



Roadmap to Plone 6.x and Plone 7

Plone 6/7 Roadmap

Timo Stollenwerk, Maurits van Rees, Victor Fernandez de Alba

September 2024

Preface	4
Previous Roadmaps	4
Status Quo	4
Plone 6.1	4
Teaser and Grid Blocks (PLIP #4207)	4
Anchor Links (PLIP #4298)	4
Link Integrity View (PLIP #4339)	5
Add Index for Blocks to ZCatalog (PLIP #4778)	5
Image Optimizations with Sizes and Srcsets (PLIP #1428)	5
Listing Block Variants for Page/News Item/Event content type (PLIP #4765)	6
Relations Control Panel (PLIP #3383)	6
TypeScript support (PLIP #4689)	6
Upgrade to React 18 (PLIP #5641)	7
Slots (PLIP #5775)	7
Make plone.app.multilingual a core add-on (PLIP #3956)	7
Make plone.app.discussion a core add-on (PLIP #3958)	7
Modernize Core Development Toolchain to the latest Python Community Best Practices (PLIP #3670)	8
Plone Distributions (PLIP #3854)	8
TinyMCE 7 in Classic UI (PLIP #3954)	8
New ContentBrowser pattern (PLIP #14)	8
Drop Python 3.8 and 3.9 (PLIP #3926)	9
Switch Package Manager from Yarn to pnpm (PLIP #5634)	9
Javascript bundle optimization (PLIP #5419)	9
Plone 6.2+	9
Improve Drag and Drop (PLIP #4204)	9
Enhanced Contents View (PLIP #4756)	9
Improve Link Widget (PLIP #4303)	10
Keyword Manager in Core (PLIP #5300)	10
Provide a simple ImageUploadWidget that can be reused in add-ons (PLIP #4286)	10
Relative Path Support (PLIP #4290)	10
Update i18n machinery to the latest react-intl (PLIP #5627)	10
Users/Groups Control Panel Overhaul (PLIP #5962)	10
Start separating Classic UI code from the core (PLIP #3953)	11
Plone 7	11
Quanta Toolbar (PLIP #4206)	11
Refactor Class Components into Functional Components (PLIP #4460)	11
Modernize Data Fetching (PLIP #4347)	11
Quanta (PLIP #4332)	11
Extract basic Volto components to an add-on (PLIP #4352)	12

Move all Classic UI code into core add-ons (PLIP #3955)	12
Update to the latest Namespace Best Practices of the Python Community, Switch to PEP 420 (PLIP #3928)	12
Classic UI: Remove deprecated mockup patterns (PLIP #4010)	12
Enhance timezone support in UI for plone.app.event based types (PLIP #5629)	13

Preface

Previous Roadmaps

The results of community discussions were set down in past [roadmap documents](#).

Status Quo

The existing Plone platform is mature, secure, flexible, and extensible. It offers a good user experience and focuses on accessibility.

Plone 6.0 was shipped in December 2022 with Volto as its default frontend. This ensures Plone has a stable and feature-rich backend for the foreseeable future and a state-of-the-art frontend stack.

Plone 6.1

Plone 6.1 is planned to be shipped in late 2024 with the following features:

Teaser and Grid Blocks (PLIP [#4207](#))

The Teaser block allows placing teasers to existing page content, and the Grid block allows sophisticated multi-column page layouts.

- PLIP: <https://github.com/plone/volto/issues/4207>
- Documentation: <https://6.docs.plone.org/volto/blocks/core/grid.html> / <https://6.docs.plone.org/volto/blocks/core/teaser.html>
- Contributors: Victor Fernandez de Alba et al.
- Status: PLIP has been accepted by the Volto team, PR has been merged and released with Volto 17 (<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>)

Anchor Links (PLIP [#4298](#))

Plone 6 generates anchor links for headlines automatically. These anchor links can be copied to the clipboard and pasted into the link widget.

- PLIP: <https://github.com/plone/volto/issues/4298>
- Contributors: Nilesch Guila, Tiberiu Ichim, Alin Voinea
- Status: PLIP has been accepted by the Volto team, PR has been merged and released with Volto 17 (<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>).

Link Integrity View (PLIP [#4339](#))

The Link Integrity view lists all content objects that currently use or link to the current object. This allows editors to clean up links and references in the view without having to click on a delete button and rely on link integrity warnings. In addition, we did a full overhaul of the link integrity feature. It now warns editors when deleting content that is still referenced elsewhere to make sure link integrity warnings work for new Plone features, such as Grid or Teaser blocks.

