EcomLabs

MIGRATION GUIDE

Polaris 2.0: changes and migration checklist

What the Polaris 2.0 release candidate brought according to the rc.1 changelog, which URL and types to use now that it is stable, how to test it in a branch, what to check in each admin design, and the checklist to migrate your components.

FOR	Shopify apps embedded in the admin (App Home) on Polaris web components or Polaris React
TESTED WITH	Node 22.22.3 · npm 11.11.1 · offline Playwright Chromium · Polaris validator (shopify-plugin) · Windows 11
UPDATED	November 11, 2026
DOCS	shopify.dev/docs/apps/build/app-home/polaris2
ARTICLE	ecomlabs.dev/g/polaris-en
AUTHOR	Luis Morales · @luismoralesdev

Contents

Before you start	Where Polaris 2.0 stands today
Step 01	What changed in rc.1
Step 02	Which URL and which types
Step 03	Test it in a branch
Step 04	What to look at in each store
Step 05	Migrate what Polaris does not control
Final	Migration checklist

BEFORE YOU START

Where Polaris 2.0 stands today

Polaris 2.0 is the new major version of Polaris web components (`s-page`, `s-section`, `s-button`…), the UI library for apps embedded in the Shopify admin (App Home). It loads as a script from Shopify's CDN; npm only carries the types, `@shopify/polaris-types`. It is not Polaris React: `@shopify/polaris` is marked deprecated on npm and does not get the new style.

- **September 24, 2026:** release candidate at `polaris-2.0-rc.js`, with both admin styles (previous and new) in one bundle.
- **October 1:** rc.1, with the changes in Step 01. npm then got rc.2 (Oct 5) and rc.3 (Oct 9) of the types.
- **October 22:** Polaris 2.0 stable at `polaris-2.js` and `polaris-2.0.js`.

Moving to 2.0 is always explicit: `polaris.js` and `polaris-1.js` stay on 1.1. Admin UI extensions and App Home UI extensions update on their own. Built for Shopify: until May 1, 2027 for your App Home to match the admin's style.

STEP 01

What changed in rc.1

Every change in "What's new in Polaris 2.0.0-rc.1" (shopify.dev/changelog, October 1, 2026), with each component's tag:

COMPONENT	CHANGE
`s-table`	Matches the updated admin design.
`s-tooltip`	Can go in the `supplemental` slot of `s-section`.
`s-section`	Equal-height cards in the same `s-grid`; on narrow screens the primary action moves below the card; tighter headings; default text color for aside headings.
`s-select`	Same height as `s-text-field` and other inputs.
`s-badge`	Pill shape on mobile.
`s-banner`	Better title contrast.
`s-color-picker`	Saturation-well shadow fixed at the corners.

LAYOUT AND RELIABILITY	CHANGE
`s-menu`	Icons use the secondary icon color at rest.
`s-page`	Aside section styling fixed; secondary actions (on `s-section` too) keep their disclosure icon when the menu or label is added after mount.
Nested `s-section`	The first one no longer sits lower than other first children.
Slotted actions	No longer disappear when markup is injected into an already-connected element.
`globalThis.polaris`	State is kept when a page loads the bundle more than once.
`s-modal`	Actions fire a bubbling click even with `commandFor`.
`s-text-field`	Accessory clicks reach React `onClick` handlers.
Performance	Faster mounting, especially without React, and fewer re-renders on `s-choice-list`, `s-menu`, `s-modal`, `s-select` and pickers.

STEP 02

Which URL and which types

Measured on November 11, 2026: each file downloaded from the CDN and loaded offline in a local Chromium.

SCRIPT	SERVES	USE IT FOR
`polaris-2.js`	2.0.0	Production: follows each stable 2.x. Types `^2.0.0`.
`polaris-2.0.js`	2.0.0	A pinned release. Types `~2.0.0`.
`polaris-2.0-rc.js`	2.0.0-rc.3	Frozen now: promotion did not move it to stable.
`polaris-1.js` and `polaris.js`	1.1.0	Polaris 1. `polaris.js` never moves to 2.x.

For a future RC (say `polaris-2.1-rc.js`, a 404 today), use types `^2.1.0-rc.0` or the exact version: a normal range like `^2.0.0` does not accept pre-releases (checked with the semver module in npm 11.11.1).

