite packages and the media site theme, which run on PHP 8.0; 1.1.0 to 1.1.3 left them out

Composer always installs the highest version whose PHP requirement the running PHP meets. The result: the same `composer require se7enxweb/exponential` (or `composer create-project` of a skeleton) gives a complete, consistent installation on every supported PHP, and nobody has to know which extension needs which PHP. The source of the metapackage is se7enxweb/exponential-php-floor; its README lists each version.

The technique is not new to the project: the Exponential Platform DXP 4.6 line uses the same kind of metapackage, `se7enxweb/exponential-platform-dxp-php-floor`, to select maintained forks of its libraries on older PHP versions.

A short history of PHP support in Exponential

Release	PHP
eZ Publish 4.x and 5.x (historical, discontinued)	PHP 5, later PHP 7 for the legacy kernel
up to 6.0.7	PHP 7.4 and 8.1
6.0.8 (April 2025) to 6.0.14	PHP 8.1 and later; PHP 8.4 support began in 6.0.8, PHP 8.5 support in 6.0.11
6.0.15	Supported: PHP 8.0 again (the stock PHP of Red Hat Enterprise Linux 9) to 8.5; usable without support: PHP 7.4

The step to PHP 8.1 in 6.0.8 was right at the time: it let the project clean up years of PHP 5 and PHP 7 code quickly. Once that work was done, the same review showed that very little of the kernel actually needed PHP 8. Only a handful of language features stood between the code and PHP 7.4, so going back cost days, not months. That is why the supported floor came down to 8.0 in 6.0.15, and the software was made usable, without support, on 7.4.

Why the project does this

Exponential grew out of a content management system that organizations have run for more than two decades. Many of them still do. The engineers at 7x, who maintain Exponential, have spent most of that time helping such organizations, and they have seen the same pattern many times: a site is not held back by a lack of will, but by a contract, a certification, a platform or a budget cycle that nobody on the project can change this year.

The project's values follow from that experience:

- **Meet people where they are.** Software that only runs on the newest platform helps the organizations that need help least. Keeping the software usable on an older PHP version, even without support, lets the others take part, get the current fixes and features, and plan their own move.
- **Separate the application upgrade from the platform upgrade.** Changing Exponential and changing PHP at the same time doubles the risk. With PHP 7.4 support, a site first moves to the current Exponential on the PHP it has, and later moves to PHP 8 with an application that already runs there. Each step can be tested and rolled back on its own.
- **Never trade the future for the past.** Supporting PHP 7.4 does not hold back PHP 8.5: the newest PHP is supported from its first release, and every feature that needs PHP 8 is still built. The old version is carried by a small set of rules and tools, not by giving anything up.
- **Be honest about the cost.** Older PHP has real drawbacks. This page names them instead of hiding them.

How security is kept

Exponential on an old PHP is only as safe as the whole stack, and on PHP 7.4 that stack is unsupported. This is how each layer is kept safe.

PHP itself. The PHP project ended security support for PHP 7.4 on 28 November 2022. Run PHP 7.4 only where someone still fixes it: the PHP packages of an operating system with long-term support (for example Debian LTS or extended support, Ubuntu Pro, or the PHP 7.4 stream of Red Hat Enterprise Linux 8 and its rebuilds within their support periods), or a host that backports security fixes. An unpatched PHP 7.4 downloaded from somewhere is a risk no application can remove.

Exponential's own code. The kernel is one code base for every PHP version. Security fixes go into it once and reach PHP 7.4 and PHP 8.5 alike, although the project makes no commitment specific to 7.4; there is no separate, older branch that could fall behind.

Dependencies. Composer's security blocking stays on: Composer refuses package versions with known security advisories during resolution, on every PHP version. The metapackage only selects versions that Composer accepts. Where an upstream library dropped an old PHP version before fixing a vulnerability, the project maintains patched forks of the last compatible line (the approach already used for the DXP 4.6 line, for example for Twig 3.11 and graphql-php 14.11) instead of pinning a vulnerable release.

Extensions. On PHP 7.4 a few extensions are installed in the last release of their line that still runs there. These releases are days or weeks older than the newest ones, not years; when a security fix lands in such an extension, the project publishes a release of that line for PHP 7.4 too, or a new version of the metapackage that requires it.

Polyfills. The polyfills come from the Symfony project, are widely used and are maintained independently of Exponential. They only add functions; they change nothing on PHP versions that already have them.

The honest trade-offs

- **Fewer features.** Velocity, Layouts, two-factor authentication and the other packages in the table are not available on PHP 7.4. For many small and medium sites that is acceptable; for others it is the reason to move to PHP 8 first.
- **Older extension releases.** Some extensions are one or a few releases behind on PHP 7.4.
- **Less testing.** The full test suite runs on PHP 8.1 to 8.5. On PHP 7.4 the continuous integration checks the syntax of every file, a production install and the loading of every class. The project's own test runs on PHP 7.4 use PHPUnit 9.6, the last PHPUnit for PHP 7.4, which does not read the PHP 8 attributes some tests use.
- **Slower PHP.** PHP 8 is faster than PHP 7.4, often by a wide margin with OPcache and the JIT. A busy site gains speed from the move alone.
- **Security depends on your vendor.** See above: an unpatched PHP 7.4 is a risk no application can remove.

The end-of-life stance

Usability on PHP 7.4 is a **bridge**, without support. It exists so that organizations can run a current Exponential while they prepare their move to PHP 8. The project's commitments:

- Usability on PHP 7.4 is not removed in a patch release. When the time comes, it is announced in the changelog and in this documentation at least one release ahead, and the release that removes it says so in its notes.
- The deciding factor is security: as long as maintained, patched PHP 7.4 packages are widely available from operating system vendors and the dependencies can be kept free of known vulnerabilities on PHP 7.4, the bridge stays.

- The newest PHP is never dropped or delayed to keep an old one.
- Every site on PHP 7.4 should have a plan to move. The PHP upgrade runbook shows the steps; with Exponential already current, the move is a change of PHP and a `composer update`, nothing more.

Installing on PHP 7.4

1. Check the machine with 2. Requirements: PHP 7.4 with the required extensions for the web server **and** the command line (the same version and extensions in both).
2. Install the 6.0.15 line, as on any PHP version (chapter 3). Run Composer with the PHP 7.4 that will serve the site, so it selects the PHP 7.4 packages.
3. Serve the site with Apache or nginx and PHP-FPM, or with `exp:velocity start --engine=php` for a quick test.
4. When the machine moves to PHP 8, run `composer update` with the new PHP: Composer moves the metapackage to the version for that PHP and installs the packages that were left out.

Related pages

- 2. Requirements: every PHP version, extension and setting
- Operating system support
- PHP 8.0 support: the fallbacks for PHP 8.0, which PHP 7.4 shares
- Supported PHP versions: the rules for code and extensions
- PHP upgrade runbook
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

Develop with Git: grow from user to developer

The recommended way to build websites with Exponential.

You know your way around the administration, you have published content, maybe you have changed a template. This page helps you take the next step: working on the code of your site the way the people who build Exponential do. The tool for that is **Git**, and the most useful single decision you can make is to install Exponential **as Git clones with their full history**, the kernel and every extension, instead of as release archives.

> **Warning**
>
> `.htaccess` **slows down every request.** When `.htaccess` files are enabled (`AllowOverride` anything but `None`), let alone used, Apache looks for them and reads and evaluates them on every request, in every directory of the path, and page loading times go up drastically (measured). The only real fix is to switch the feature off for the virtual host: `AllowOverride None` in the HTTPS virtual host, and the rules in the virtual host's own configuration as mod_rewrite rules. That is far faster in production speed tests and well worth the time it takes to optimize a production configuration. Switch over right away: copy the rules of the example HTTPS virtual host into your virtual host (with the static cache, also `staticcache_offload.conf`), set `AllowOverride None`, run `apachectl configtest`, reload Apache and test the site. The same rules for other servers: nginx, LiteSpeed, Caddy, IIS, Velocity (built in); where virtual host rules go in a control panel: Plesk, cPanel, DirectAdmin, ISPConfig. **On LiteSpeed**

> **or OpenLiteSpeed,** the server's own page cache can also answer anonymous visitors and crawlers without PHP: Page caching with LiteSpeed and OpenLiteSpeed.

Nothing here needs you to be an expert. Each step shows the command to type and what you should see, and every command on this page was tried on a fresh machine before it was written down.

Why Git clones and not archives

Composer can install a package in two ways:

	Release archive ("dist")	Git clone ("source")
What you get	the files of one version, nothing else	the files **and** the complete history of the project
Size and speed	small, fast	larger, a few minutes more on the first install
See what you changed	`composer status -v` names changed files	`git status` and `git diff` show every change, line by line
Find out why a line is there	not possible	`git log` and `git blame` show who changed it, when and why
Find which change broke something	not possible	`git bisect` finds it in a few steps
Use a fix before it is released	copy files by hand	`git cherry-pick` or check out the commit
Keep your own changes	lost at the next update	kept on your own branch; Composer refuses to overwrite them
Send a fix back to the project	not possible	push a branch to your fork and open a pull request

Archives are fine for a server that only runs a finished site. For **building** a website, Git clones are the recommended way: they turn every package into something you can read, search, understand and improve.

What you need

- **Git** 2.28 or later (`git --version`).
- **Composer** 2 (`composer --version`), and PHP as listed in Requirements.
- A free **GitHub account**, only when you want to send changes back. Installing needs none.

The kernel is github.com/se7enxweb/exponential, and every extension has its own repository in the se7enxweb organization. They are published on Packagist under the same names, so Composer can fetch each one either as an archive or as a clone.

Step 1: tell Composer to prefer clones

One setting makes Composer install every `se7enxweb/*` package as a Git clone, and everything else (the Zeta Components, the libraries) as a fast archive. Set it once for your user account:

Shell

```
composer config --global 'preferred-install.se7enxweb/*' source
```

Keep the quotes around `se7enxweb/*`, so that your shell does not try to expand the `*`. To see the setting:

Shell

```
composer config --global preferred-install
```

```
{"se7enxweb/*":"source","*":"dist"}
```

If you prefer to keep the setting with one project instead (for example so that everyone on your team gets it), run the same command without `--global` inside the project. It is then written into that project's `composer.json`.

Step 2: install with Git

A new site

Shell

```
git clone https://github.com/se7enxweb/exponential.git exponential
cd exponential
git switch -c my-site
composer install
```

- `git clone` fetches the kernel with its whole history: more than 27,000 commits, a download of about 370 MB. On a normal connection that takes two to three minutes.
- `git switch -c my-site` starts **your own branch** for your site's changes, on top of `main`, the current development line. To build on a release instead, name its tag: `git switch -c my-site v6.0.14`. The release tags are listed with `git tag -l 'v*' --sort=version:refname`; read Versions, tags and branches before choosing one, because older tags need a newer PHP for their test tools.
- `composer install` then clones every extension the kernel requires into `extension/<name>`, each with its own history (on `main` that is 67 repositories, a few minutes), and fetches the libraries into `vendor/` as archives.

Then install the site as usual: Choosing an install method or the one command `php bin/php/console exp:install --apply` from the console install.

For a local development site under Apache, put the rules and the constants into place from the console. A copy of `config.php-RECOMMENDED` sets nothing, so the built-in defaults apply, which suit development (INI changes are seen at once); `exp:config:list` shows what could be set:

Shell

```
php bin/php/console exp:setup:status
php bin/php/console exp:setup:htaccess --apply
php bin/php/console exp:setup:config --apply
php bin/php/console exp:config:list
```

The clone tracks only the templates (`.htaccess_root`, `.htaccess_root_static`, `config.php-RECOMMENDED`); the copies show up as untracked files in `git status`. Never commit `config.php`: it can hold a password. In a repository of your own site, add `/config.php` to its `.gitignore`. Details: Setting up .htaccess and config.php from the console.

Prefer `composer create-project`? It can make clones too, but you need two options together:

Shell

```
composer create-project se7enxweb/exponential exponential --prefer-source --keep-vcs
```

Without `--keep-vcs`, Composer deletes the kernel's `.git` folder at the end and you are left without history. Without a version, Composer picks the newest release, so the kernel is then a clone sitting on that release's tag ("HEAD detached at v6.0.14"); run `git switch -c my-site` in it before you change anything. Here `--prefer-source` makes every package a clone, the libraries included; with the setting of step 1 you can leave it out and get clones for the `se7enxweb/*` packages only.

Check that it worked

Shell

```
composer status -v
ls -d extension/*/.git | wc -l
git -C extension/eztags log --oneline -3
git -C extension/eztags status
```

`composer status -v` should say `No local changes`, the second command counts the extensions that are clones, and the last two show the newest commits of one extension and where it stands:

```
HEAD detached at v2.4.21
nothing to commit, working tree clean
```

"HEAD detached" is normal. Composer checks out the exact release tag your `composer.json` asks for, not a branch. You can read, search and test as much as you like in that state. When you want to change something, start a branch first (see Contribute a change with a pull request).

`git status` in the installation root lists the extensions as untracked folders: they are separate repositories, not part of the kernel's, and that is expected.

Step 3: turn an existing site into clones

A site that was installed from archives can be moved over without reinstalling it. **Make a backup first** (with git_manager's backup manager, or a copy of the folder and the database), and try it on a copy of the site before the live one.

The kernel

Find out which version you run, then connect the folder to the kernel's repository at exactly that version:

Shell

```
php bin/php/console --version
git init -b my-site
git remote add origin https://github.com/se7enxweb/exponential.git
git fetch origin --tags
git reset v6.0.14
git status
```

Use your own version in place of `v6.0.14`; a site installed from `main` uses `git reset origin/main`. `git reset` without `--hard` only moves Git's pointer: **it does not change a single file**. `git status` then shows exactly what differs from the release, which on a site installed from an archive means your own changes and the files your installation created (such as `settings/override/`; most of `var/` is ignored):

```
On branch my-site
 M design/standard/templates/pagelayout.tpl
?? settings/override/
```

Commit the changes you want to keep on `my-site`. The files in `settings/override/` and `settings/siteaccess/` hold database passwords: never push them to a public repository.

The extensions

Shell

```
composer config --global 'preferred-install.se7enxweb/*' source
composer status -v
composer reinstall 'se7enxweb/*'
composer status -v
```

`composer reinstall` removes each `se7enxweb/*` package and installs it again as a clone, at the same version.

> **Warning**
>
> `composer reinstall` deletes the package's folder first. If the first `composer status -v` lists changed files in an extension that is still an archive, copy that folder somewhere safe before you reinstall, then compare it with the clone (`diff -ru <copy> extension/<name>`) and bring your changes back on a branch.

After the reinstall, regenerate the class maps and clear the caches as after any code change:

Shell

```
php bin/php/ezpgenerateautoloads.php -e
php bin/php/ezcache.php --clear-all
```

The administration can do the same without a shell: the ezupdate extension (**Setup** > **Updates and packages**) lets you choose "dist or source" for each install or update run.

Contribute a change with a pull request

Found a bug and fixed it? Send the fix back, so that it is in the next release and you do not have to carry it yourself. This works the same for the kernel and for any extension; the example uses `eztags`.

1. On GitHub, open the repository (github.com/se7enxweb/eztags) and press **Fork**. You now have your own copy, `github.com/<you>/eztags`.
2. Add your fork as a second remote and start a branch:

   Shell

   ```
   cd extension/eztags
   git remote add fork https://github.com/<you>/eztags.git
   git switch -c fix-tag-search
   ```

3. Make the change, test it on your site, and commit it. Start the message with what the change does: `Fixed:`, `Added:`, `Updated:`, `Removed:` or `Renamed:`.

   Shell

   ```
   git commit -am "Fixed: The tag search finds tags with an apostrophe"
   git push -u fork fix-tag-search
   ```

4. GitHub prints a link to open a **pull request**; follow it, describe what you fixed and how you tested it, and send it. (With the GitHub command line tool: `gh pr create`.)

A clone made by Composer already has a remote named `origin` that points at the project; pushes go to your fork instead, so you need no write access to the project. For the kernel itself, read `CONTRIBUTING.md` in the root of the installation first: it explains the commit hook (`make hooks`) and the file manifest (`make manifest-fix`) that every kernel pull request keeps current.

