



Plugins versus themes for custom code

Working with WordPress customizations always presents the question: Should they be packed in a plugin or should they be bundled with the theme they were built for?

The following sections make this assumption: CPTs (Custom Post Types) in WordPress are data, they don't provide a design to make the data usable.

The case for plugins

Building the customizations into a plugin makes them portable, they can be used regardless of what theme you're using and will still work if you switch themes.

If you move to another theme you have some work to do to make the new CPTs work with your new theme. You have to make the theme design work with the data you get from the CPTs and figure out how you want to show that data to your users.

The case for themes

Plugins only provide data, not the design that will make the data usable. From this perspective it would make more sense to add the CPTs to a theme and work directly with the theme.

If a custom post type lives in a theme then it will only be available to that specific version of a theme and will not work with any other theme, you may lose data and the new version of your site will not look as intended.

The Gutenberg (Added) Complexity

We should ask the same questions we ask of CPTs should when we talk about Gutenberg custom blocks, patterns and variations. Are they tied to a theme or do we want them to be portable and reusable?

Which one to use?

This is a case-by-case answer but most of the time, the amount of work styling items for your new theme will be far less than the amount of work you will have to do to make the new CPTs work with a new theme so, unless there is a particular reason to do otherwise, I'd suggest bundling all custom functionality in a plugin.