STEP 03

Test it in a branch

```
git switch -c polaris-2
npm install --save-dev @shopify/polaris-types@^2.0.0
shopify store create dev --name polaris-2-qa --feature-preview new_admin_design
shopify app dev
```

The store with `new_admin_design` shows the new design; use another one without the preview for the previous design. If your app comes from the React Router template, set `polarisUrl` in both places, to the same value (needs `@shopify/shopify-app-react-router` 2.1.0 or later; from 4.0.0 on, the default is already `polaris-2.js`):

```
// app/routes/app.tsx
<AppProvider
  apiKey={apiKey}
  polarisUrl="https://cdn.shopify.com/shopifycloud/polaris-2.js"
>
  <Outlet />
</AppProvider>

// app/shopify.server.ts
const shopify = shopifyApp({
  // ...
  polarisUrl: 'https://cdn.shopify.com/shopifycloud/polaris-2.js',
});
```

If you write your own tag, change its URL to `polaris-2.js` and keep it after `app-bridge.js`, as the only Polaris tag. To check which bundle loaded, look at the network panel: one request to `polaris-*.js`. In the console, `globalThis.polaris?.version` returned `'2.0.0'` in our test (undocumented property: debugging only).

STEP 04

What to look at in each store

Open your app's screens in the store with the new design and in the other one, and check what rc.1 touched. The article has a test page with all of these components, validated against types 1.0.1 and 2.0.0 and rendered with no errors on 1.1, the RC and 2.0.

- `s-table` in the updated admin design.
- `s-select` next to `s-text-field`: same height.
- Several `s-section` elements in one `s-grid`: equal-height cards.
- An `s-section` with a primary action on a narrow screen: the action moves below the card.
- `s-badge` on mobile (pill) and the `s-banner` title.
- Tooltips in the `supplemental` slot of `s-section`, secondary-action menus, and modals with `commandFor`.

STEP 05

Migrate what Polaris does not control

- **Polaris React:** whatever `@shopify/polaris` renders keeps the old style. Migrate page by page: the shell (`Page`, `Card`, `Layout`), then `Badge`, `Banner`, `Text`, `Icon`; buttons, menus and modals; forms; tables and filters. There is no `s-card`: use `s-section`.
- **Web component rules:** camelCase props, kebab-case slots, `url` becomes `href`, overlays open with `commandFor`. Controlled fields from React need React 19.
- **CSS:** `.Polaris-*` selectors cannot reach into the shadow DOM; CSS with values copied from `@shopify/polaris-tokens` stays on the old palette.
- **Fonts:** if you hardcode Inter, remove it; Polaris 2.0 loads ShopifyInter when the admin renders the new design.
- **Fixed-bottom elements:** `bottom: var(--shopify-safe-area-inset-bottom, 0px);` so the Sidekick bar does not cover them.

- **CSP:** `https://cdn.shopify.com` in `script-src`.

> **CAUTION**
>
> We did not run `shopify store create dev` or `shopify app dev`: they need a Shopify login. The local test ran outside the admin, without App Bridge: it confirms which version loads and that there are no errors, not how the app looks inside the admin. Loading `polaris-1.js` and `polaris-2.js` together showed no error and the second one won; the docs say two bundles make the app fail, so do not count on a warning.

FINAL

Migration checklist

- [ ] You work in a branch; production stays on Polaris 1 until the merge.
- [ ] The URL is `polaris-2.js` (or a pinned `polaris-2.0.js`), not `polaris-2.0-rc.js`.
- [ ] In template apps, `polarisUrl` matches on `AppProvider` and `shopifyApp`.
- [ ] There is a single Polaris tag, after `app-bridge.js`, and a single request in the network panel.
- [ ] `@shopify/polaris-types` on `^2.0.0`, matching the URL.
- [ ] Whatever is left on `@shopify/polaris` is on a dated list, starting with Page, Section and Button.
- [ ] You reviewed CSS built on Polaris tokens, hardcoded fonts and hand-built components.
- [ ] Fixed-bottom elements use `--shopify-safe-area-inset-bottom`.
- [ ] You walked the main flows, empty and error states in a store with `new_admin_design` and in one without it.
- [ ] If you bump `@shopify/shopify-app-react-router` to 4.x, you know it switches Polaris to 2.