Expect many review rounds. 7x reviews a commit or pull request in many more rounds than usual, sometimes hundreds of revisions; be ready to revise your change until it is simple to accept. A fix for behavior that turns out to be deliberate, a negative feature other sites depend on, needs a clear justification on top, and an option is easier to accept than a new default. While the pull request is open, squash it and rebase it onto `main` as the reviews ask, which 7x requires:

Shell

```
git fetch origin
git rebase -i origin/main
git push --force-with-lease
```

Push with force only to your own pull request branch. After the merge, `main` is never rewritten (Code reviews).

Copyright assignment (CLA). The Exponential Foundation and 7x merge a code contribution to the core kernel (`se7enxweb/exponential`), sent in any form (pull request, patch, issue or mail), only after you have signed the Contributor Licensing Agreement. **By signing and submitting, you assign the copyright in your contribution to the Exponential Foundation and 7x, which become its owners, and give up control over it:** The Exponential Foundation and 7x may use, change, relicense and distribute it in any way, also under other licenses and in commercial products, without asking or paying you, and you cannot withdraw it or set conditions on it. Your own custom extensions, and the branch of your own site, are not affected. See Copyright assignment (CLA) and the CLA page.

Built an extension of your own? Contribute extensions back shows how to publish it on GitHub and Packagist so every site can install it, and what the GPL requires when you hand an extension to others.

Improving the documentation works the same way: these help pages are the files in `doc/` of the kernel. See Free documentation for what you may do with them.

Everyday work with clones

See what changed

Shell

```
git status
git diff
composer status -v
```

`composer status -v` looks into every package at once and names the ones with uncommitted changes, unpushed commits or a different version than the lock file says.

Find out why and when

Shell

```
git log --oneline -10 -- kernel/content/view.php
git blame kernel/content/view.php
git log -S "CheckValidity" --oneline
```

The first shows the last ten changes to one file, the second who last changed each line, the third every

commit that added or removed a piece of text. In an extension, run the same commands inside `extension/<name>` or with `git -C extension/<name> ...`.

Find the change that broke something

Shell

```
git bisect start
git bisect bad
git bisect good v6.0.13
```

Git checks out a version halfway between; test it and answer `git bisect good` or `git bisect bad` until Git names the commit. `git bisect reset` brings you back to where you started.

Use a fix before it is released

When a fix is already in the project's history but not yet in a release:

Shell

```
git -C extension/eztags fetch origin
git -C extension/eztags switch -c with-fix
git -C extension/eztags cherry-pick <commit>
```

While that branch exists only on your machine, Composer refuses to update the package (see When Composer refuses): push the branch to your fork, or go back to the release with `git -C extension/eztags switch --detach v2.4.21` (your release tag) once a release contains the fix.

Keep up to date

The kernel, on your branch, then the extensions to match:

Shell

```
git fetch origin --tags
git merge origin/main
composer update
```

(or merge the newest release tag instead of `origin/main`). Merging keeps your own commits on `my-site`; `composer update`, not `composer install`, because the kernel's `composer.json` changed and your lock file has to follow it. Always Git first (the kernel), then Composer (the extensions). Update with Git has the whole routine, on the command line (Track A) or in the administration with Git Manager and Updates and packages (Track B), for the schedule you choose (hourly, daily, weekly, monthly or only at each release), how to see what changed, and an automatic update. Moving the code is half of an upgrade; Upgrading has the database half.

One extension on its own, after reading its release notes:

Shell

```
composer update se7enxweb/eztags
```

Composer moves the clone to the new tag. Its history stays, and so do your branches.

When Composer refuses

Composer protects your work in a clone. These are the messages you will meet, and what to do:

Composer says	Why	What to do
`Source directory ./extension/eztags has uncommitted changes.`	you changed files and did not commit them	commit them on a branch, or `git -C extension/eztags stash`, run the update, then `git -C extension/eztags stash pop`
`Source directory ./extension/eztags has unpushed changes on the current branch`	you committed on a branch that exists only on your machine	`git push -u fork <branch>` to your fork, then update; your branch is kept
`You have version variations in the following dependencies` (from `composer status`)	the clone is on a branch, not on the tag in the lock file	expected while you work on a fix; `composer update <package>` puts it back on the tag
the package is still installed from an archive	it was installed before the setting of step 1: Composer does not convert packages that are already installed	`composer reinstall <package>` (see step 3)

Three more things worth knowing:

- **Never `git clone` by hand into a folder Composer installed as an archive.** Composer still thinks the folder is an archive and, at the next update of that package, replaces it without asking: your clone, its history and your commits are gone. Let Composer make the clone with `composer reinstall <package>` (see step 3).
- An update can stop half-way when it refuses: the lock file may already name the new version while the folder still has the old one. Fix the cause, then run `composer install` to finish.
- `composer update` moves **every** package to the newest version its constraint allows. On a live site, update one package at a time (`composer update se7enxweb/<package>`), as Getting the code explains.

Where to go next

- Update with Git: stay in sync with GitHub on the schedule you choose.
- Getting the code: every way to install, versions and tags, extensions as Composer packages.
- Extensions and the extension pages.
- Setting up .htaccess and config.php from the console: the local `.htaccess` and `config.php`.
- git_manager: see and change the state of a Git checkout from the administration, and make backups before you change anything.
- When something goes wrong: which issue tracker to use and how to write a good report. The **Report an error** page of this help prepares the issue for you.
- The project on GitHub: se7enxweb/exponential, its issues, and every se7enxweb repository.
- Free documentation: these pages are free to copy, translate and improve.
- Git itself: the free book Pro Git, and GitHub's guide to pull requests.

See also: Ecosystem, PHP Developers, AI, agents and agentic, Software Engineers, Web Framework Developers, CMS Developers, DXP Developers, Integrators.

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. Start with Partners: start here and How to apply.

Logging into the Editor

Sign in to the Editor, the simplified interface for people who write and publish content: its address, the sign-in form, two-step sign-in, a forgotten password, signing out and what to do when it does not work.

This page is for editors: people who write, edit and publish content. Many installations give them the **Editor**, an interface of its own that looks like the administration but shows only what writing and publishing need: the dashboard, the content and the media library. Administrators use the administration instead; both are parts of the same installation and work on the same content.

The address

The Editor has its own address. Which one depends on how your site is set up; your administrator tells you. The usual forms are:

[Image: The sign-in page of the editor's administration]

The sign-in page of the editor's administration

Form	Example
A path after the site's address	`https://www.example.com/editor`
A host name of its own	`https://edit.example.com`

Every page of the Editor needs a signed-in user: whatever address of it you open, you see the sign-in form first.

Bookmark the address in your browser, so you never have to look for it again.

Sign in

[Image: The sign-in page of the Editor, with the fields Username and Password and the Sign in button]

The sign-in page of the Editor, with the fields Username and Password and the Sign in button

1. Type your **Username**. Most sites also accept the **e-mail address** of your account in this field.
2. Type your **Password**. The eye button in the field shows what you typed, to check it; press it again to hide it.
3. Press **Sign in** (or Enter).

The line under the title names the part of the installation you are signing in to, for example *Administration of Editor*.

If your account uses **two-step sign-in**, a second page follows: **Confirm it is you**. Enter the 6-digit code from your authenticator app (or from the e-mail the site sent; the code is also in its subject line) and press **Verify and sign in**. Spaces in the code are fine; each code works once. After too many wrong codes the sign-in ends, and you start again with your password. Setting it up is explained in Your account and signing in.

Your first sign-in

An administrator creates your account in the **Editors** group and gives you the address, your username and a temporary password (the password by a second way, such as a phone call). At your first sign-in the site asks you to replace the temporary password before anything else: enter it as the current password, choose your own, and confirm once more with the temporary one. Every step, with screenshots, is in Setting up accounts for editors and administrators. What your account then allows is described in What the Editor role allows.

After signing in

The Editor opens on its **Dashboard**: a greeting with your name, buttons to **Content structure** and the **Media library**, your numbers (drafts, pending items) and the latest content.

[Image: The dashboard of the Editor after signing in, with the tabs Dashboard, Content structure and Media]

The dashboard of the Editor after signing in, with the tabs Dashboard, Content structure and Media

From here everything in A tour of the administration applies: the tree, the item pages, the edit form. What you may open and change is decided by your **role** (usually *Editor*): an editor sees fewer tabs and buttons than an administrator, and pages for settings, users and the system are not there.

In the right column, **Interface language** chooses the language of the menus and buttons for you; see Languages.

When signing in does not work

What you see	What to do
The system could not log you in. Make sure that the username and password is correct.	Check the spelling and the caps lock key: letters must be entered in the correct case. Try your e-mail address instead of the username.
... is not allowed to log in because failed login attempts by this user exceeded allowable number of failed login attempts!	The site locks an account after a number of wrong passwords. Ask an administrator to reset the failed sign-ins of your account.
Access denied! You do not have permission to access ...	Your account is right, but your role may not use this part of the installation. Ask an administrator.
The page asks for a user name and password in a small browser window, before the site appears	Some test and staging servers are protected by an extra browser password. It is not your Exponential account; ask whoever gave you the address.
You are sent back to the sign-in form after signing in	Your browser blocks cookies for the site. Allow cookies for it (signing in needs one), or try a private window to rule out an extension.

[Image: The sign-in page with the message that the system could not log you in]

The sign-in page with the message that the system could not log you in

Forgot your password?

1. On the sign-in page, choose **Forgot your password?**.
2. Enter the **e-mail address** of your account and press **Generate new password**.
3. Open the mail you receive and follow its link. The link works **once** and, by default, for **one hour**; the mail says how long. Opening it gives your account a new password, which a second mail sends you.
4. Sign in with it and change it at once under **Change password**.

The page answers the same whether or not an account has that address. No mail after a few minutes? Check the spam folder, and make sure you used the address of your account. A link opened too late says *This link has expired.*; ask for a new one.

Staying signed in and signing out

You stay signed in while you work. A sign-in belongs to one browser and to this part of the installation: being signed in to the Editor does not sign you in to the administration or the public site, and the other way round. A session that is not used for a long time (360 days with the shipped settings) ends, and you sign in again.

To sign out, press the last button of the header, an arrow leaving a box (**Logout**). While an item is open for editing it is grayed out: store or discard your draft first. Always sign out on a shared computer.

Keep your account safe

- Use your **own** account, never a shared one, so that your work carries your name and the version history and the audit trail tell who did what.
- Use a long password that you use nowhere else; a password manager remembers it for you. The site refuses short and well-known passwords.
- Turn on **two-step sign-in**: your password alone is then useless to anyone else. See Your account and signing in.
- Never send your password by e-mail or chat, not even to an administrator; nobody needs it.

Learn more

- Getting started
- Your account and signing in
- Logging into the admin, for administrators
- Setting up accounts for editors and administrators, for administrators
- Managing members

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. Start with Partners: start here and How to apply.

Update with Git: stay in sync with GitHub

Keep a site made of Git clones up to date, first with Git (the kernel) and then with Composer (the extensions), either on the command line or in the administration during a maintenance window.

An update of a site made of Git clones (Develop with Git) is always two moves, in this order:

1. **Git** brings the **kernel** up to date: the new commits from GitHub are merged into your site's branch. That changes the kernel's files, and with them `composer.json`, the list of extensions and their versions.
2. **Composer** brings the **extensions** up to date to match the new `composer.json`: new extensions are cloned, existing clones move to their new release tags.

Both moves can be made on the **command line** (Track A) or in the **administration** (Track B). This page describes both tracks side by side with the same seven steps, says what each can and cannot do, and how to mix them.

Command line or administration?

	Track A: command line	Track B: administration
For	developers and administrators with a shell on the server (SSH)	administrators without a shell, or who prefer pages and buttons
Needs	Git, Composer and PHP on the server, as the user that owns the files	the extensions `git_manager` and `ezupdate` active, a role with their policies (see the table in Track B)
Kernel (Git)	`git merge` of anything: the newest **release tag**, `origin/main`, another branch or one commit	**Git Manager** checks out a **branch** and pulls it (`git pull origin <branch>`); it cannot merge a release tag
Extensions (Composer)	`composer update`, run by you	**Updates and packages** runs Composer for you, in the background on the server, as the user the web server's PHP runs as: a full update (**Preview**, then **Run it for real**) or the packages you tick (**Install updates**). Off until `[UpdateSettings] AllowUpdate` is switched on
Database update of a release	the release's SQL file with your database's own client (`mysql`, `psql`, `sqlite3`)	**DS Editor**: run SQL or import the file, on the live database
Caches and code reload	`php bin/php/ezcache.php --clear-all`, or `php bin/php/console exp:velocity deploy` on a Velocity site (caches, PHP-FPM, Velocity restart)	**Cache management**: every cache and Velocity's response cache; it cannot restart Velocity, which a code update needs on a Velocity site
Conflicts and local changes	you see them and resolve them (`git merge --abort`, `git stash`, a commit)	Git Manager shows Git's output; resolving a conflict or committing changes needs a shell
Automatic, on a schedule	yes: Let it update itself	no

In short: Track A can do everything. Track B covers a routine update of a site that follows a branch (`main`, or your own branch that you merge releases into) without opening a shell, but a release tag, a merge conflict and the Velocity restart still need the command line.

Choose how closely you follow

You want	Follow	Good for
every change as soon as it is pushed	`main`, the development line: check **hourly** or **daily**	development machines, staging sites, people who test and contribute
changes in batches	`main`, checked **weekly** or **monthly**	sites that change often but want quiet weeks
only finished, tested versions	the newest **release tag** (`v6.0.14`, ...)	live sites (Track A)

Release tags are on the history of `main`, so you can start with releases and move to `main` later, or the other way round once the next release has caught up.

The seven steps

Step	What	Track A: command line	Track B: administration
1	Back up, start the maintenance window	A1	B1
2	See what is waiting	A2	B2
3	Update the kernel with Git	A3	B3
4	Update the extensions with Composer	A4	B4
5	Check files and database, apply the database update	A5	B5
6	Clear the caches, reload the code	A6	B6
7	Check the site, leave the window	A7	B7

Track A: the command line

Run every command in the root of the installation, as the user that owns the files (not as root: as root, the scripts of `bin/php/` ask for `--allow-root-user`). The pictures are the real output of the commands in a test installation: a kernel clone on the branch `my-site`, made from release `v6.0.13` with every extension a Git clone, updated to `main`. The captures were made as root, so they show `--allow-root-user`; the two-line warning that root gets from every script is left out.

A1. Back up and start the maintenance window

Back up the database (with your database's dump tool) and the folder. Then take the public site offline:

Shell

```
php bin/php/maintenance.php on --message="Updating, back at 06:00" --until=1h
php bin/php/maintenance.php status
```

[Image: Terminal: maintenance.php on prints "Maintenance is ON: page requests get the maintenance page (503)", and status shows since, until, the message and the page served]

Terminal: maintenance.php on prints "Maintenance is ON: page requests get the maintenance page (503)", and status shows since, until, the message and the page served

`maintenance.php` is part of 6.0.15 (`main`); on an older release skip this step or switch maintenance on in the administration (B1), which works as soon as the new code is in place.

A2. See what is waiting

Shell

```
git status --short --untracked-files=no
git fetch origin --tags
git log --oneline HEAD..origin/main | head -5
git log --oneline HEAD..origin/main | wc -l
git tag -l 'v6.*' --sort=version:refname | tail -3
```

[Image: Terminal: git status prints nothing (no uncommitted changes), git fetch, the five newest commits on origin/main the site does not have, their count (5651 from v6.0.13), and the three newest release tags v6.0.12, v6.0.13, v6.0.14]

Terminal: git status prints nothing (no uncommitted changes), git fetch, the five newest commits on origin/main the site does not have, their count (5651 from v6.0.13), and the three newest release tags v6.0.12, v6.0.13, v6.0.14

`git status` must print nothing: changes you did not commit can stop the merge. `HEAD..origin/main` lists what the development line would bring; for a release, `HEAD..v6.0.14` does the same. Always sort tags by version (by name, `v6.0.10` comes before `v6.0.8`). `composer outdated 'se7enxweb/*'` names the extensions that have a newer release.

