

Getting Started with Interface Customizations in Primo VE

Setting Up a Local Environment for Testing

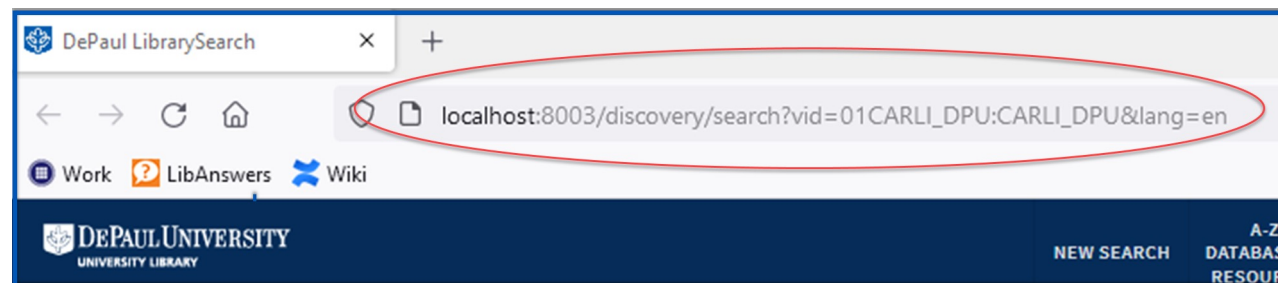
Christine McClure

Digital Services Coordinator
DePaul University Library

What It Means

- Run PrimoVE on your desktop—*using your library's data**—to test out changes to the look and feel without affecting your live site.

- CSS colors and styles
- Home page text
- Javascript customizations



*Authentication method not included. (You'll need to create a test account manually to be able log in)

Why Not Use Primo Studio?

- No historical information
- Bare CSS only, no extensions/frameworks like [Sass](#)
- No review workflow other than creating multiple views
- *Javascript cannot be modified!*



Setting It Up

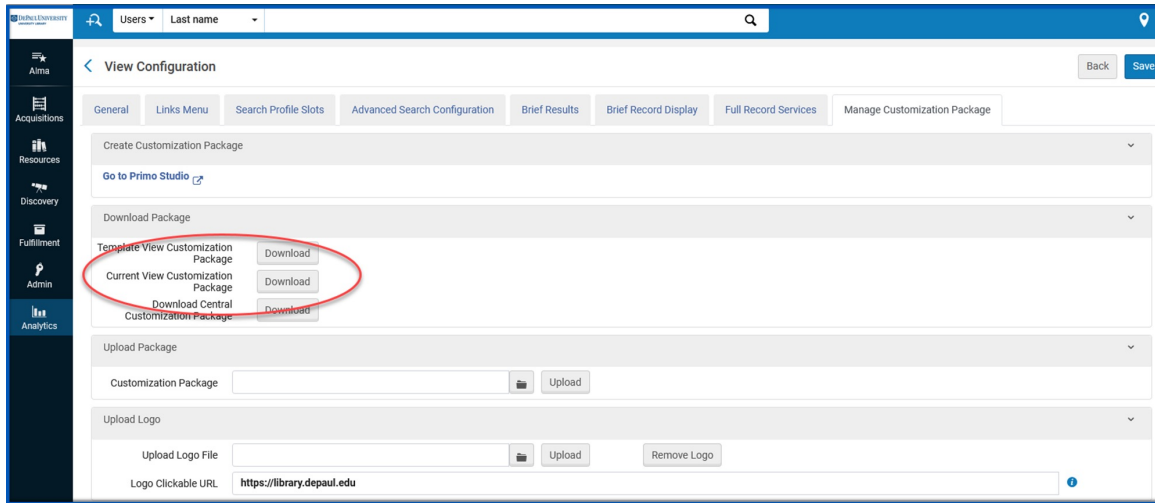
1. Use a Node.js version manager for best results ([Windows](#) instructions)
2. Install the Ex Libris Code Environment:
<https://github.com/ExLibrisGroup/primo-explore-devenv>
3. Download your view(s), extract the folder and copy it to the `.\primo-explore\custom` folder
4. Run the view with this command:

```
gulp run --view <view name> --ve --useScss
```