- PLIP: <https://github.com/plone/volto/issues/4339>
- Pull Request: <https://github.com/plone/volto/pull/4787> / <https://github.com/plone/volto/pull/5234>
- Documentation: <https://6.docs.plone.org/volto/user-manual/links-to-item.html>
- Contributors: Paul Grunewald, Dante Alvarez, Steve Piercy, Katja Süss, Roel Bruggink, Victor Fernandez de Alba, Jakob Kahl
- Status: PLIP has been accepted by the Volto team, PR has been merged and released with Volto 17 (<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>).

Add Index for Blocks to ZCatalog (PLIP [#4778](#))

Add a new index to the ZCatalog that keeps track of where blocks are used. This allows us to expose statistical information in the future, as well as a way to query the catalog for information about where Blocks are used.

- PLIP: <https://github.com/plone/volto/issues/4778>
- Pull Request: <https://github.com/plone/plone.volto/pull/121>
- Contributors: Margarida Paixao, Fred van Dijk
- Status: PLIP has been accepted by the Volto team, PR has been merged and released with plone.volto 4.1.0 (<https://github.com/plone/plone.volto/blob/main/CHANGES.rst#410-2023-08-07>).

Image Optimizations with Sizes and Srcsets (PLIP [#1428](#))

Implement a new image component in Volto that offers the browser different options (srcset) for every image. With this information, the browser can decide which image to load. This helps improve page loading times, especially on mobile devices.

- PLIP: <https://github.com/plone/volto/issues/1428>
- Pull Request: <https://github.com/plone/volto/pull/3337>
- Documentation: <https://6.docs.plone.org/volto/recipes/images.html>
- Participants: Piero Nicolli, David Glick.
- Status: PLIP has been accepted by the Volto team, PR has been merged and released with Volto 17

(<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>).

Listing Block Variants for Page/News Item/Event content type (PLIP [#4765](#))

- PLIP: <https://github.com/plone/volto/issues/4765>
- Pull Request: <https://github.com/plone/volto/pull/6055>
- Documentation: <TODO>
- Participants: Alok Kumar, Dante Alvarez, Timo Stollenwerk
- Status: PLIP has not been accepted by the Volto team, we are currently working on this.

Relations Control Panel (PLIP [#3383](#))

A new control panel for managing relations has been added to Volto. These are the main features:

- Inspect/Edit/Batch edit relations
- Add relations to arbitrary targets
- Fix broken relations
- PLIP: <https://github.com/plone/volto/pull/3382>
- Participants: Katja Süss
- Status: PLIP has been accepted and released with Volto 17
(<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>)

TypeScript support (PLIP [#4689](#))

TypeScript support has been added to Volto core. ESLint has been configured to accept code bases written both in JavaScript and in TypeScript. TypeScript in projects and add-ons has been supported since Volto 14. It is now supported in the project generator by default.

- PLIP: <https://github.com/plone/volto/issues/4689>
- PR: <https://github.com/plone/volto/pull/4662>
- Documentation: <https://6.docs.plone.org/volto/contributing/typescript.html>
- Status: PLIP has been accepted and released with Volto 17
(<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>)

Upgrade to React 18 (PLIP [#5641](#))

Volto has been upgraded to use React 18 in Volto 18. This will allow us to keep up to date with the latest developments in the React ecosystem, and be prepared for the upcoming React releases.

- PLIP: <https://github.com/plone/volto/issues/5641>
- PR: <https://github.com/plone/volto/pull/3221>
- Status: The PLIP has been merged and released with Volto 18.0.0-alpha.15 (<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1700-2023-09-30>)

Slots (PLIP [#5775](#))

Slots provide a way for Volto add-ons to insert their own components at predefined locations in the rendered page.

This concept is inspired by the Plone Classic UI viewlets.

- PLIP: <https://github.com/plone/volto/issues/6021>
- PR: <https://github.com/plone/volto/pull/5775>
- Status: The PLIP has been merged and released with Volto 18.0.0-alpha.14 (<https://github.com/plone/volto/blob/main/packages/volto/CHANGELOG.md#1800-alpha14-2024-03-01>)

Make plone.app.multilingual a core add-on (PLIP [#3956](#))

Most Plone sites do not need multilingual support, so we made this optional. We made this a dependency of the Plone package instead of the Products.CMFPlone package.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3956>
- Status: available since Plone 6.1.0a1

Make plone.app.discussion a core add-on (PLIP [#3958](#))

Most Plone sites do not need support for comments or discussions, so we made this optional. We made this a dependency of the Plone package instead of the Products.CMFPlone package. If you want discussions or comments in your site, you need to include the plone.app.discussion add-on, and activate it in the add-ons control panel. After activation, discussions are then globally enabled, which as before means they can be switched on per content item or portal type, depending on how you have configured your portal types.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3958>
- Status: available since Plone 6.1.0a5.