A3. Update the kernel with Git

Following the development line:

Shell

```
git merge --no-edit origin/main
```

Following releases only, the newest release tag:

Shell

```
git merge --no-edit "$(git tag -l 'v6.*' --sort=version:refname | tail -1)"
```

[Image: Terminal: the end of git merge origin/main, listing new files such as the 6.0.15 database update files, and git log -1 showing the merged commit]

Terminal: the end of git merge origin/main, listing new files such as the 6.0.15 database update files, and git log -1 showing the merged commit

`git merge` keeps your own commits on `my-site` and adds the new ones. When one of your commits and the update change the same lines, Git stops with `CONFLICT`; `git merge --abort` puts everything back.

On a clone without a branch of its own (`HEAD detached at v6.0.14`), switch to the new release instead: `git switch --detach "$(git tag -l 'v6.*' --sort=version:refname | tail -1)"`.

A4. Update the extensions with Composer

Shell

```
composer update
composer status -v
```

[Image: Terminal: composer update reports "Lock file operations: 39 installs, 11 updates, 1 removal", upgrades such as se7enxweb/ezflow v6.0.2 to v6.1.10, new extensions cloned from cache, then composer status says "No local changes" and 66 extensions are Git clones]

Terminal: composer update reports "Lock file operations: 39 installs, 11 updates, 1 removal", upgrades such as se7enxweb/ezflow v6.0.2 to v6.1.10, new extensions cloned from cache, then composer status says "No local changes" and 66 extensions are Git clones

`composer update`, not `composer install`: the kernel's `composer.json` changed, and your lock file (it is not part of the kernel's repository) has to follow it. From `v6.0.13` to `main` this added 39 packages and updated 11, each new `se7enxweb/*` package as a Git clone thanks to the `preferred-install` setting of Develop with Git. The class maps are regenerated at the end.

What Composer does with clones:

- An update keeps the way a package was first installed: a clone stays a clone, an archive stays an archive, even with `--prefer-source`. `composer reinstall <package>` turns an archive into a clone (Develop with Git).
- It never overwrites your work: a clone with uncommitted changes, or with commits on a branch that exists only on your machine, stops it with a message (When Composer refuses). Fix the cause and run `composer update` again to finish.

A5. Check and update the database

Shell

```
php bin/php/checkmanifest.php --all
ls update/database/mysql/6.0/
```

[Image: Terminal: checkmanifest.php --all prints "share/filelist.md5 matches", and the 6.0 database update files dbupdate-5.4.0-6.0.0.sql and dbupdate-6.0.0-6.0.15.sql]

Terminal: checkmanifest.php --all prints "share/filelist.md5 matches", and the 6.0 database update files dbupdate-5.4.0-6.0.0.sql and dbupdate-6.0.0-6.0.15.sql

`checkmanifest.php` `--all` compares every file with the release's checksums; after a clean Git update it reports that they match. A release can need a **database update**: Upgrading names the file for each release and engine (`update/database/<engine>/6.0/...`). Apply it once, whole, in order, with your database's client, for example:

Shell

```
mysql -u USER -p DATABASE < update/database/mysql/6.0/dbupdate-6.0.0-6.0.15.sql
```

(the test installation has no database, so there is no picture of this step).

A6. Clear the caches and reload the code

Shell

```
php bin/php/ezcache.php --clear-all
```

[Image: Terminal: ezcache.php --clear-all clearing all caches, starting with the content view cache, the HTTP cache and the INI caches]

Terminal: ezcache.php --clear-all clearing all caches, starting with the content view cache, the HTTP cache and the INI caches

On a site served by Exponential Velocity, run `php bin/php/console exp:velocity deploy` instead: it regenerates the class maps, clears the caches, lets PHP-FPM see the new code and restarts Velocity, in the right order (Velocity engines).

A7. Check the site and leave the window

Shell

```
git log --oneline ORIG_HEAD..HEAD | wc -l
php bin/php/console --version
php bin/php/maintenance.php off
```

[Image: Terminal: 5651 commits came with the update, the console reports Exponential 6.0.15stable, and maintenance.php off prints "Maintenance is OFF: the site answers again"]

Terminal: 5651 commits came with the update, the console reports Exponential 6.0.15stable, and maintenance.php off prints "Maintenance is OFF: the site answers again"

After a merge, `ORIG_HEAD` is where you were before it: `git log --oneline ORIG_HEAD..HEAD` lists what the update brought. Open the front page and the administration, and read the logs (`var/log/error.log`, or **Setup** > **Log files** > **Error digest** in the administration) before switching maintenance off. To go back instead: `git reset --hard ORIG_HEAD`, then `composer update` (from `main` back to `v6.0.13` that removes the 39 added packages and moves 11 back), and restore the database from the backup if a database update ran.

Track B: the administration

The same seven steps on the pages of the administration. The screenshots are of the **Admin 4** design with a test administrator; host names, server paths and addresses are replaced by examples. Each page needs a policy (`module/function`); the Administrator role has them all.

Step	Page	Policy
B1	**Setup** > **Backup** (`git_manager/backup`), **Setup** > **Maintenance** (`setup/maintenance`)	`git_manager/backup`, `setup/administrate`
B2, B3	**Setup** > **Git Manager** (`git_manager/dashboard`)	`git_manager/git_manager` (`git_manager/push` to push)
B4	**Setup** > **Updates and packages** (`update/dashboard`)	`update/ezupdate` to look, check and preview; `update/manage` to install, switch updating on and change servers
B5	**Setup** > **Upgrade check** (`setup/systemupgrade`), **Setup** > **DS Editor** (`dse/dashboard`)	`setup/setup`, `dse/dse`
B6	**Setup** > **Cache management** (`setup/cache`)	`setup/managecache`
B7	**Setup** > **Log files** (`setup/logs`)	`setup/logs`

Git Manager, Updates and packages and the DS Editor are extensions (`git_manager`, `ezupdate`, `sevenx_dse`); they appear in the menu when they are active.

B1. Back up and start the maintenance window

Make a backup: **Setup** > **Backup**, type **DB + Files caption**. Then open **Setup** > **Maintenance**. While the site is online, the page shows a green **The site is online** banner and a plan in three steps:

1. **Message for visitors** (optional; left empty, the page says "We are working on the site and will be back shortly.").
2. **Expected duration**: type the minutes or choose **15 min**, **30 min**, **1 h**, **2 h** or **4 h**. Maintenance does not end by itself.
3. **Who still gets in**: the administration under `/admin` always; tick **Let my own address still see the site** to check your work while everyone else gets the maintenance page.

Tick **I understand that the public site goes offline for every visitor not listed above, until I switch maintenance off again.** and press **Take the site offline**. The **Preview** shows the page visitors get.

[Image: Setup > Maintenance with the site online: the green banner, the three steps of the plan (message, expected duration, who still gets in) and the preview of the 503 page visitors will see]

Setup > Maintenance with the site online: the green banner, the three steps of the plan (message, expected duration, who still gets in) and the preview of the 503 page visitors will see

B2. See what is waiting

Open **Setup** > **Git Manager** and press **Fetch now** in the **Upstream (origin)** card. It runs `git fetch --prune` and nothing else. The card then shows the remote, the branch it is compared with, **AHEAD / BEHIND** (local commits / commits to get), the files **NOT COMMITTED**, and **Missing in this installation**: the commits the update brings, each with its author and a link. Files you changed and did not commit can make the next step fail; commit them from a shell first.

[Image: Setup > Git Manager: the Upstream card with Fetch now, the remote, the compared branch, ahead and behind, the commits missing in this installation, and below it the Local branch and Remote branch boxes with Regenerate autoloads and Checkout]

Setup > Git Manager: the Upstream card with Fetch now, the remote, the compared branch, ahead and behind, the commits missing in this installation, and below it the Local branch and Remote branch boxes with Regenerate autoloads and Checkout

Updates and packages > **Overview** > **Check for updates** shows which extensions have a newer release; it changes nothing.

B3. Update the kernel with Git Manager

Under **Local branch**, choose the branch the site runs on (the one shown after **HEAD**, for example `main` or `my-site`), tick **Regenerate autoloads**, and press **Checkout**. Git Manager checks out that branch and runs `git pull origin <branch>`: it fetches the branch of the same name on GitHub, merges it and regenerates the class maps. Git's output is shown on the page; the badge then says **Up to date** and **AHEAD / BEHIND** shows `0 / 0`.

Git Manager follows **branches**. It cannot merge a release tag into your branch; for release-only updates use A3. The **Commits log** further down lists the history, 50 commits to a page, with a filter by author and date; a commit can be checked out by its hash (to go back).

B4. Update the extensions with Updates and packages

Updates and packages runs Composer for you: in the installation's folder, in the background, as the user the web server's PHP runs as, with a time limit. It does what `composer update` does in A4. Updating is **Switched off** until you choose otherwise (`[UpdateSettings] AllowUpdate`), because Composer rewrites `vendor/` and the files of every package it updates.

1. **Fetch from**: `auto`, `dist (archives)` or `source (git, with history)`. Choose `source` so that new extensions arrive as Git clones; packages already installed keep the way they were installed.
2. **Preview (dry run)** runs `composer update --dry-run` and shows what would change. It opens the run's page.
3. To change anything, updating must be on: **Switch on updates** (it writes `AllowUpdate=enabled` into `settings/override/ezupdate.ini.append.php`).
4. A **full update**, the one that follows the kernel's new `composer.json`: on the dry run's page (with updating switched on, it offers **Run it for real** next to **Run again**), tick **A backup of files and database exists** and press **Run it for real**. **Install updates** on the Overview instead updates only the packages you tick after **Check for updates**.
5. The page follows Composer's output as it runs. Afterwards the class maps are regenerated and the INI, template and content caches cleared. A notice says *The installation is up to date!* or what changed. Switch updating off again.

[Image: Updates and packages > Overview: the counts, the Composer card with the Updating and Installing new packages switches, and the Updates box with Switch on updates, Check for updates, Fetch from, Preview (dry run) and Install updates]

Updates and packages > Overview: the counts, the Composer card with the Updating and Installing new packages switches, and the Updates box with Switch on updates, Check for updates, Fetch from, Preview (dry run) and Install updates

[Image: The page of a dry run in Updates and packages: "Dry run finished. It would change: 2 installed, 63 updated. Nothing was changed.", the Composer command and its output, and Run again]

The page of a dry run in Updates and packages: "Dry run finished. It would change: 2 installed, 63 updated. Nothing was changed.", the Composer command and its output, and Run again

The other tabs help you check before and after:

- **Installed packages** puts side by side what `composer.json` requires, what `composer.lock` resolved, what is on disk and which extensions are switched on; **Needs attention** shows what does not agree, **Git clones** which packages are clones.
- **Find packages** searches packagist.org and marks what is installed.
- **Package servers** lists the Composer servers (packagist.org can be switched off) and the `.ezpkg` package servers.
- **Funding** lists who is behind the installed packages and how to support them.

[Image: Updates and packages > Installed packages: the dates of composer.json, composer.lock and installed.json, the counts, the filters (Git clones, Needs attention and others) and the table of packages with their versions and status]

Updates and packages > Installed packages: the dates of composer.json, composer.lock and installed.json, the counts, the filters (Git clones, Needs attention and others) and the table of packages with their versions and status

[Image: Updates and packages > Find packages: the search form with Where and Type, and packages with downloads, stars and the installed version]

Updates and packages > Find packages: the search form with Where and Type, and packages with downloads, stars and the installed version

[Image: Updates and packages > Package servers: the Composer servers with packagist.org switched on, and the Exponential package servers with the built-in server]

Updates and packages > Package servers: the Composer servers with packagist.org switched on, and the Exponential package servers with the built-in server

[Image: Updates and packages > Funding: how many packages ask for funding, from how many vendors, and the funding links grouped by vendor]

Updates and packages > Funding: how many packages ask for funding, from how many vendors, and the funding links grouped by vendor

B5. Check and update the database

Setup > **Upgrade check** has two checks; both only read. **Check file consistency** compares every file with the release's checksums (after a clean Git update, every file matches). **Check database consistency** compares the tables with the schema of the new code and shows the SQL that would bring the database in line; **Download SQL** saves it.

[Image: Setup > Upgrade check: the version, the files in the manifest, the database engine, and the two checks with Check file consistency and Check database consistency]

Setup > Upgrade check: the version, the files in the manifest, the database engine, and the two checks with Check file consistency and Check database consistency

A release can need a **database update**: Upgrading names the file. Without a shell, **Setup** > **DS Editor** opens the site's database in AdminNeo after a warning page you confirm (**I have taken a full backup and accept full responsibility.**, **Proceed to the database**); there you can run SQL or import the release's file. It works on the live data, past every check of the administration. Run **Check database consistency** again afterwards.

[Image: Setup > DS Editor: the warning "Direct database access: stop and read this", the database type and tables, what the editor can do, and the confirmation before Proceed to the database]

Setup > DS Editor: the warning "Direct database access: stop and read this", the database type and tables, what the editor can do, and the confirmation before Proceed to the database

B6. Clear the caches

Setup > **Cache management**: **Clear all caches…**and confirm. On a site served by Exponential Velocity, also **Clear Velocity's response cache** (group **Pages, queries and the server**). The administration cannot restart Velocity: after an update that changes PHP code, Velocity's workers keep the old classes until `php bin/php/console exp:velocity restart` (or `exp:velocity deploy`) runs on the command line; the page says so where it matters. A site served by Apache or nginx with PHP-FPM normally picks up the new code by itself (OPcache revalidates the files).

[Image: Setup > Cache management: Clear all caches, Clear content caches, Clear template caches, Clear INI caches, Measure sizes, and the caches grouped by what they hold]

Setup > Cache management: Clear all caches, Clear content caches, Clear template caches, Clear INI caches, Measure sizes, and the caches grouped by what they hold

B7. Check the site and leave the window

Open the front page and a few typical pages (with **Let my own address still see the site** ticked you see the site, everyone else the maintenance page), and an item in the administration. Then **Setup** > **Log files**: **Error digest** lists the distinct errors and warnings of the last 24 hours, most frequent first; **Log health** shows the logs that grow fastest and the files the web server cannot write. New errors since the update are the ones to look at.

[Image: Setup > Log files > Error digest: the errors of the last 24 hours grouped by message, with count, level, when last seen, Show and Mark as known]

Setup > Log files > Error digest: the errors of the last 24 hours grouped by message, with count, level, when last seen, Show and Mark as known

[Image: Setup > Log files > Log health: disk use against the budget, the fastest growing logs, logs over their size limit, and files the web server cannot write]

Setup > Log files > Log health: disk use against the budget, the fastest growing logs, logs over their size limit, and files the web server cannot write

Only when everything passes: **Setup** > **Maintenance** > **Bring the site back online**. If something does not pass, go back first: in Git Manager, check out the commit the site ran before from the **Commits log**, run the full update in Updates and packages again, and restore the database from the backup if a database update ran.

Checklist

- [ ] Backup made (**Setup** > **Backup**)
- [ ] Maintenance on (**Setup** > **Maintenance** > **Take the site offline**)
- [ ] Kernel updated (**Git Manager**: **Fetch now**, **Checkout** of the site's branch with **Regenerate autoloads**; badge **Up to date**)
- [ ] Extensions updated (**Updates and packages**: **Preview (dry run)**, **Run it for real**; updating switched off again)
- [ ] **Upgrade check**: files and database consistent; the release's database update applied
- [ ] Caches cleared (**Cache management** > **Clear all caches…**; Velocity's response cache, and a Velocity restart from a shell)
- [ ] Front page, a few pages and the administration work
- [ ] **Log files** > **Error digest**: no new errors
- [ ] Maintenance off (**Bring the site back online**)

Mixing the two tracks

The tracks work on the same files, so they can be mixed, for example Git Manager for the kernel and Composer on the command line for the extensions. The rules:

1. **Always in the order of the steps: Git first, then Composer, then the database, then the caches.** Composer reads the `composer.json` the kernel has, so running it before the kernel moved updates to the old list.
2. **One Composer at a time.** Do not start `composer update` on the command line while a run of Updates and packages is still going (its Overview shows it under **Recent runs**), or the other way round.
3. **The same user.** Updates and packages runs Composer as the web server's user; on the command line, run Git and Composer as that same user (`sudo -u www-data composer update`, with your web server's user) so that the files do not end up owned by two users, which the web server then cannot overwrite.
4. **Restart Velocity from the command line** after either track changed PHP code on a Velocity site.

