Microsoft Power Platform

Migrating to Power Apps modern controls from Creator Kit

Power CAT Tools

April 2023

Getting started

If you are considering migrating to the recently announced modern controls in your Power Apps, off from Creator Kit components, then we would first like to ***thank you*** and let you know how excited we are for you to experience all the improvements and features we have been working on!

When to use modern controls

- Once modern controls are generally available, you can use them in production apps (indicated by automatic presence in the controls menu, not under the preview flag)
- Experiment with modern controls in development before determining whether to use them in production.
- If the control you need is not there, you can continue to use Creator Kit controls (controls that are not in modern controls will not be deprecated)
- The properties in modern controls might not have all properties available in the Creator Kit counterparts, so use which one has the feature set you need at the time.
- If there are both controls of the same type present in your app, it will not break your app.
- Providing feedback is essential for the product team to know which properties and functionality you need in the new controls. Use the in-product feedback mechanism to share missing functionality you need to successfully adopt that control. Provide feedback in the community forum.

Expected challenges.

There will be challenges to overcome during migration. The good news is modern controls was designed to be adopted incrementally and used side-by-side with previous versions of Fluent components (such as those from the Creator Kit). This guidance is not necessarily applicable to native Power Apps canvas native controls because they are not Fluent UI controls that follow the design framework.

Design differences.

Creator Kit components and modern controls look different. The differences are subtle, but noticeable. You may decide you are okay living with some visual differences during migration.

Alternatively, you can make everything look like Fluent 1 (Creator Kit / custom pages standard controls) or Fluent 2 (modern controls), but it takes some extra effort.

- **You can choose to live with and limit style differences during migration.**
  The visual style differences between modern controls and previous versions are slight. If your migration effort is small, you can migrate all at once, or if your users are OK with some inconsistency, you might decide to avoid extra effort and live with it.

  For example, you can see differences between Creator Kit's Pivot and modern control's TabList component when you put them side-by-side. However, the design differences between v8's Pivot and v9's TabList are difficult to detect.

  You can reduce the visual friction between the two designs during migration by migrating all instances of one component type to modern controls.

- **Wait until all the controls you need are available.**
  Alternatively, you can wait it out until all the controls you want to use together are available. There is no public roadmap that the Power Apps team has announced, but the more simplistic controls in Creator Kit will likely be succeeded by the Fluent V9 version.

Custom styles.

Fluent UI's approach to theming will be changing going forward. While in the past, more theming properties allowed fine-grained control over each component's style, it led to an explosion of theme properties. Too many theme properties can make defining new themes an arduous chore, can make themes fragile when properties are added or removed, and can leave dead theme properties behind when components stop using them.

Theming is planned for modern controls but will not immediately be available. At the time of writing this article, the modern controls offer virtually no customization around colors, and use a blue accent where color is normally applied to the theme.

You can provide feedback directly in the studio or in the community forum, which helps the product team know which features to prioritize most.

The Theme property that is used in Creator Kit components will not be available in modern controls.

If you have deeply customized the styles of components from previous versions, it will require effort to bring them forward. However, the new design for theming is intended to make it much easier to implement and manage themes across Power Apps, once available in the product.

Props changes.

In modern controls, component props differ from Creator Kit (or custom page) versions. You will need to update code to use new components.

Below is a comparison table of properties (or notable implementation differences) available in modern controls that are not available in Creator Kit components, and vice versa.

⚠ **Important!** As modern controls are still in preview, property names are subject to change.

Component	Modern control properties	Creator Kit control properties
TabList (aka Pivot)	• Fields ○ Selecting displayed columns is now mandatory. ○ Specific schema structure [e.g., ItemKey, ItemDisplayName] is no longer required ○ First field selected will be rendered • Render size has more options • Alignment • DefaultSelectedItems ○ Tip: Use a collection or data source that can be referenced as the TabList's Items, then use LookUp() to that same reference	• Render type (maps to "Render size") • Selected key (maps to "DefaultSelectedItems") • Input event • Items properties ○ ItemIconName ○ ItemCount • Theme • Tooltip • Tab index
Spinner (Creator Kit Spinner)	• Appearance • "Label position" enum values (options have different names, but same positions) • Spinner size has more options	• Background color • Spinner alignment (maps to "Label position") • Theme • Tooltip • Tab index
ProgressBar (aka ProgressIndicator)	• Thickness • Shape • Indeterminate • Color • Value • Max	• Label • Description • Percent complete (maps to "Value" / "Max") • Type of indicator (maps to "Indeterminate") • Hide progress bar • Bar height (maps to "Thickness") • Theme • Tooltip • Tab index
Horizontal and Vertical container (now has Elevation props)	• Border radius • Drop shadow	• Depth • Hover depth • Dark overlay padding • Hover fill color

Build updates.

You may need to adjust your solution dependencies, which will no longer support the old version of the component from Creator Kit. If you are using applications that no longer use the Creator Kit components, then do not include the Creator Kit solution in the environment you deploy the application to.

Stages of migration

If you have a large codebase, it will be helpful to break up your plan into stages.

1. **Get familiar with modern controls.**

   Read the documentation and try using several components.

2. **Assess your application.**

   Analyze your codebase to measure how many places you use Fluent UI, how much custom styling you do, how much complex data binding you do, and what architectural abstractions you have in place.

3. **Decide when to migrate (milestones).**

   Consider how you might flight new components to vet them in your application.

   Decide if you want to:

   - migrate your entire codebase at once,
   - migrate a single component type (i.e. horizontal migration), or
   - migrate all components in a single experience or page (i.e. vertical migration).

   Plan out some milestones where you can verify all your tests pass, you meet your performance benchmarks, and you can confidently ship using new components.

4. **Migrate some experiences/components.**

   We recommend migrating Spinner as your first component. It tends to be used across an application, has a straightforward interface, and provides performance improvements.

   Record how much effort each replacement requires and use this information when planning out future component migrations. Monitor bundle size, build time, and render performance as you adopt modern controls.

5. **Build new experiences in modern controls.**

   If you are experimenting with new experiences and features to your product, we strongly recommend building them with modern controls.

Once modern controls are generally available, you should opt to build new experiences with modern controls rather than incurring additional migration debt (if you have decided modern controls property set will suit your needs).

6. **Incrementally migrate core experiences.**

   Plan out a rinse-and-repeat plan to migrate your core experiences to modern controls. From previous stages, you should feel confident handling any migration challenges.

7. **Migrate the remaining long tail.**

   Build a list of the remaining Creator Kit components and check against the roadmap to see when there will be modern control replacements.

   Consider building a placeholder component using the Creator Kit components that align with the new styling.

Migrating your first component

We'll use Spinner as an example.

1. Locate all the usages of the Creator Kit Spinner control you want to replace.
2. At each place you use the Creator Kit Spinner, add a new modern Spinner next to it.
3. Replace the prop values from the old Spinner in the new Spinner (use the table above to map properties that have different names but similar functionality)
4. Replace any reference to the Spinner if applicable.
5. Delete the old Spinner control, and resolve any errors using the App Checker.

You can reference each component's documentation and compare it to the new one, to determine which props are carried forward, were changed, or were deprecated/removed.