Modernize Core Development Toolchain to the latest Python Community Best Practices (PLIP [#3670](#))

This is a technical improvement, not visible for end users. In the Plone 6.0 backend, the pip constraints.txt file is generated from the Buildout versions.cfg file when a release is made. In Plone 6.1, they will be at the same level, where both files will be kept in sync all the time. This moves us more to using standard tools from the Python ecosystem, instead of building and maintaining our own infrastructure.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3670>
- Status: Done. The pip constraints are kept in sync with the buildout versions. Related improvements are ongoing, but nothing of that is relevant for end users.

Plone Distributions (PLIP [#3854](#))

In Plone 6.1 we will have an API to create “distributions” of Plone. Technically, both Volto and Classic UI could be called Plone distributions. We will make a way to define a Python package as a Plone distribution. You can define your own list of add-ons that get installed when creating a new Plone site, and also define default content.

An example use would be for a Plone company to make a distribution with their favorite add-ons, including sample content that uses these add-ons, and then easily create a demo site for a new customer.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3670>
- Status: Work in progress

TinyMCE 7 in Classic UI (PLIP [#3954](#))

The Classic UI uses the TinyMCE rich text editor. In Plone 6.0 we use TinyMCE version 5. This reached end of support in April 2023. In Plone 6.1 we upgraded to TinyMCE version 7.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3954>
- PR: <https://github.com/plone/mockup/pull/1346> and <https://github.com/plone/mockup/pull/1378>
- Status: done, and available since Plone 6.1.0a3.

New ContentBrowser pattern ([PLIP #14](#))

Add pat-contentbrowser as a replacement for pat-relateditems and add plone.app.z3cform contentbrowser widget as a replacement for relateditems widget.

Drop Python 3.8 and 3.9 (PLIP [#3926](#))

This means less burden on maintainers within the Plone community, like the release team and A/I team (including testing), and opens up the possibility for Plone core developers to use some more recent features. For inspiration, here is the Python document explaining what is new in each Python version:

<https://docs.python.org/3/whatsnew/index.html>

See what's new in [3.9](#) and [3.10](#) for what we can now use in 6.1 that was not possible earlier, for example the `zoneinfo` module and structural pattern matching (`match/case`).

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3926>
- Status: Done since the second alpha release of Plone 6.1

Switch Package Manager from Yarn to pnpm (PLIP [#5634](#))

pnpm is faster and better engineered than Yarn.

- PLIP: <https://github.com/plone/volto/issues/5634>
- Status: Done with Volto 18

Javascript bundle optimization (PLIP [#5419](#))

Optimize the JavaScript bundle for a smaller bundle size.

- PLIP: <https://github.com/plone/volto/issues/5419>
- Status: Done with Volto 18

Plone 6.2+

Improve Drag and Drop (PLIP [#4204](#))

Refactor the current Volto drag and drop infrastructure and components to use the react-aria library. Extend the feature set to support moving blocks between the main blocks stream and block containers (row block, grid block, group block, etc).

- PLIP: <https://github.com/plone/volto/issues/4204>
- Status: Work in Progress (WIP)

Enhanced Contents View (PLIP [#4756](#))

Refactor the contents view:

- Use own core package `@plone/contents`
- Use TypeScript

- Better slicing of components so they are easily shadowable
- Don't use SemanticUI on it, use react-aria-components
- Use @plone/client instead of the Volto redux-based request API
- PLIP: <https://github.com/plone/volto/issues/4756>

Improve Link Widget (PLIP [#4303](#))

Improve the overall UX for the "add a link to internal content or an external URL".

- PLIP: <https://github.com/plone/volto/issues/4303>

Keyword Manager in Core (PLIP [#5300](#))

Implement the base functionality of Products.PloneKeywordManager in Volto / Plone Core.

- PLIP: <https://github.com/plone/volto/issues/5300>

Provide a simple ImageUploadWidget that can be reused in add-ons (PLIP [#4286](#))

Add a new self-contained widget with the image upload feature, like the one used in image block, for reusability in Volto core and in add-ons.