Safe combinations:

Kernel	Extensions	Notes
Git Manager (B3)	`composer update` on the command line (A4)	Run Composer only after Git Manager's checkout finished; then `ezcache.php` or `exp:velocity deploy` (A6)
`git merge` on the command line (A3)	Updates and packages (B4)	The way to take a **release tag** and still update the extensions in the administration
Either	Either	Maintenance, the upgrade check, the database update and the caches can be done in either track

Let it update itself

When you have done the update by hand a few times and trust it, a small script can do it on your schedule. Test it by hand on a staging copy first, and keep the live site on releases. An automatic update never runs database updates, and it stops instead of guessing when something needs a person.

Save this as `/usr/local/bin/exp-update` and make it executable (`chmod +x /usr/local/bin/exp-update`):

Shell

```
#!/bin/sh
# exp-update: bring an Exponential installation made of Git clones up to date.
#   exp-update release   the newest release tag of the 6.x line (v6.*)
#   exp-update main      the development line, origin/main
# Stops without changing anything when the kernel has uncommitted changes or the merge needs a ha
nd.
set -eu
SITE=${EXP_SITE:-/var/www/exponential}
MODE=${1:-release}
cd "$SITE"
if [ -n "$(git status --porcelain --untracked-files=no)" ]; then
    echo "exp-update: the kernel has uncommitted changes; commit or stash them first" >&2
    exit 1
fi
git fetch --quiet origin --tags
if [ "$MODE" = main ]; then
    TARGET=origin/main
else
```

```
    TARGET=$(git tag -l 'v6.*' --sort=version:refname | tail -1)
fi
if git merge-base --is-ancestor "$TARGET" HEAD; then
    echo "exp-update: already up to date with $TARGET"
    exit 0
fi
if ! git merge --no-edit "$TARGET"; then
    git merge --abort
    echo "exp-update: merging $TARGET needs a hand; nothing was changed" >&2
    exit 1
fi
composer update --no-interaction --no-progress
php bin/php/ezcache.php --clear-all
echo "exp-update: now at $(git describe --tags --always)"
```

Set `EXP_SITE` to your installation's folder. What it does, as tested:

Situation	What `exp-update` does
nothing new	prints `already up to date with v6.0.14` (or `origin/main`) and stops
something new	merges it, runs `composer update` and clears the caches, then prints `now at ...`
the kernel has uncommitted changes	changes nothing and exits with an error
a merge conflict with your own commits	aborts the merge, changes nothing and exits with an error
Composer refuses an extension (local changes, unpushed commits)	stops with Composer's message; run `composer update` by hand once the cause is fixed

On a site served by Velocity, replace the `ezcache.php` line with `php bin/php/console exp:velocity deploy`.

With cron

Edit the crontab of the user that owns the files (`crontab -e`) and add **one** of these lines:

```
# every hour, the development line
5 * * * *   EXP_SITE=/var/www/exponential /usr/local/bin/exp-update main >> $HOME/exp-update.log
2>&1
# every day at 04:15, the development line
15 4 * * *  EXP_SITE=/var/www/exponential /usr/local/bin/exp-update main >> $HOME/exp-update.log
2>&1
# every Monday at 04:15, the development line
15 4 * * 1  EXP_SITE=/var/www/exponential /usr/local/bin/exp-update main >> $HOME/exp-update.log
2>&1
# on the first of every month at 04:15, the development line
15 4 1 * *  EXP_SITE=/var/www/exponential /usr/local/bin/exp-update main >> $HOME/exp-update.log
2>&1
# every day at 04:15, but only when a new release has been published
15 4 * * *  EXP_SITE=/var/www/exponential /usr/local/bin/exp-update release >> $HOME/exp-update.l
og 2>&1
```

Checking daily in `release` mode updates only on the day a new release appears; on every other day it prints `already up to date`.

With a systemd timer

Two files in `/etc/systemd/system/`. `exp-update.service`:

INI

```ini
[Unit]
Description=Update the Exponential installation from GitHub
Wants=network-online.target
After=network-online.target

[Service]
Type=oneshot
User=www-data
Environment=EXP_SITE=/var/www/exponential
ExecStart=/usr/local/bin/exp-update release
```

and `exp-update.timer`:

INI

```ini
[Unit]
Description=Update the Exponential installation from GitHub, daily

[Timer]
OnCalendar=daily
RandomizedDelaySec=30min
Persistent=true

[Install]
WantedBy=timers.target
```

Use the user that owns your files in `User=`, and `main` in `ExecStart=` to follow the development line. `OnCalendar=` takes `hourly`, `daily`, `weekly` (Mondays), `monthly` (the first of the month) or a time such as `*-*-* 04:15:00`; `systemd-analyze calendar weekly` shows when it would run. Then:

Shell

```sh
systemctl daemon-reload
systemctl enable --now exp-update.timer
systemctl list-timers exp-update.timer
journalctl -u exp-update.service
```

The last two show when it runs next and what it said last time.

Where to go next

- Develop with Git: installing as clones, working on a branch, sending fixes back.
- Contribute extensions back: publish your own extensions so everyone can use them, and what the GPL requires when you distribute one.
- Upgrading: the database and settings steps of each release.
- The upgrade check and git_manager.
- The project on GitHub: se7enxweb/exponential and its releases.

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. Start with Partners: start here and How to apply.

Contribute extensions back: share your work with everyone

Why and how to give the extensions you build back to the Exponential project so everyone can use them; what the GNU GPL really requires when you distribute an extension (and what it does not require for a site you only.

Almost every Exponential site ends up with an extension of its own: a datatype, an operator, an admin page, a workflow event, a theme. Many of them would help other sites too. This page explains why it is worth sharing them, what the license of Exponential asks of you when you hand an extension to someone else, and how to publish one so that anyone can install it with one command.

This page explains the license in plain words. It is **not legal advice**: for a real decision about a commercial project, read the license text and ask someone qualified.

Why contribute an extension back

- **Everyone gets it.** An extension on GitHub and Packagist can be installed by every site with `composer require`. The more extensions exist, the more each site can do without new code.
- **Others maintain it with you.** Users report bugs, send fixes and keep it working with new PHP and Exponential releases. A private extension is maintained by you alone, forever.
- **It is how Exponential itself grew.** Most extensions in the se7enxweb organization are continuations of work that their authors once shared; the project keeps them working because their source was published.
- **It shows what you can do.** A public extension, with your name in its copyright line, is a portfolio for a developer, a designer or a partner company.

What the GPL requires, and when

Exponential is free software under the **GNU General Public License, version 2 or (at your option) any later version** (`COPYRIGHT.md` and `LICENSE.md` in the root of the installation; `composer.json` says `GPL-2.0-or-later`). Almost every extension uses the same license: of the 79 extensions installed on the project's reference site in October 2026, 75 say GPL 2 or later in `composer.json` (`GPL-2.0-or-later` or `GPL-2.0+`), three say GPL 2 (`GPL-2.0-only`, or the older identifier `GPL-2.0`) and one `GPL-3.0-or-later`. Always read the license of the extension in front of you (Reading licenses before you reuse anything).

The GPL is about **copying, changing and distributing** the code. What it asks of you depends on what you do with your extension.

Using it yourself: no obligation to publish

What you do	Does the GPL require you to give out the source?
You write or change an extension and run it on your own site	**No.** The GPL says that running the program is not restricted (GPL 2, section 0).
Your site is public on the internet and visitors use it	**No.** Visitors receive web pages, not a copy of the extension. GPL 2 and GPL 3 have no clause about use over a network (that clause is in a different license, the GNU Affero GPL, which Exponential does not use).
You use a changed extension inside your company or organization only	**No.** The Free Software Foundation's GPL FAQ says that using copies within one organization is not distribution.

So a site that is only run privately or internally, and whose code is never handed to anyone else, does not trigger the distribution clause. You may keep that extension to yourself. This page still hopes you will share it, but the license does not force you to.

Distributing it: the source goes with it

You **distribute** an extension when you give a copy to someone else: you publish it, you sell it, or you hand it to another company or person to use as their own. From that moment the GPL applies in full:

- **The source comes with it.** GPL 2, section 3, requires that whoever receives the program can get its complete source code: you give the source with it, or a written offer to provide it. An Exponential extension is PHP, templates and settings, so the files you hand over normally *are* the source; make sure nothing is left out (compressed or generated JavaScript and CSS need the files they were built from).
- **Under the same license.** A changed version, as a whole, is distributed under the GPL too (GPL 2, section 2). You keep every copyright and license notice that is already in the files.
- **No extra restrictions.** You may charge for the copy, for installation and for support, but you may not forbid the receiver to study, change or pass on the code (GPL 2, section 6).
- **To the people who receive it.** The obligation is towards the people you distribute to, not towards the whole world. Whoever receives your extension gets its source and the GPL's rights, and may pass it on in turn; you do not have to publish it on GitHub for everyone.

Agencies and freelancers. The project's own licensing FAQ takes the view that an agency or freelancer who assembles a site's code for a client under contract acts as the client's agent and is not distributing the code in the GPL's sense. The client owns the result and is the one who decides whether to distribute it. Selling or giving the same extension to other clients, or publishing it, is distribution.

An extension runs inside the Exponential kernel's PHP process and calls its classes. Code combined that closely with a GPL program is, in the Free Software Foundation's view (GPL FAQ: plug-ins), covered by the GPL when it is distributed together with it. The project's licensing FAQ says the same in its own words: modules and designs are derivative works of Exponential, so when you distribute them you do so under the GPL version 2 or later, while you are never obliged to distribute them at all. Images and other files your PHP code only sends to the browser are data, not covered by that rule, unless you made them from Exponential's own GPL files. That is why most extensions use GPL 2 or later.

The licensing FAQ also answers related questions this page does not repeat: GPL-compatible libraries in an extension, "bridge" extensions that talk to other systems, selling extensions, and what "source code" means for PHP, JavaScript, CSS and images. Read it before you decide on a license.

Choose GPL 2 or later for your extension

Write "GNU General Public License v2.0 (or any later version)" unless you have a reason to choose another GPL-compatible license. It keeps your extension combinable with the kernel and with nearly every other extension, including the few under GPL 3 (Reading licenses before you reuse anything). If your extension contains code from someone else, keep their notices and their license; if you built it for a customer, make sure you may publish it (see Publishing an extension as a partner).

How to publish an extension, step by step

The developers' guide Extensions explains how to build an extension; its section Release it gives the exact commands. Here is the whole path from a folder on your site to a package anyone can install.

1. Clean it up

- Remove everything that belongs to one site: customer names, addresses, content, passwords, API keys and other secrets. Turn site-specific behavior into a setting with a neutral default.
- Keep the usual extension layout: `settings/`, `design/`, `classes/`, `modules/`, `translations/`, `autoloads/` as needed (What an extension is).

2. Add the license file

- A `LICENSE` file with the full license text, for example the GPL 2 text from gnu.org.
- A copyright line naming you or your company, and every earlier author whose work it contains.
- A short license header in each PHP file (`@copyright` and `@license`), as the kernel's files have.

3. Write ezinfo.php and extension.xml

The about page of the administration (**Setup > About**) and the upgrade checks read these two files. The project's convention for every release:

- `ezinfo.php`: a class `<extension>Info` with a static `info()` that returns `'Name'`, `'Version'` (the release number, without a `v`: tag `v1.0.0` gives `"1.0.0"`), `'Copyright'`, `'License'` and `'Info_url'` (the repository's web address).
- `extension.xml`: a `<metadata>` block with the same `<version>`, `<license>` and `<info_url>`. Never leave a placeholder such as `//autogentag//` or `dev-master` in it.
- Both files must show the same number. Check them:

Shell

```
php -l extension/myextension/ezinfo.php
xmllint --noout extension/myextension/extension.xml
```

4. Write composer.json

Composer installs the extension into `extension/<name>/` through the installer the kernel already uses:

JSON

```
{
    "name": "yourvendor/myextension",
    "description": "What it does, in one sentence.",
    "type": "ezpublish-legacy-extension",
    "license": "GPL-2.0-or-later",
    "authors": [ { "name": "Your Name" } ],
    "require": { "se7enxweb/ezpublish-legacy-installer": "*" },
    "extra": { "ezpublish-legacy-extension-name": "myextension" }
}
```

Copy the `type`, the installer and the `extra` key from an extension that is already installed on your site (`extension/<name>/composer.json`) so that yours matches what your kernel expects. The `license` is an SPDX identifier.

5. Write the README

A good README answers, in this order: what the extension does (two sentences); the requirements (Exponential and PHP versions, other extensions); how to install it (`composer require yourvendor/myextension`, or a Git clone); how to switch it on (**Setup > Extensions**, or

`ActiveExtensions[]`); how to configure it; where to report issues; the license and the credits.

6. Put it on GitHub

Shell

```
cd extension/myextension
git init
git add .
git commit -m "Added: First public version of myextension"
git remote add origin https://github.com/<you>/myextension.git
git push -u origin main
```

Enable **Issues** on the repository, so users can report bugs where you will see them.

7. Release a version and register it on Packagist

1. Choose the number from the tags that already exist, sorted by version, and write it into `ezinfo.php`, `extension.xml` (and `composer.json` if it has a `version`). Commit that, **then** tag that commit:

   Shell

   ```
   git tag -l 'v*' --sort=version:refname | tail -5
   git commit -am "Updated: Version 1.0.0"
   git tag -a v1.0.0 -m "Version 1.0.0"
   git push origin main v1.0.0
   ```

2. Sign in at packagist.org and submit the repository's address. Let Packagist install its GitHub webhook, so that every new tag appears without you doing anything.
3. A published tag is permanent. If a release is wrong, never move or delete its tag: publish the next version with the fix and say in its notes what was wrong. Packagist reads each tag once, so a moved tag would give two different packages the same version number.

Now anyone can install it with `composer require yourvendor/myextension`.

8. Offer it to the community

- **The se7enxweb organization.** Most extensions of the Exponential ecosystem live in github.com/se7enxweb, where they are kept working with each new release. If you would like the project to carry your extension, or you have improved one of the extensions already there, open an issue or a pull request in the repository concerned, or in se7enxweb/exponential for a new extension, and say what it does and under which license. A repository can be transferred to the organization or forked into it, with your history and your copyright kept. Code that goes into the project's own repositories must be under the same license as Exponential, GPL version 2 or later (not "GPL 2 only" or "GPL 3 only"), as the licensing FAQ explains. For the kernel, contributors also give up control of their code under the CLA (below).
- **Copyright assignment (CLA).** The Exponential Foundation and 7x merge a code contribution to the **core kernel**, `se7enxweb/exponential`, sent in any form, only after you have signed the Contributor Licensing Agreement. **By signing and submitting, you assign the copyright in your contribution to the Exponential Foundation and 7x, which become its owners, and give up control over it:** The Exponential Foundation and 7x may use, change, relicense and distribute it in any way, also under other licenses and in commercial products, without asking or paying you, and you cannot withdraw it or set conditions on it. Publishing your own extension in your own repository is not affected: the CLA page says "if you have written a contribution and want to make it available on GitHub [...] you do not need to sign the CLA". The page does list "complete/incomplete extensions" among the code that needs a signed CLA "which you want 7x to include", so an extension handed over to 7x for inclusion is covered by it; the page

names no individual repositories. See Copyright assignment (CLA).

- **Exponential Projects**, projects.exponential.earth: the project forge that lists extensions with their news, downloads and source links (Exponential Projects). Give your extension a project page there that links to its repository.
- **Share Exponential**, share.exponential.earth: the community site with the forums. Announce the extension in the forums, ask for testers and answer questions (Share Exponential).

9. Keep it alive

Test it against each new Exponential release, answer issues, merge good pull requests, and say in the README when you stop maintaining it, so that someone else can take it over.

Improving someone else's extension

Contributing back also means sending your fixes to an extension you did not write, instead of keeping a changed copy. Fork it, fix it on a branch and open a pull request, exactly as Develop with Git: Contribute a change with a pull request shows. Whether the extension belongs to Exponential or to any other project, a change to behavior its author chose on purpose, which other sites rely on (a negative feature), needs a clear justification and may take several rounds of review; offer it as an option. If the original author no longer answers, fork it respectfully: keep the history, the copyright and the license, and credit them in the README (Contributing extensions back upstream).

