Release engineering

How releases of Exponential and its extensions are made - the main branch and its 6.0.x-dev alias, where the kernel version lives, the extension release procedure with ezinfo.php and extension.xml, the file manifest...

Source file: doc/eng-us/help/software-engineers/release-engineering.md Last changed 2026-11-04
https://doc.exponential.earth/exphelp/viewer/help/software-engineers/release-engineering.md

This page is for engineers who cut releases of the kernel or of an extension, and for those who need to know what a release guarantees. The rules come from Source control and releases and Compatibility, releases and deployment; the course for extension maintainers is Releasing.

Contents

1. Lines and versions
2. Before you tag
3. Releasing an extension
4. Publishing
5. When a release is bad

Lines and versions

- `main` is the development line. `composer.json` aliases it as `6.0.x-dev` (`extra.branch-alias`), and CI runs on it.
- The kernel version lives in `lib/version.php` (`VERSION_MAJOR`, `VERSION_MINOR`, `VERSION_RELEASE`) and in the database row `ezsite_data` `ezpublish-version`, which the update SQL writes. The root `composer.json` has no `version` field; Composer takes the version from the tag.
- Kernel releases are tagged `v6.0.<n>`. Tags `v6.0.0` to `v6.0.14` exist; 6.0.15 is the current line after them, released as the tag `v6.0.15` once it is published.
- Extensions are separate Composer packages with their own versions.

Choosing a number: Version numbers and immutable tags.

Before you tag

A tag is cheap to get right and impossible to correct. Before tagging:

- ☐ The working tree is clean.
- ☐ Quality and PHPUnit are green on the commit you will tag (CI gates).
- ☐ The version fields are bumped in the commit the tag points to.
- ☐ `share/filelist.md5` matches the files (`php bin/php/checkmanifest.php --all`).
- ☐ The changelog page for the version exists (Changelogs and upgrade notes).
- ☐ Every upgrade-relevant change has its page in `doc/eng-us/bc/`.

☐ Build artifacts committed to the repository match their sources.

Releasing an extension

Every release writes its number into the extension, in the commit the tag points to:

1. Choose the number from what is already published.
2. Set `'Version'` in `ezinfo.php` and `<metadata><version>` in `extension.xml` to it, without the `v` prefix (tag `v1.8.4` gives `1.8.4`). Never leave `//autogentag//` or `dev-master`. Both files also state the license and an `info_url` on the extension's public repository.
3. Move any other version field the repository keeps (`version` in its `composer.json`, the header of its own `share/filelist.md5`) and refresh that manifest's checksums.
4. Check: `php -l ezinfo.php` and `xmllint --noout extension.xml`.
5. Commit, **then** tag that commit, then push both. Never tag first and bump afterwards.

Extensions shipped inside the kernel repository carry the kernel's version. The about page reads both files (About and package pages, Extension metadata).

Publishing

- **GitHub release**: made from the tag, with notes taken from the changelog (GitHub releases and release notes).
- **Packagist**: each `se7enxweb/*` repository is a Composer package. A webhook on the GitHub repository tells Packagist about new tags; a package without it stays at an old version until updated by hand (The Packagist webhook).
- **Repinning the root**: release the extension, see the version on Packagist, then commit the root's new constraint. The push gate refuses a constraint Packagist cannot satisfy yet, because CI's install would fail on every leg.

When a release is bad

Do not move, delete or re-cut the tag, and do not replace the release's assets. Publish the next version with the fix, and say plainly in its notes what was wrong with the previous one. Consumers who pin ranges pick up the new version; nobody is helped by a hole in the history.

Related pages

- The release checklist
- Versions and releases, for application creators
- Dependency management with Composer
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are

> planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

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