- PLIP: <https://github.com/plone/volto/issues/4286>

Relative Path Support (PLIP [#4290](#))

Add support to deploy Volto under a non-rooted directory.

- PLIP: <https://github.com/plone/volto/issues/4290>

Update i18n machinery to the latest react-intl (PLIP [#5627](#))

Upgrade to latest react-intl all of Volto's internal machinery. This includes @plone/scripts i18n helpers. In the process, upgrade @plone/scripts to ESM.

- PLIP: <https://github.com/plone/volto/issues/5627>

Users/Groups Control Panel Overhaul (PLIP [#5962](#))

Improve usability and feature set of the existing users/groups control panel.

- PLIP: <https://github.com/plone/volto/issues/5952>

Start separating Classic UI code from the core (PLIP [#3953](#))

- Separate Classic UI code from the core code to support Plone Distributions and for a smaller Plone core to support the Plone headless CMS efforts.
- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3953>
- Package: <https://github.com/plone/plone.classicui/>

Plone 7

Quanta Toolbar (PLIP [#4206](#))

The Quanta Design System for Volto features a common interface to interact with blocks in the block engine, called the Quanta Toolbar. It's a Volto CMSUI artifact that acts as a placeholder of buttons, handlers and menus that allows the user to interact with a block's settings.

Related Quanta PLIP: <https://github.com/plone/volto/issues/4332>

- PLIP: <https://github.com/plone/volto/issues/4206>

Refactor Class Components into Functional Components (PLIP [#4460](#))

Refactor all the possible components in Volto that still are class-based components. Convert them to functional components, whenever possible.

- PLIP: <https://github.com/plone/volto/issues/4460>

Modernize Data Fetching (PLIP [#4347](#))

GSoC project that @sneridagh will lead. This will allow us to move outside the boundaries of Volto, and query the Plone headless CMS from any framework. It should be included in core by default in Plone 7, and available as an add-on before.

- PLIP: <https://github.com/plone/volto/issues/4347>
- PR/PoC: <https://github.com/plone/plone.restapi-client>

Quanta (PLIP [#4332](#))

Quanta is not only the UX/UI that Albert designed, but applies a new theming story. The blueprints of the approach are already in the PLIP. These days Adobe has published

react-aria-components, a headless library of components based in react-aria. It will allow us to tailor our interfaces to exactly what we have in mind, while being both small and approachable.

The idea is that Quanta has these parts:

- the fundamentals
- the CMSUI
- the Theme UI
- PLIP: <https://github.com/plone/volto/issues/4332>
- PoC: <https://github.com/plone/volto/pull/3877>

Extract basic Volto components to an add-on (PLIP [#4352](#))

The idea of this one is to start creating a library of components (@plone/components) that is based on the “modern data fetching” PLIP (@plone/client) and on the outcome of the “Refactor class components to functional components” PLIP. This library will be used in frameworks outside Volto (and eventually by Volto itself, in Plone 7).

PLIP: <https://github.com/plone/volto/issues/4352>

PoC: <https://github.com/plone/components>

Move all Classic UI code into core add-ons (PLIP [#3955](#))

When you create a Plone 7 site with the default front end, you should not get Classic UI code, like portlets, viewlets, and most backend page templates. Some code will likely remain. The Zope Management Interface will still be available for power users, so it might make sense to still have a limited Classic UI available.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/3955>
- Package: <https://github.com/plone/plone.classicui/>

Update to the latest Namespace Best Practices of the Python Community, Switch to PEP 420 (PLIP [#3928](#))

See also <https://github.com/zopefoundation/meta/issues/194>

This will be an effort across the whole backend code base, including Zope. It will lessen our dependence on setuptools, and open up possibilities for the use of other Python packaging tools.

Classic UI: Remove deprecated mockup patterns (PLIP [#4010](#))

Removing the relateditems and select2 pattern will also remove the ancient and customized select2 module.

- PLIP: <https://github.com/plone/Products.CMFPlone/issues/4010>

- Pull Request:
 - pat-relateditems: <https://github.com/plone/plone.classicui/issues/15>
 - pat-select2: <https://github.com/plone/plone.classicui/issues/16>

Enhance timezone support in UI for plone.app.event based types (PLIP [#5629](#))

This is a collection of over two dozen features.

- PLIP: <https://github.com/plone/volto/issues/5629>
- Project board: <https://github.com/orgs/plone/projects/43/views/1>
- Design document:
 - 📖 [PLIP #5629: Enhance timezone, date, and time support in event content types](#)