Related pages

- Develop with Git and Update with Git: work with clones, so that sending a change back is one pull request.
- The extension-development book: from user to extension and theme developer, step by step.
- Extensions: build, configure and release an extension.
- Reading licenses before you reuse anything
- Publishing an extension as a partner
- Contribute your site package: share a whole site (design, classes, demo content) through the setup wizard.
- The full license texts: GNU GPL version 2 and the GPL FAQ.
- 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. Start with Partners: start here and How to apply.

Sign in to the administration, find the content you work on, make a first change, and know where to get help.

This page is for anyone who has just been given an account: an editor who writes for a site, or a user who wants to know their way around. In ten minutes you sign in, find your way to the content you work on, make a first change and know where to look when you are stuck.

What Exponential is

Exponential is the content management system behind your site. Everything on the site, from a page or a news article to an image, a file or a user account, is a **content item** (also called an object). Items are arranged in a tree, like folders on a computer: a folder holds articles, an article can hold images, and so on. You write and change items in the **administration**, and visitors see them on the **site**.

A change is never visible at once: you work on a **draft**, and the site shows it only when you **publish** it. The earlier version is kept, so you can always go back. Publishing, versions and the trash explains how.

1. Sign in

Step by step, with every message you may meet: Logging into the Editor for editors, Logging into the admin for administrators.

1. Open the address of the administration that you were given. On many sites it is the site's address followed by `/admin`, for example `https://www.example.com/admin`. Some sites have a separate address for editors.
2. Type your **Username** and **Password** and press **Sign in**. The eye button in the password field shows what you typed.
3. If your account uses two-step sign-in, the next page asks for a code from your phone or your e-mail. See Your account and signing in.

[Image: The sign-in page of the administration]

The sign-in page of the administration

Forgot your password? Choose **Forgot your password?** below the button. See Your account and signing in.

2. Look around

After signing in you see the **Dashboard**: a greeting, quick links, a few numbers (your drafts, your pending items, content published lately) and lists of recent content. The tabs at the top lead to the main areas:

[Image: The dashboard after signing in: a greeting with quick links, counters for content, drafts and pending items, and the main tabs in the header]

The dashboard after signing in: a greeting with quick links, counters for content, drafts and pending items, and the main tabs in the header

[Image: The header of the administration with the tabs Dashboard, Content structure, Media, Users, Store, Layouts, Setup and More outlined]

The header of the administration with the tabs Dashboard, Content structure, Media, Users, Store, Layouts, Setup and More outlined

- **Content structure**: the pages of the site, as a tree.
- **Media**: the media library, with images, files and videos.
- **Users**, **Layouts**, **Setup** and others, when your role allows them.

A tour of the administration explains every part of the screen.

3. Find the item you want to change

- Click **Content structure** and open the folders in the tree in the left column, or
- type a word into the **Search...** field at the top and press Enter. Searching has more ways.

[Image: The Search field at the right of the header, outlined]

The Search field at the right of the header, outlined

[Image: The page of the Fitness folder with its buttons Edit, Upload, Import/Export, Move, Copy and Remove outlined]

The page of the Fitness folder with its buttons Edit, Upload, Import/Export, Move, Copy and Remove outlined

Click an item's name to open it. Its page shows the item's content and the buttons for what you can do with it: **Edit**, **Move**, **Copy**, **Remove** and more.

4. Make a first change

1. On the item's page, press **Edit**.
2. Change a field, for example the title.
3. Press **Send for publishing**. The change is now on the site.

Not ready yet? Press **Store draft and exit** and come back later: the draft waits under **My drafts** on the dashboard. Changed your mind? **Discard draft** throws the draft away and leaves the published item as it was. Creating and editing content explains the form in detail.

[Image: The top of the edit form with the buttons Send for publishing, Store draft and exit and Discard draft outlined]

The top of the edit form with the buttons Send for publishing, Store draft and exit and Discard draft outlined

5. Sign out

Use the last button at the right end of the header, an arrow leaving a box; pointing at it shows **Logout** and your username. While an item is open for editing the button is grayed out: finish or leave the edit first. On a shared computer, always sign out.

[Image: The sign-out button, the last one at the right end of the header, outlined]

The sign-out button, the last one at the right end of the header, outlined

Getting help

- The help pages are under **Help** in the administration. The dashboard of the help lists every page by audience and topic, and its search finds any page in a second.
- Many pages of the administration explain themselves in a sentence or two under the title; buttons explain themselves when you point at them.
- If something looks broken, see When something goes wrong.

[Image: The help center with its search field and buttons that filter the help pages by audience]

The help center with its search field and buttons that filter the help pages by audience

Grow from user to developer

Everything you use here is free software, and every page of this help is free documentation. When you are ready to go further than editing content, to change templates, try a fix or build a site of your own, start with Develop with Git: it shows the recommended way to install Exponential as Git clones with their full history, step by step, and how to send your improvements back to the project.

A Demo mode banner? On a public demo the administration shows a yellow **Demo mode** banner: the administrator password is published, and administrator passwords cannot be changed or reset. See Demo mode.

Next steps

- A tour of the administration
- Creating and editing content
- Keyboard shortcuts
- Develop with Git

See also: Non-technical people.

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. Start with Partners: start here and How to apply.

Keyboard shortcuts

Every key the administration and the help pages answer to: search, menus, the content tree, tabs, sub-item order, column width and the editors.

The administration can be used with the keyboard alone. Besides the usual keys of every web page (**Tab** and **Shift+Tab** to move between links, buttons and fields, **Enter** to follow a link or press a button, **Space** to tick a checkbox), it answers to the keys below. They work in the **Admin 4** design; the other designs have most of them.

On a Mac, **Alt** is the **Option** key.

Everywhere in the administration

Key	What it does
`/`	Jumps to the **Search...** field in the header (not while you type in another field).
`Ctrl` + `/` (`Cmd` + `/` on a Mac)	Jumps to the **Search...** field in the header, also from inside another field.
`?` / `?`, `Enter` (in the search field)	Move through the suggestions under the field and open the chosen one (Searching).
`Esc`	Closes the open menu, list or pop-up, and returns the focus to its button.
`Tab` (first press on a page)	Shows **Skip to main content**; press `Enter` to jump past the header and the left column.

Menus and lists of choices

This covers the tab menus of the header, the user menu, the **More** list, the search scope, and the menu of an item.

Key	What it does
? on the button	Opens the menu and moves to its first entry.
? / ?	Next / previous entry.
`Home` / `End`	First / last entry.
`Enter` or `Space`	Chooses the entry.
`Esc`	Closes the menu.

The content tree

Move into the tree with `Tab`; then:

Key	What it does
? / ?	Next / previous item.
`->`	Opens a closed folder; on an open one, moves to its first item.
`<-`	Closes an open folder; on a closed one or an item, moves to its parent.
`Home` / `End`	First / last item of the tree.
Letters	Jump to the next item whose name starts with what you type.
`Enter`	Opens the item.
`Shift+F10` or the menu key	Opens the item's menu (edit, copy, move, create here ...). The same works on a row of a **Sub items** list.

Tabs of an item's page

With the focus on a tab (**View**, **Details**, **Translations** ...):

[Image: The tabs of an item's page: View, Details, Translations, Locations, Relations, Audit, Intelligence, Tags, Authors, Layouts and Ordering]

The tabs of an item's page: View, Details, Translations, Locations, Relations, Audit, Intelligence, Tags, Authors, Layouts and Ordering

Key	What it does
`->` / `<-`	Opens the next / previous tab.
`Home` / `End`	Opens the first / last tab.

Lists of sub items

When a **Sub items** list is sorted by priority, the grip at the start of each row changes the order:

Key on the grip	What it does
`Space` or `Enter`	Picks the row up.
`?` / `?`	Moves the picked-up row.
`Home` / `End`	Moves it to the top / bottom of the page.
`Space` or `Enter`	Drops it there; the new order is saved.
`Esc`	Puts it back where it was.
`Alt+?` / `Alt+?`	Moves the row by one place at once, without picking it up.

The width of the left column

Focus the handle with three dots on the edge of the left column:

[Image: The three buttons at the bottom of the left column that set its width]

The three buttons at the bottom of the left column that set its width

Key	What it does
`->` or `?`	Wider (in a right-to-left language: `<-` or `?`).
`<-` or `?`	Narrower.
`Shift` with an arrow	A bigger step.
`Home` / `End`	Narrowest / widest.

The edit form

Key	What it does
`Enter` in a one-line field	Same as **Send for publishing**.
Keys of the rich text editor	The usual ones: `Ctrl+B` bold, `Ctrl+I` italic, `Ctrl+Z` undo and more (`Cmd` on a Mac). The **Help** button of TinyMCE lists them all.

Lists you choose from

In a list box, such as **Interface language**, type the first letters of an entry to jump to it; the arrow keys, `Home`, `End`, `Page Up` and `Page Down` move through it. Nothing changes until you press **Apply**.

In the help pages

Key	What it does
`/`	Jumps to the help's search field.
`?` / `?`	Moves through the search results as you type.
`Enter`	Opens the chosen result.
`Esc`	Clears the results; a second `Esc` leaves the field.
`j` / `k`	Next / previous heading of the page you read.

Key	What it does
] / [	Next / previous help page.

[Image: A page of the help: the contents of the help on the left, the page in the middle and its headings on the right]

A page of the help: the contents of the help on the left, the page in the middle and its headings on the right

These keys are ignored while you type in a field.

Learn more

- A tour of the administration
- Searching

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. Start with Partners: start here and How to apply.

Languages

Translate an item into another language, see which translations exist, and choose the language the administration itself is shown in.

There are two different kinds of language in Exponential, and this page explains both:

- **Content languages**: the languages your items are written in. An article can exist in English and in German; each is a **translation** of the same item.
- **The interface language**: the language of the administration's menus, buttons and messages. You choose it for yourself; it changes nothing in the content.

See which translations an item has

On an item's page, the line under the buttons shows a small badge per language of the site. A filled badge, such as **EN**, is a translation that exists; a dashed badge, such as **DE**, is one that does not exist yet. Point at a badge to read it, for example *English (American) – main language* or *Translate into German*.

[Image: The Translations tab of an article: the existing translations with their locale and main language, and the missing ones]

The Translations tab of an article: the existing translations with their locale and main language, and the missing ones

The **Translations** tab of the item lists its translations in full.

Translate an item

1. Open the item.
2. Click the dashed badge of the language, or choose **New translation** in the language list next to **Edit** and press **Edit**.
3. If the page asks, choose the language under **New translation** and, under **Translate based on**, the existing translation to start from (its text is copied into the form), or **None** for empty fields.

4. Translate the fields and press **Send for publishing**.

[Image: The language list next to Edit, where you choose the language to edit or translate into, outlined]

The language list next to Edit, where you choose the language to edit or translate into, outlined

To change an existing translation, choose that language in the list next to **Edit**, then press **Edit**. The edit form names the language it is editing next to the title.

Each translation is published on its own: publishing the German text does not change the English one.

The main language

One translation of each item is its **main language**, marked as such on its badge. It is the language the item was created in, and the one shown where a translation is missing, if the item is set to be always available.

Remove a translation

The **Translations** tab lists the translations with a checkbox each; removing one needs the right permission and cannot remove the main language. Ask an administrator if you are not sure.

The language of the administration

The right column has the box **Interface** with the list **Interface language**.

[Image: The Interface box in the right column with the Interface language list and the Apply button]

The Interface box in the right column with the Interface language list and the Apply button

[Image: Setup, Interface languages: every locale of the administration, what it shows and how complete its translation is]

Setup, Interface languages: every locale of the administration, what it shows and how complete its translation is

1. Open the list. It shows the most used languages first, in groups, then every language from A to Z. Each entry shows the language's own name, its name in the current interface language, and how complete its translation is, for example *Deutsch (Deutschland) —German (Germany) (100%)*. With the list focused, typing the first letters of a name jumps to it.
2. Choose a language and press **Apply**. The page loads again in that language.

What changes: the menus, buttons and messages, and the format of dates, times and numbers. A language marked *no translation yet* keeps English texts but uses its own date and number formats.

What does not change: the content (you still edit the translation you open), other users, and other parts of the installation. The choice lasts until you sign out. Choosing the site's own language again returns to it.

The box is not shown while you edit an item, and not to roles without the permission. When allowed, the same box also offers the **Administration design**.

Learn more

- Content languages, the full guide, including adding a language to the site.
- Interface language and the UI switch
- Reading the help in your language: the help follows the interface language.

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. Start with Partners: start here and How to apply.

Searching

Find content in the administration with the quick search, the search scope and the filters of the search page, open an item by its number or address, and search the site.

This page is for editors who need to find an item quickly, and for visitors who search the site. The quick search, its suggestions and the search page work alike in every administration design (Admin 4, Admin 3, AdminUI and Classic).

The quick search

The **Search...** field in the header of the administration searches all content you may read.

[Image: The results of a quick search for running: each item with its place in the tree and its type]

The results of a quick search for running: each item with its place in the tree and its type

1. Click the field, or press `/` anywhere on the page (not while typing in another field).
2. Type one or more words and press Enter.

The results page lists each match with its name, its path in the tree and its type. Click a name to open the item. Tick results and press **Edit selected** to edit several of them in one form.

[Image: The search page with one match]

The search page with one match

Suggestions while typing

From the third character on, a list opens under the field while you type. It shows, in groups:

- **Content**: items whose text matches, each with its type and where it is in the tree.
- **Document pages**: the pages of a PDF file that hold the words, with a link that opens the document at that page (only on sites that index PDF pages; elsewhere a document is found as a whole, under **Content**).
- **Users**: user accounts by login, e-mail address or name.
- **Content classes**: classes by name or identifier, if you may open the class list.
- **Administration**: pages of the administration's menus, such as *Setup › Cache management*.

You only see what you may open. The words you typed are highlighted. **View all results** at the end opens the full results page, with the number of matches.

- `?` / `?` move through the list, `Enter` opens the chosen entry (or searches when none is chosen).
- `Esc` or a click outside closes the list.
- `Ctrl` + `/` (`Cmd` + `/` on a Mac) jumps to the field from anywhere, also from inside another field.

For administrators: the groups, their order, how many entries each shows and the number of characters before the list opens are set in `admininterface.ini` `[SearchSuggestSettings]`; `Enabled=false` switches the suggestions off. The list is filled by the view `content/suggest`, which uses the search engine of the site (eZ Find when it is installed and active, the built-in search otherwise). It needs the policy `content/read`: a role without it gets no suggestions, not even the administration's pages. A word is found while it is still being typed ("Oste" finds "Ostern") only with `site.ini` `[SearchSettings]`

`EnableWildcard=true` (or `Wildcard=enabled` in `[SearchSuggestSettings]`); otherwise the list matches whole words.

Narrow the search

The funnel button in the search field opens **Search scope**:

[Image: The search page with the In list, which narrows the search to one section, outlined]

The search page with the In list, which narrows the search to one section, outlined

- **Section**: search only in one section of the site, such as *Media* or *Users*.
- **Where**: search everywhere, or only below the place you are looking at (**Current location**).
- **Advanced search** opens the search page with its filters.

The search page and its filters

The search page (**Search** in the header, or `content/search`) has the search field on top, the **Filters** beside the results (above them on a narrow screen, folded under **Show filters**) and the results in a table. It also takes in the former advanced search: an old link to `content/advancedsearch` opens this page. The address of a search keeps every filter (`content/search/(text)/easter/(class)/2,16`), so you can bookmark it or send it to a colleague.

- **Words**: **All words** (the default), **Any word** or **Exact phrase**;
- **Content type**: one or more types, grouped by class group; the less used ones are under **Show more**. With exactly one type chosen, **Class attribute** searches only one field of that type;
- **Section**: one or more sections;
- **Modified** and **Published**: any time, the last day, 7 days, month or year, or a **Custom range** with a from and a to date;
- **Owner**: a user's ID, login or e-mail address;
- **Language**: only the translations in one language;
- **Location**: all content, or the place the search started from (the header's **Current location**);
- **Results per page**.

Choose filters and press **Apply** (it is active once something changed); **Reset** keeps the words and removes every filter. The page works without JavaScript as well: then every field is sent with **Apply**.

Above the results:

- the header **Results for “words” (count)**;
- every active filter as a chip: click it to search again without that filter;
- **Sort by**: relevance, name A–Z or Z–A, modified or published, newest or oldest first. The pages keep the sort order and all filters;
- **Did you mean** with a spelling suggestion, when the search engine offers one.

Each result shows its type, its name with its place in the tree and the words found, when it was modified, its content type, its translations (each a link to the item in that language) and **Edit**. Tick results and press **Edit selected** to edit several of them in one form. A PDF lists the pages that hold the words, when the deep search for documents is installed.

What the filters can do depends on the search engine. With eZ Find all of them are there, with counts next to

the content types, sections and languages. With the built-in search there is no **Any word**, no **Modified**, no **Owner**, no **Language**, no counts and no sorting by relevance, and a search needs words.

Open an item by its number or address

When you know an item's number or address, you do not need to search. The **Content finder** below the tree in the left column takes:

[Image: The Content finder in the left column, which opens an item by its id, path or address]

The Content finder in the left column, which opens an item by its id, path or address

- the **node** number (shown as *Node* on the item's page),
- a path such as `media/images`,
- or the full address of the page on the site.

Type it and press **Find**.

Search within lists

- The **Sub items** list of a folder shows the items below it. Sort a column by clicking its heading.
- **My bookmarks** has its own **Find a bookmark** field.
- In the **Trash**, filters narrow the list by who removed the item, its type and the date.

Searching the site

Visitors search the public site with its own search field, usually in the header. The results show the title, date, type and the first lines of each match, a few per page with links to the next pages. Only published, visible items that the visitor may read are found.

[Image: The search of the public site with the results for healthy]

The search of the public site with the results for healthy

Suggestions on the site

From the third character on, the site's search field opens a list of suggestions while the visitor types: the pages that match, each with its name, its type and a small picture. Up and Down move through them, Enter opens the one chosen (or searches when none is), Escape closes the list, and Ctrl+/ (Cmd+/ on a Mac) reaches the field from anywhere on the page. The last entry, "View all results", opens the search page.

Only content is suggested, and only what the visitor may read: an anonymous visitor sees what is public. User accounts, content classes, the administration's pages and settings are never suggested. On a site with its own part of the tree (such as /bold) only the pages of that part are found.

Searching on the site

The site's search page works as the administration's: the words and the filters travel in the address (`/content/search/(text)/easter/(class)/16`), so a search can be bookmarked and shared, and a filter bar beside the results narrows the search down by content type and changes the sort order. Each active filter is shown as a chip above the results; a click on it removes that filter. When eZ Find is the search engine, "Did you mean" offers the same search with corrected words. An old link of the form `/content/search?SearchText=...` opens the empty search page.

Settings for the site search (for administrators)

The suggestions are on for every siteaccess. They are set in `site.ini [SiteSearchSuggestSettings]`, per siteaccess in its `site.ini.append.php`:

Setting	Default	Meaning
`Enabled`	`true`	`false`: a plain search field; `content/sitesuggest` answers 404
`MinLength`	`3`	the fewest characters typed before suggestions are asked for
`Limit`	`6`	the most suggestions (0 to 20)
`ExcludeClasses[]`	`user`, `user_group`	classes never suggested (a class with a user account never is)
`IncludeClasses[]`	(empty: every class)	only these classes are suggested and offered in the search page's content type filter; for a site whose tree holds other sites' classes (a multisite package)
`ThumbnailAlias`	`small`	the image alias of the small picture; empty: the class icon
`PublicCacheTTL`	`60`	seconds an anonymous answer is kept (on the server and in shared caches)
`RateLimit`, `RateWindow`	`60`, `60`	requests of one address per window; more are answered 429

The visitor needs the policy `content/read`, as for the search itself. A signed-in user's suggestions are never cached.

For a design: put `{include uri='design:parts/front_search_suggest.tpl'}` inside the header's search form (with `method="post"` and a field named `SearchText`), and let the design's `content/search.tpl` hold `{include uri='design:content/front_search_page.tpl'}`. Both are drawn by `stylesheets/front_search.css` with `--exps-*` tokens that a design sets to its own colors, in light and dark.

Tips

- Search finds whole words. Try a shorter or a different word if nothing is found.
- New and changed items are usually found within moments; on some sites the search index is updated in the background and takes a few minutes.
- If you cannot find an item you know exists, it may be hidden from you by your role, it may be in the trash, or it may be only a draft. Ask an administrator.

The help pages

The help has a search of its own: on the help dashboard and on every help page, type in the help's search field (or press `/`) and the matching pages appear as you type. Filters narrow them to an audience or a topic. See Keyboard shortcuts.

[Image: The help center's search results for upload, with a snippet of each page]

The help center's search results for upload, with a snippet of each page

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. Start with Partners: start here and How to apply.

When something goes wrong

What to do when something goes wrong: understand the error page, tell your administrator, and file a bug in the right issue tracker with what is needed to fix it.

This page is for everyone who meets an error: an editor in the administration, a visitor of the site, or an administrator who passes a problem on. It explains what the common error pages mean, who to tell, and how to describe a problem so that it gets fixed.

First: is it an error?

Some pages look like errors but only say that something is not allowed or not there:

[Image: The page a visitor gets for an address that does not exist: a friendly not found page, not an error of the site]

The page a visitor gets for an address that does not exist: a friendly not found page, not an error of the site

What the page says	What it means	What to do
The requested page could not be displayed. (1), *...because of security issues*, *Your account does not have the proper privileges*	Your role may not open this page or do this action.	Ask your administrator for access, if you need it.
The requested page could not be displayed. (20), *The requested address or module could not be found*	The address is wrong, or that part is not installed on this site.	Check the address; go back and use the menus.
Possible edit conflict	Someone (or you) already has a draft of the item.	See Creating and editing content.
A button is grayed out	You may not use it here, or now (for example while editing).	Point at it to read why.
The page is very slow, or the site is in maintenance	The site may be busy, or being updated.	Wait a moment and try again.

When something really went wrong

An error of the site shows a page such as **Something went wrong on our side**, with a **reference** at the bottom, for example `ERR-0123456789`. The error has already been logged; the reference lets an administrator find exactly that moment in the logs.

[Image: Setup, Log files: the digest of recent errors and warnings across the log files]

Setup, Log files: the digest of recent errors and warnings across the log files

1. **Note the reference**, the time, and the address of the page (copy it from the browser's address bar).
2. **Write down what you did**, step by step, right before the error: which page, which button, what you typed.
3. Try once more. If it works now, mention that it happened once.
4. **Tell your administrator** (or the site's contact) and pass on the reference, the time, the address and the steps. Administrators find the request from the reference, see Finding an error by its reference.

A screenshot helps, as long as it shows no passwords or other people's personal data.

Filing a bug in Exponential

> **Before you file it:** not everything that looks like a bug is one. A behavior can be deliberate or open to interpretation, what engineers call a negative feature. Changing it can affect every site that relies on it, so it needs a strong justification and often several rounds of review. See Negative features.

When your administrator (or you, as one) finds that the problem is in the software itself and not in the site's own setup, file it publicly so it can be fixed for everyone. The **Report an error** page of the help (`exphelp/report` in the administration) guides you through it, fills in a summary of the installation for you and opens a new issue with a prepared title and text.

[Image: The Report an error page of the help center: which tracker to use and a prefilled issue with a safe summary of the installation]

The Report an error page of the help center: which tracker to use and a prefilled issue with a safe summary of the installation

Choose the right tracker:

Tracker	For
Exponential (the software)	Errors in the kernel, the administration, the bundled extensions and the documentation.
Exponential Velocity	Errors of the Velocity application server: starting, workers, its response cache, HTTPS.
The extension's own repository	Errors in one extension that has its own repository; the **Report an error** page lists them.
Exponential project and community	The project, the website, the ecosystem, and everything that is not an error in a piece of software.

You need a free GitHub account to open an issue. Search the tracker first: if the problem is already known, add what you know to that issue instead of opening a new one.

A good bug description

- **A short title** that names the problem: *Upload page: ZIP file with a folder fails with "temp_dir"*.
- **What you did**, step by step, so someone else can make it happen again.
- **What you expected**, and **what happened instead**, with the exact message.
- **The environment**: the Exponential version, the PHP version, the web server (Exponential Velocity, Apache or nginx with PHP-FPM, FrankenPHP), the database and the administration design. The **Report an error** page and **Setup** > **System information** (**Copy report as text**) give you this summary; both leave out passwords, keys and server paths. On the command line, `./console exp:info --plain --no-logo` (or `--json`) gives the same and more (every extension with its version, the engine, the caches): it never prints a password, key, token, database user or full database address, so you can paste it as it is (The system card).
- **Log lines** around the time, if an administrator can get them; remove anything private first.

Never put passwords, keys, session cookies, database addresses, or the personal data of customers or users into a public issue.

Even better: send the fix

If you found the cause yourself, you can send the fix along with the report. With the code installed as Git clones, a fix is a branch in your fork and a pull request away, and it reaches every site in the next release. Develop with Git shows how, step by step. Mention the issue in the pull request (for example `Fixes #123`) so both are linked.

Security problems

If you find a security problem, such as a way to see or change what you should not, do **not** file it in a public tracker. Send it privately to **security@se7enx.com**, or use GitHub's private vulnerability reporting of the repository. Say which version you run, what an attacker could do, and how to reproduce it. The full policy is in the file `SECURITY.md` at the root of the installation.

"This installation is a public demo". That message, or a **Demo mode** banner, means the site runs in demo mode on purpose: administrator passwords cannot be changed or reset there. If it shows on a real site, tell the people who run the server at once. Demo mode FAQ, Demo mode problems.

Asking for help

For questions that are not bugs ("how do I ...?"), the community is the right place: the Exponential community forums, the Telegram community chat, the Discord server and the Slack workspace, all free. Before you ask, look in the Exponential documentation, on eZpedia and in the articles on Share Exponential. The other help pages may already have the answer: see Help for editors and users.

[Image: Report an error in the dark mode: where each kind of problem goes, and the safe summary of the installation to copy]

> *Report an error in the dark mode: where each kind of problem goes, and the safe summary of the installation to copy*

Free help depends on the time people give, so nobody promises an answer time. When you cannot wait, the project's support page recommends paid support from 7x, the company that maintains Exponential (se7enx.com/services). Both kinds of support are explained on Getting support.

Related pages

- When a migration fails

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. Start with Partners: start here and How to apply.

Your account and signing in

Sign in and out, register, reset a forgotten password, change your password and profile, turn on two-step sign-in, and manage e-mail and notifications.

This page is for everyone with an account, on the site or in the administration. It explains how to sign in and out, what to do when you forgot your password, and how to keep your account safe with two-step sign-in.

Sign in

Every detail of the sign-in pages: Logging into the Editor and Logging into the admin.

1. Choose **Login** on the site (often in the user menu of the header), or open the administration's address.
2. Type your **Username** and **Password** and press **Sign in** (**Login** on the site).
3. If two-step sign-in is on for your account, enter the code the next page asks for. See Two-step sign-in.

Your browser must accept cookies to sign in.

Your first sign-in with a temporary password

When an administrator created your account, you received the address and your username, and the temporary password by a second way (a phone call, for example). At your first sign-in the site opens **Change password** with the note *Choose a new password before you continue*:

1. Under **Current password**, enter the temporary password.
2. Enter your new password under **New password** and again under **Confirm new password**, and press **Change password**.
3. **Verify it's you** asks for the temporary password once more: enter it and press **Confirm**.
4. *Your password was changed.* Press **Continue**.

From now on only you know your password. Every step with screenshots: Setting up accounts for editors and administrators.

Sign out

In the administration, use the last button of the header, an arrow leaving a box (**Logout**). On the site, choose **Logout** in the user menu or on your profile page. Always sign out on a shared computer.

Register an account

Where a site lets visitors register, the sign-in page has **Sign Up** (or **Register new account**). Fill in your name, a username, a password and your e-mail address. Under **E-mail from us (optional)** tick only the e-mail you want; you can change that later. Some sites send a link to confirm your address before the account can be used.

If you forgot your password

1. On the sign-in page choose **Forgot your password?**.
2. Enter the e-mail address of your account and press the button (**Send me a link** on the site).
3. Follow the instructions in the e-mail you receive. Check your spam folder if nothing arrives within a few minutes.

If you no longer have access to that e-mail address, ask an administrator to reset your password.

My account

In the administration, **Edit profile** in the dashboard's left column, or your name in the user menu, leads to **My account**: your account at a glance (login, e-mail, groups, roles, last sign-in) and every page to change it.

[Image: My account in the administration]

My account in the administration

[Image: The Current user box in the right column with Change information and, outlined, Change password]

The Current user box in the right column with Change information and, outlined, Change password

Card	What you can do
Profile	Change your name, e-mail address, image and the other fields (**Edit profile**).
Password	**Change password**.
Account settings	Only for roles that manage accounts: enable or disable the account, reset failed sign-ins.
Two-step sign-in	Turn the second step on or off. See below.
API keys	Only for roles that may use them: personal keys for your own scripts and tools.
Bookmarks	Your bookmarks, in folders.
Notifications	The content you follow, and how often you hear about changes.
E-mail preferences	Which e-mail you get, a download of your e-mail data, or stop all optional e-mail.

On the site, your **User profile** page has the same links: e-mail preferences, two-step sign-in, your drafts, your pending items and your notification settings.

Change your password

1. Open **Change password** (user menu, dashboard, or **My account**).
2. Enter your **Current password**, then the **New password** twice. The page shows what the new password needs, for example *At least 10 characters*, and checks it as you type. **Generate a strong password** makes one for you; **Show** reveals what you typed.
3. Press **Change password**.

[Image: The Change password form with the old password, the new password and its confirmation]

The Change password form with the old password, the new password and its confirmation

Use a password you do not use anywhere else. A password manager remembers it for you.

Two-step sign-in

With two-step sign-in, your password alone no longer opens your account: after the password, the sign-in asks for a 6-digit code from an app on your phone or from an e-mail. Someone who learns your password still cannot get in.

Turn it on

1. Open **My account** and press **Turn on** on the **Two-step sign-in** card (on the site: **Two-step sign-in** on your profile page).
2. Under **Second step at sign-in** choose how you confirm it is you:
   - **Authenticator app (TOTP)**: a code from an app such as Google Authenticator, Microsoft Authenticator, Authy or 1Password. Works without a network. Recommended.
   - **E-mail one-time code**: a code is sent to your e-mail address at each sign-in.
   - **Off**: the password alone signs in.
3. For the app: in the app, add an account and scan the QR code, or type the key shown below it. Then type the code the app shows into **Enter the code the app shows**. Until that code is right, nothing changes.
4. Press **Save settings**. The new setting applies from your next sign-in.

[Image: Choosing the second step]

Choosing the second step

Signing in with it

After your password, enter the code the app shows for the site (or the one in the e-mail; it is also in the subject). Each code works once and an app's code changes every 30 seconds. After five wrong codes the sign-in starts again with the password.

Change it or turn it off

Switching to e-mail codes, turning it off, or resetting it asks for a current code from the app first, so that nobody at a computer you left signed in can weaken your account.

Lost your phone?

Ask an administrator. They can switch your account to e-mail codes or turn the second step off; you then sign in and set up the app again on your new phone.

Some sites require two-step sign-in for everyone. You are then asked to set it up right after your password at your first sign-in.

E-mail and notifications

- **My e-mail preferences** chooses which e-mail you get from the site: content notifications, approvals, newsletters and others. Every optional e-mail can be stopped there.
- **My notification settings** lists the items you follow. You get an e-mail when something is published below them. To follow an item, choose **Add to my notifications** in its menu, or **Add items** on the settings page. A **digest** collects the messages into one e-mail per day, week or month instead of one per change.

[Image: My e-mail preferences: the categories of mail and whether you receive each]

My e-mail preferences: the categories of mail and whether you receive each

[Image: My notification settings: the parts of the site you follow and how often you get mail about them]

My notification settings: the parts of the site you follow and how often you get mail about them

On a public demo (a **Demo mode** banner in the administration) the passwords of administrator accounts cannot be changed or reset; other accounts work as usual. Demo mode.

From the console

Administrators can help with your account from the server as well: reset a forgotten password, unlock an account after too many wrong passwords, or reset two-step sign-in after a lost phone. See Sessions and sign-in security.

Learn more

- Two-step sign-in, for administrators who set it up.
- Setting up accounts for editors and administrators, for administrators
- What the Editor role allows
- Getting started

See also: Non-technical people.

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. Start with Partners: start here and How to apply.

Reading the help in your language

The help is shown in the language of the administration.

This page is for everyone who reads the help of the administration and would rather read it in their own language.

The help follows the language of the administration

The help is shown in the language the administration is shown in. You do not choose a language for the help separately: switch the interface language and the help switches with it.

1. Open the interface language switch in the top bar of the administration (see Languages if your administration does not show it).
2. Choose a language, for example German.
3. Open the help, or reload the help page you are on.

The page you are on stays open: every help page has the same address in every language, so you see the same page, now in the language you chose.

Switching from the help itself

The start page of the help and every help page show the language you are reading in, next to the search: "Reading in: English / Deutsch". The language you read in is highlighted; choose another one to switch. This is the same switch as the interface language switch in the top bar, so the whole administration changes to that language, and you stay on the page you were reading.

Only the languages that have help pages are offered. Next to the switch you see how much of the help is translated, for example "German: 120 of 7,265 pages".

Choosing a language without the switch

On a public website, such as a documentation site that visitors read without signing in, and for an account that may not change the interface language, there is no interface language switch. The help then offers each language that has pages as a link: "Reading in: English / Deutsch". Choose one and the same page opens in that language.

The language you chose stays while you read the help: the start page, the contents, the page tree, the search, the topics and every page you open from there are shown in it, and their links keep it. Behind the scenes the help adds `/(lang)/<language>` to its addresses, for example `/exphelp/dashboard/(lang)/ger-de`, so you can also bookmark or share a page in your language. Choose English again, or open the help without `/(lang)/`, to go back.

A page that is not translated yet is shown in English with the note described below, and the search looks through the pages of the language you chose. The labels, buttons and menus around the help stay in the

language of the website.

The English original

A translated page has a link to its English original ("Read the English original") above the text. The original opens at the same address followed by `/(lang)/eng-us`; it is shown in English for that one visit, with a link back to the page in your language. Your interface language does not change. (The older key `/(lang)/en-us` still works; a `?lang=` in the address is not read.)

A page that is not translated yet

The documentation is written in English first and translated page by page. When a page has no translation in your language yet, the help shows the English page and says so above it, for example:

> This page is not translated into German yet. It is shown in English. Open the English original

Everything else works the same: the search, the navigation tree and the links. The changelogs are always in English; they are not translated.

Searching in your language

The search looks through the pages in your language, and through the English pages that are not translated yet. It does not show the English original of a page that is translated, so every page appears once, in the best language there is for you.

Which languages there are

English is the original. German is being translated. Your administrator can add a language in the settings (`exphelp.ini`, `[DocLanguages]`); developers and translators find how in Translated documentation.

In the files of the documentation, each language is a directory named after the code Exponential uses for the language, in small letters: `doc/eng-us/` is the English original (`eng-US`), `doc/ger-de/` the German translation (`ger-DE`). A new language gets its directory the same way, for example `doc/fre-fr/` for French (`fre-FR`). You do not need these names to read the help: every help address is the same in every language, and links and bookmarks from before the directories were renamed still open the right page.

Related

- Languages: content languages and the interface language
- Searching
- Getting started

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. Start with Partners: start here and How to apply.

Copyright and copyleft: who owns Exponential and what the GPL lets you do

Exponential's one copyright notice, "Exponential Copyright ©1998-2026 Exponential Foundation and 7x", and its file header form; the historical shift of 2026-10-29 from 7x alone to the Exponential Foundation and 7x.

Exponential has **one copyright notice**. Wherever a person sees it (the footer of a page, the sign-in page, the about page, the help, the console) it reads, word for word:

> **Exponential Copyright ©1998-2026 Exponential Foundation and 7x**

In the source files, in the metadata of extensions and in the license files, the same notice is written in the form programmers have always used for file headers:

> **Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.**

The software is free software under the **GNU General Public License, version 2 or (at your option) any later version** (GPL-2.0-or-later). The copyright says *who* owns the work; the license says *what everybody else may do* with it. This page explains both, in plain words, and the rules every file, extension and page of the project follows.

This page is a summary, not legal advice. The license text in the file `LICENSE` and the notices in `COPYRIGHT.md` are what count.

Contents

1. The notice in one minute
2. A historical shift: 29 October 2026
3. Why the Exponential Foundation is named first
4. The two forms and where each is used
5. What "copyleft" means
6. What the GPL means for you
7. The support window and why the notice matters for it
8. Contributors and the CLA
9. Notices that stay: eZ Systems AS and third-party software
10. Writing the notice in new files
11. Writing the notice in an extension
12. Showing the notice in a design or theme
13. Translations: the notice is never translated
14. How the project checks the notice
15. The documentation's own license
16. Questions people ask
17. Where to read more

The notice in one minute

Question	Answer
Who holds the copyright of Exponential?	The **Exponential Foundation and 7x**, together.
Since when does the notice read this way?	Since **29 October 2026** (the 6.0.15 line).
Which years does it name?	**1998 to 2026**. The end year moves with the releases.
What does a user see?	"Exponential Copyright ©1998-2026 Exponential Foundation and 7x"
What does a file header say?	"Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved."
Under which license is the software?	GNU General Public License, version 2 or any later version (GPL-2.0-or-later).
Under which license is this documentation?	GNU Free Documentation License, version 1.3 or any later version.
Is the original developer still named?	Yes: "Copyright (C) 1999 - 2014 eZ Systems AS", the developer of eZ Publish, the discontinued product Exponential grew from.
Is the notice translated?	No. The words around it are; the notice is the same in every language.
Do I have to change my own extension?	Only if it says it is copyright of 7x or of the Foundation. Your own work carries your own notice.

A historical shift: 29 October 2026

On **29 October 2026** the copyright of Exponential moved from **7x alone** to the **Exponential Foundation and 7x together**, and the project replaced every notice it had with one. It is a milestone in the history of the software, and it is recorded here, in `COPYRIGHT.md` and in the 6.0.15 changelog.

How the notice read before

Over the years the notice of the project's own work had grown into several spellings, each from a different time and a different file:

Where	How it read before 2026-10-29
Footer of the administration	"Exponential Copyright ©1998-2026 7x and others."
Sign-in page of the administration	"Exponential Copyright ©1998-2026 7x and others."
Standard design footer	"Copyright ©1998-2026 7x (except where otherwise noted). All rights reserved."
Setup wizard	"Exponential copyright ©1998-2026 7x & Exponential Foundation"
Fatal error page	"Copyright ©1998-2026 7x and others."
`./console --version`	"Copyright (C) 1998-2026 7x / Exponential Foundation."
/ezinfo/copyright	"Copyright (C) 1998-2026 7x / Exponential Foundation. All rights reserved."
`LICENSE`	"Copyright (C) 1998 - 2026 7x / Exponential Foundation. All rights reserved."
`COPYRIGHT.md`	"Copyright ©1998 - 2025, 7x / Exponential Foundation." and "Copyright (C) 1998 - 2026 7x & Exponential Foundation."

Where	How it read before 2026-10-29
Most PHP file headers (about 2,850)	"@copyright Copyright (C) 1998 - 2026 7x & Exponential Foundation. All rights reserved."
Many newer PHP file headers (about 590)	"@copyright Copyright (C) 1998 - 2026 7x and the Exponential Foundation. All rights reserved."
Other headers	"Copyright (C) 7x / Exponential Foundation.", "Copyright (C) 7x.", "Copyright (C) 1998 - 2026 7x.", "Copyright (C) 1998 - 2026 7x and others."
Translation catalogues	"Copyright ©1999-2014 7x AS and others." (a leftover of the 2014 rename of the original notice)

Before the 2026-10-02 change "Every copyright notice names 1998 - 2026 7x & Exponential Foundation first", many files named only eZ Systems AS. That change put the project's own line above the line of eZ Systems in some 3,000 files. The change of 2026-10-29 finishes the work: one holder statement, one order, two forms, everywhere.

What changed on 29 October 2026

- Every footer and page a user sees shows the visible notice (see the list).
- About 3,600 file headers use the header form: the line was rewritten by a script, and nothing else in the files changed.
- `LICENSE` and `COPYRIGHT.md` name the new holders. **The text of the GNU General Public License was not touched**: only the holder lines above it changed.
- The metadata of the extensions shipped in the kernel (`ezinfo.php`, `extension.xml`) and the generator of new extensions (`./console exp:ini:move`) write the header form.
- `composer.json` lists both holders as authors.
- Every third-party notice and the notice of eZ Systems AS stayed exactly as it was.
- A test (`CopyrightNoticeTest`) and a check script fail if a file names 7x or the Foundation in any other form.

The extensions that live in repositories of their own (the se7enxweb extensions and their `-ezpackage` twins, Exponential Velocity, the installers and site packages) follow in their next releases, each with its version raised in `ezinfo.php` and `extension.xml`.

Why the Exponential Foundation is named first

The order of the names is deliberate. The **Exponential Foundation** is named first because it is the steward of the project: the nonprofit that operates Exponential for its users and community, and that is meant to carry the software's rights, its license and its history through the whole life of a release line, independent of the fate of any one company. **7x** is named second and remains a full holder: the company that leads the development, maintains the code, reviews and merges the contributions and supports the customers.

Naming the Foundation first says, in the place every user and every lawyer looks first, that Exponential is not the product of a single vendor. That is the lesson of the software's own history: when the original developer stopped the free product, users who had relied on it for years had to find a new steward. A notice that puts a nonprofit first, with the company that does the work next to it, is the project's answer to that history.

The order is the same in every form and every language: "Exponential Foundation and 7x", never "7x and the Exponential Foundation" and never "7x & Exponential Foundation".

The two forms and where each is used

The visible notice

> Exponential Copyright ©1998-2026 Exponential Foundation and 7x

Used wherever a person reads it on a screen or a page:

Place	File
Footer of the administration (admin, admin3, admin4, admin4l, editor)	`design/admin/templates/page_copyright.tpl`, `design/admin4/templates/page_copyright.tpl`
Sign-in page of the administration	`design/admin/templates/page_login_copyright.tpl`, `design/admin4/templates/page_login_copyright.tpl`
Footer of the standard design (plain, print, popup, full screen, login and OAuth layouts)	`design/standard/templates/page_copyright.tpl`
Footer of the base design and its sign-in page	`design/base/templates/pagelayout.tpl`, `design/base/templates/loginpagelayout.tpl`
Setup wizard	`design/standard/templates/setup_pagelayout.tpl`, `design/standard/templates/setup/page_copyright.tpl`
Page of an undefined module	`design/standard/templates/undefinedmodule.tpl`
Fatal error page	`lib/ezutils/classes/ezexecution.php`
The copyright page (/ezinfo/copyright)	`kernel/private/classes/views/ezinfo/copyright.php`, `design/*/templates/ezinfo/copyright.tpl`
The help: the license box of each page, the PDF of a page	`settings/exphelp.ini [License]`, `kernel/classes/help/exphelplicense.php`
`./console --version`	`bin/php/console`
The system card	`share/syscard/about.txt`

In HTML the sign is written `©`. Links may sit on the words (the admin footer links "Exponential", "Exponential Foundation" and "7x"), but the text a reader sees is always exactly the notice.

The header form

> Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.

Used in files and metadata: the `@copyright` line of PHP docblocks, the `// COPYRIGHT NOTICE:` line of JavaScript and C++ sources, comment headers of templates, style sheets and scripts, `<copyright>` of `extension.xml`, `'Copyright'` of `ezinfo.php`, `LICENSE` and `COPYRIGHT.md`. "(C)" stands for the copyright sign in plain ASCII, which every file can hold. "All rights reserved" is the traditional closing of a header notice; it does not take away any freedom the GPL gives you (see Questions people ask).

What "copyleft" means

Copyleft is a way of using copyright to keep software free. The holders use their copyright to attach a license that says: anyone may use, study, change and share this work, **on the condition that** everyone who passes it on, changed or not, passes on the same freedoms with it. The GNU General Public License is the best-known copyleft license.

The copyright notice and the copyleft belong together. Without a holder, there is nobody who can grant the license or enforce it; without the license, the holder's copyright would forbid copying. The notice says who can enforce the GPL on behalf of everyone who relies on it. That is why the project keeps it precise.

What the GPL means for you

Exponential's license is the **GNU General Public License, version 2 or any later version**. In short:

If you use Exponential (site owners, editors, administrators)

- You may **run** it for any purpose, commercial or not, on as many servers and sites as you like. No fee, no registration, no license key.
- You may **change** it for your own use. Changes you only run on your own servers, without giving copies to others, create no duty to publish them.
- The notices stay in place. You may not remove the copyright notice or the license from the files, and a footer you change in an override template should keep the notice visible when it shows one.

If you build sites for others (integrators, agencies, hosting companies)

- You may **deliver** Exponential to customers, installed or as files, and charge for your work, your support and your hosting.
- When you give a customer a copy (the files of the installation, a repository, an archive), the GPL travels with it: keep `LICENSE`, `COPYRIGHT.md` and every notice, and give the customer the source code of what you deliver, including your changes to the kernel.
- Hosting a site for a customer on your servers, without handing over the files, is not distribution under GPL version 2.
- See GPL obligations when distributing and License texts and notices to ship.

If you write extensions (extension authors)

- **Your extension is your work.** Its copyright belongs to you (or your employer), and its notice names you, not 7x and not the Foundation.
- An extension that runs inside Exponential and uses its classes and templates is generally treated as a work based on Exponential, so when you **distribute** it, distribute it under the GPL version 2 or later, or a license compatible with it. Extensions you keep for your own sites need no license at all.
- If your extension contains code copied from Exponential, keep the notice of that code ("Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.") next to yours. See Attribution and copyright notices.

If you fork or redistribute Exponential

- You may fork it, rename your fork and publish it. Keep every notice, add your own for your changes, and keep the license GPL-2.0-or-later.
- You may not present your fork as the Exponential project or use the notice to suggest that the Foundation or 7x stand behind it.

The support window and why the notice matters for it

Exponential 6.x is meant to be used for a long time: the project plans releases and fixes for this line over a support window of **twenty years and more**, in line with a code base whose history goes back to 1998 and that its users have relied on for more than two decades. A release made in 2026 must still be clear about its rights in 2046.

A clear, single notice protects that window:

- **Enforcement.** Copyleft only works if the holders can be named and can act. One notice, with two holders that can each act, leaves no doubt who can defend the GPL for the users.
- **Continuity.** With the Foundation as a holder, the rights do not depend on the life of one company.
- **Due diligence.** Buyers, public bodies and legal departments read the notice first. One form, everywhere, with the same years and names, answers their question in seconds.
- **No drift.** The test that checks every notice stops new files from reintroducing old spellings.

Contributors and the CLA

Exponential is written by many people. Their contributions reach the kernel in two ways:

- **Code contributions to the kernel** (the repository se7enxweb/exponential) are merged after the author has signed the **Exponential Foundation and 7x Contributor Licensing Agreement**, version 1.1.0 (*Exponential Foundation and 7x CLA v1.1.0.pdf* on the CLA page). It assigns the copyright in the contribution to the **Exponential Foundation and 7x** jointly (section 4.1), which become its owners, the Foundation named first as in the notice. The two holders of the notice are thereby also the holders of every merged contribution. Version 1.1.0 (29 October 2026) is the first assignment: the earlier version 1.0.1 only granted 7x a perpetual, worldwide, non-exclusive, royalty-free, irrevocable copyright license (its section 4, from the Apache Individual Contributor License Agreement 2.0), and contributions made under a signed 1.0.1 stay licensed to 7x under it (section 12 of 1.1.0). Section 4.3 keeps a license as a fallback where an assignment is not effective; under section 6 the work is released under the GNU GPL version 2 or later. Read the agreement before you sign: The CLA explained, `CONTRIBUTING.md`.
- **Documentation contributions** are part of the same repository (`doc/`) and covered by the same CLA; the documentation is published under the GNU Free Documentation License 1.3 or later (see The license of this documentation).

There is no separate copyright of "the Exponential contributors". Every contributor assigns the copyright in a contribution to the Exponential Foundation and 7x through the CLA, so the project has exactly two copyright holders and the notice names only those two, in code and in the documentation alike. Earlier versions of the documentation's notice added "and the Exponential contributors"; that wording is gone (owner decision of 29 October 2026).

Contributors are credited, and the credit is never removed: in the changelogs, in the authorship of every commit (the git history is the record of who wrote what), on the recognition pages and in the list of contributors on the about page (/ezinfo/about). Credit is not a copyright claim.

Your own extensions, themes and site packages that you keep in your own repositories are **not** contributions and need no CLA. They carry your notice.

Notices that stay: eZ Systems AS and third-party software

The GPL requires every copyright notice that came with the code to stay. The project never changes, removes or reorders them. The ones kept on 2026-10-29:

Notice	Where	Why it stays
"Copyright (C) 1999 - 2014 eZ Systems AS. All rights reserved." and "@copyright Copyright (C) eZ Systems AS. All rights reserved."	`LICENSE`, `COPYRIGHT.md`, about 1,390 PHP files, the `// COPYRIGHT NOTICE` lines of 39 JavaScript files, the /ezinfo/copyright page	eZ Systems AS developed eZ Publish, the discontinued product Exponential grew from. Their notice stays in every file that contains their code, below the project's own line.
"Copyright (C) 1999 - 2026 7x / Brookins Consulting / eZ Systems AS"	`lib/eztemplate/classes/eztemplateleoperator.php`	A joint notice of Brookins Consulting (7x's name before 2022) and eZ Systems AS for the original operator; the approved line is directly above it.
"Copyright CJW Network and Exponential Foundation. All rights reserved."	about 40 files (the image editor `expimageeditor`, the package archive classes, a number of tests)	The notice of CJW Network, a co-author of these files; another author's notice is kept as it is.
"Copyright (C) Mugo Web, GPL v2"	templates and classes with code taken from LoveStack	Code derived from Mugo Web's work keeps its notice and names the commit it came from.
"Copyright (C) 1989, 1991 Free Software Foundation, Inc."	`LICENSE`, `LICENSE.md`, the `LICENSE` files of ezjscore, ezoe and ezformtoken	Part of the license text, which must never change.
"Copyright ..., Moxiecode Systems AB" and "Ephox Corporation DBA Tiny Technologies, Inc."	TinyMCE in `extension/ezoe`	The editor component bundled with the online editor.
"Copyright (c) Unicode, Inc., Unicode License v3"	`share/locale` (CLDR data)	Locale data from the Unicode CLDR.
"Copyright (C) 2009 Nokia Corporation and/or its subsidiary(-ies)", "Trolltech AS"	`support/ezlupdate-qt4.5`	The Qt Linguist sources of the translation updater.
"Copyright (C) 2002 and following years KDE Artists"	`share/icons/crystal*`	The Crystal icon set.
"Copyright (C) 1999-2010 Tango_Icon_Library"	`extension/ezoe`	Icons of the online editor.
The book notices ("Copyright (c) ... 7x and Graham ...")	documentation about the books	The books are separate works with their own copyright.

The complete list of these exceptions, with the reason for each, is the file `tests/tests/bin/copyright-notice-allowlist.json`. Notices that name only other holders need no entry there: the test recognizes them as third-party notices and leaves them alone.

The libraries in `vendor/` are not part of this list: they are separate packages with their own license and notices. The kernel's libraries, the **expcomponents** (since 6.0.15 Exponential's fork of the Zeta Components, licensed under the Apache License 2.0), keep the notices of the Zeta Components development group, the Apache Software Foundation and eZ Systems AS in each package's `LICENSE.txt` and in their source files, as Apache 2.0 requires; the fork adds its own changes under the same license (Exponential Components).

Writing the notice in new files

Every new file of the kernel starts with the header form, in the comment style of its language. Keep the layout of the surrounding files: the `@copyright` line, then the `@license` line.

PHP

PHP

```php
<?php
/**
 * File containing the expExampleThing class
 *
 * @copyright Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
 * @license GNU General Public License v2.0 (or any later version)
 * @package kernel
 */
```

A file that also contains code of eZ Systems AS keeps their line directly below:

PHP

```php
 * @copyright Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
 * @copyright Copyright (C) eZ Systems AS. All rights reserved.
 * @license For full copyright and license information view LICENSE file distributed with this so
urce code.
```

JavaScript

JavaScript

```javascript
/**
 * Exponential: the bookmark button of the page header.
 *
 * Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
 * GNU General Public License v2.0 (or any later version).
 */
```

Older JavaScript and C++ files use the `// COPYRIGHT NOTICE:` style; keep it in those files:

JavaScript

```javascript
// COPYRIGHT NOTICE: Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved
.
// COPYRIGHT NOTICE: Copyright (C) 1999-2014 eZ Systems AS
```

Templates (.tpl)

```
{* The footer of the example page.
   Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
   GNU General Public License v2.0 (or any later version) *}
```

INI files, shell and Python scripts

INI

```ini
# Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
# GNU General Public License v2.0 (or any later version)
```

Style sheets

CSS

```
/* Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
   GNU General Public License v2.0 (or any later version) */
```

Rules for all of them:

- Write the notice **exactly**: "Copyright (C) 1998 - 2026", a space on each side of the dash, "Exponential Foundation and 7x", a full stop, "All rights reserved." with its full stop.
- Never "7x & Exponential Foundation", "7x and the Exponential Foundation", "7x / Exponential Foundation", "7x and others" or "7x" alone.
- Never remove or reword another holder's line. Add yours above it.
- Do not change the years of an existing line to make it "fit"; the project changes the end year for all files at once, at the turn of the year.

Writing the notice in an extension

An extension of the project (shipped in the kernel or released by se7enxweb)

`ezinfo.php`:

PHP

```
<?php
/**
 * @copyright Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.
 * @license GNU General Public License v2.0 (or any later version)
 */
class exampleextInfo
{
    static function info()
    {
        return array( 'Name' => "Example extension",
                      'Version' => "1.0.0",
                      'Copyright' => "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. Al
l rights reserved.",
                      'License' => "GNU General Public License v2.0 (or any later version)",
                      'Info_url' => "https://github.com/se7enxweb/exampleext" );
    }
}
```

`extension.xml`:

XML / HTML

```
<metadata>
    <name>Example extension</name>
    <version>1.0.0</version>
    <copyright>Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved.</cop
yright>
    <license>GNU General Public License v2.0 (or any later version)</license>
    <info_url>https://github.com/se7enxweb/exampleext</info_url>
</metadata>
```

An extension that contains code of eZ Systems AS keeps both notices in one value, the project's first:

XML / HTML

```
<copyright>Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved. Copyrigh
t (C) eZ Systems AS. All rights reserved.</copyright>
```

`./console exp:ini:move` writes exactly these files when it creates a settings extension.

Your own extension

Your extension carries **your** notice, for example:

PHP

```
 * @copyright Copyright (C) 2026 Example Agency GmbH. All rights reserved.
 * @license GNU General Public License v2.0 (or any later version)
```

Only if you copied code from Exponential into it, add the project's line for that code. Do not put the project's notice on code you wrote: it would give your copyright away on paper.

Showing the notice in a design or theme

A front-end theme does not have to show the notice. Many sites show their own footer instead, which is their right. If your design shows the software's notice, show it exactly:

```
<a href="https://exponential.earth">Exponential</a> Copyright &copy; 1998-2026 <a href="https://f
oundation.exponential.earth">Exponential Foundation</a> and <a href="https://se7enx.com">7x</a>
```

An override of `page_copyright.tpl` in your design may move the notice, style it or wrap it in links, but should not change its words.

Translations: the notice is never translated

The notice is a legal statement and a name, so it reads the same in every language. In the catalogues (`share/translations/<locale>/translation.ts`) the footer message's translation is the notice itself, in all 156 catalogues; `translation-identical-allowlist.json` lists it as allowed to stay identical. Translators translate the words around it ("For more information see ezinfo/about", "Powered by ...", "Help center"), never the notice. A translation that changes the notice is a bug.

The German documentation explains the notice in German, but quotes it in its original form: "Exponential Copyright ©1998-2026 Exponential Foundation and 7x".

How the project checks the notice

- `tests/tests/bin/CopyrightNoticeTest.php` (PHPUnit, testsuite `quality`, no database) reads every file of the repository and fails on any line that names 7x or the Exponential Foundation as a copyright holder in a form other than the two approved ones, unless the line is listed in `tests/tests/bin/copyright-notice-allowlist.json` with its reason.
- `tests/tests/design/CopyrightFooterNoticeTest.php` checks the footers, the sign-in pages, the setup wizard, the error page, the console, the copyright page and the help, and that no catalogue translates the notice.
- Both run with the continuous integration on every push; the change-based test runner maps changes of any file to the first test.

A line counts as a notice when it has the word "copyright" and a notice marker: "(C)", "©" (or `©`), a year, `@copyright`, `<copyright>`, `'Copyright' =>`, or `Copyright=` / `Copyright:`. Prose such as "the Foundation and 7x hold the copyright" is not a notice and is not checked.

The documentation's own license

This documentation is free documentation under the GNU Free Documentation License, version 1.3 or any later version. Its notice names the same holders; see The license of this documentation and Free documentation. The pages inherited from the original software keep the notice of eZ Systems AS and their own terms.

Questions people ask

Does "All rights reserved" contradict the GPL? No. It is the traditional ending of a copyright notice and means the holders keep their copyright. The GPL is the permission the holders give on top of it; you keep every freedom the license grants.

Why 1998? 1998 is the first year of the notice of the project's own work, as it has read since the 6.x line began. eZ Systems AS's work is dated separately, 1999 to 2014.

Does the order of the names mean the Foundation owns more? No share is implied by the order. Both are holders. The Foundation is named first for the reasons above.

To whom do I assign my contribution under the CLA? To the Exponential Foundation and 7x together, which become its owners and may use, change, relicense and distribute it. Since version 1.1.0 of the CLA (29 October 2026) the agreement names both, in the same order as the notice; version 1.0.1 had been a non-exclusive license to 7x, not an assignment. Your own extensions are not affected.

Why doesn't the notice name the contributors? Because they hold no separate copyright in Exponential. Every contribution is assigned to the Exponential Foundation and 7x through the Contributor Licensing Agreement, so those two are the only holders, and a notice names holders. The people who did the work are credited by name in the changelogs, the commit history and the recognition pages; that credit is not a copyright claim.

Did the license change on 29 October 2026? No. Only the holder notice changed. The license is still the GNU General Public License, version 2 or any later version, and its text is unchanged.

I quoted the old notice in my own documentation. Do I have to change it? Your text is yours. We recommend quoting the current notice in anything new.

My extension's `ezinfo.php` says "Copyright (C) 1998 - 2026 7x & Exponential Foundation". Is that wrong now? If the extension is one of the project's, its next release fixes it. If it is your own extension that copied the project's metadata as a template, replace the notice with your own.

Can I remove the notice from the footer of my site? The GPL is about the source code: keep every notice in the files. A footer is part of your site's design, and a customized front-end footer does not have to show the software's notice. The administration's footer belongs to the software; leave it as it is.

Where to read more

- `LICENSE` and `LICENSE.md`: the license text, with the holder notice above it.
- `COPYRIGHT.md`: the copyright notice of the project, with this history.
- The license of this documentation

- Free documentation: use it to help yourself and others
- Attribution and copyright notices
- Copyright and author lines of an extension
- Copyright notices in files
- The CLA explained
- GPL obligations when distributing
- The Exponential Foundation and 7x
- The 6.0.15 changelog

Free documentation: use it to help yourself and others

This help is free documentation under the GNU Free Documentation License 1.3 or later.

The help you are reading is **free documentation**. It is published under the GNU Free Documentation License, version 1.3 or any later version, the license the Free Software Foundation wrote for manuals and guides. "Free" here is about freedom, not price: you may use these pages, pass them on and make them better, and so may everyone you pass them on to.

This page explains in plain words what that lets you do. It is a summary, not legal advice: the license text is what counts. Read it in the help (GNU Free Documentation License 1.3) or on the web at https://www.gnu.org/licenses/fdl-1.3.html. Who holds the copyright, and which older pages keep different terms, is on The license of this documentation.

What you can do

Read and study it

Read any page as often as you like, for any purpose: to learn the system, to train for a job, to prepare a course or to check how something works before you change it. No permission is needed to read or study.

Copy it and share it with other users

You may copy pages, or the whole documentation, and give or send the copies to anyone, on paper or as files, for free or for a fee, in any medium. Print a page for a colleague, put the guides on your intranet, attach a page to a support ticket, or hand a printed set to the editors of your team. When you do, keep with every copy:

- the **copyright notices** of the pages,
- the **license notice**, the paragraph that says the GNU Free Documentation License applies, and
- a **copy of the license** itself (the file `doc/eng-us/LICENSE-GFDL-1.3.txt`, or the license's text in the same document).

Do not add conditions of your own, and do not use technical measures (copy protection, for example) that stop the people you give copies to from reading or copying them further.

If you give out more than 100 copies in a format that is hard to edit, such as a PDF without its source, you also have to provide an editable copy (the Markdown files of `doc/` serve), or tell readers where they can get one. More than 100 printed copies need a cover that shows the full title and names you as the publisher. Section 3 of the license gives the details.

Translate it

A translation is a modified version of the documentation (see the next section), so it is allowed on the same terms. You may also translate the license notices; the original English text of the license and of the notices must go with your translation, and where the two differ, the English original counts.

Adapt and improve it, and share your version

You may change pages: correct them, shorten them, extend them, rewrite them for your own site or team, or take sections into your own manual. You may share your changed version as well, as long as:

- your version is published under the **same license**, the GNU Free Documentation License, so that the people who receive it get the same freedoms you had;
- it **keeps the copyright notices** of the original and adds your own notice for your changes;
- it **says that it was changed**, and by whom (the license asks for the authors of the changes in the title page and a "History" section that lists the versions);
- it **carries the license notice** and a copy of the license;
- it does not pretend to be the original: give your version a title of its own, unless the publisher of the original agrees that you keep its title.

Section 4 of the license lists every condition for modified versions.

Combine it with other works

You may put these pages next to other, independent documents in a collection or on the same medium (on a website with your own pages, or a training folder with your slides). The license of these pages does not extend to your own, independent works; it applies to the pages from this documentation and versions made from them.

Cite it

Point to a page by its title and its address in the help, or by its file in the repository (`doc/eng-us/help/getting-started.md`, for example). Quoting a short passage with its source is normally possible under copyright law anyway; when you copy whole pages or large parts, follow the conditions above.

Help others

Free documentation gets better every time someone uses one of these freedoms and shares the result.

- **Share what helped you.** When a page solved your problem, send it to the next person with the same question.
- **Report what is wrong.** If a page is out of date, unclear or wrong, tell the project: Help, *Report an error* shows where and how.
- **Contribute your fixes back.** Every help page has an *Edit on GitHub* link to its source file. Correct it and open a pull request (Develop with Git shows how, step by step). A change to these pages is accepted under the same license, the GNU Free Documentation License 1.3 or any later version, so it stays free for everyone.
- **Offer your translation.** A translated page helps every reader of that language; contribute it the same way.

If the conditions are not kept

The license gives these freedoms on its conditions. Someone who copies, changes or shares the documentation in a way the license does not allow loses the rights the license gave them; section 9 of the license explains how those rights come back once the problem is fixed.

Older pages with their own terms

A few older files in `doc/`, inherited from the documentation of the original software by eZ Systems, are not under this license; the help shows their own notice on those pages. The license of this documentation lists them.

Copyright and copyleft

Since 29 October 2026 the copyright of Exponential is held by the Exponential Foundation and 7x together, the Foundation named first: "Exponential Copyright ©1998-2026 Exponential Foundation and 7x", in files "Copyright (C) 1998 - 2026 Exponential Foundation and 7x. All rights reserved." Who holds it and why, what the GPL lets you do, the third-party notices that stay and how to write the notice in new files and extensions: Copyright and copyleft.

GNU Free Documentation License 1.3