The “-ve” command is missing from the Ex Libris instructions and will not work without it. The “useScss” command is needed if you want to use Sass commands

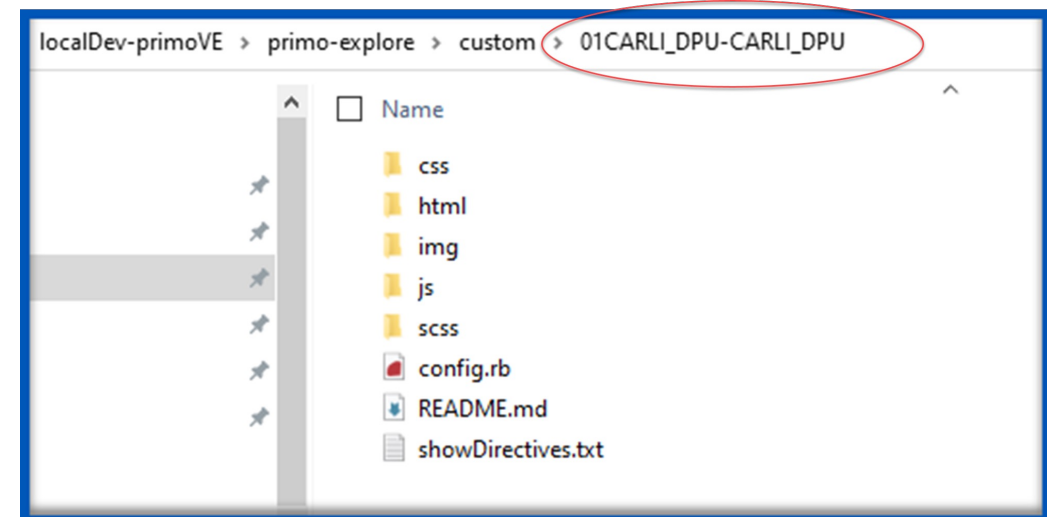


Step 3 Expanded



Download the view

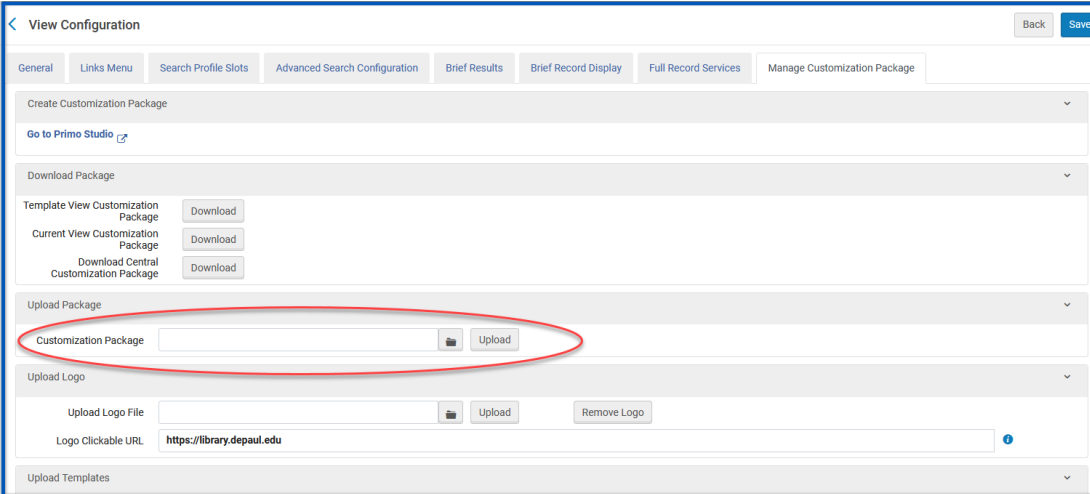
Extract and copy the folder over



Publish Your Changes

1. When finished making changes, create a package with the following command and upload it to the Alma Views module

```
gulp create-package --view <view name> --ve --useScss
```



The screenshot displays the 'View Configuration' interface with several tabs: General, Links Menu, Search Profile Slots, Advanced Search Configuration, Brief Results, Brief Record Display, Full Record Services, and Manage Customization Package. The 'Manage Customization Package' tab is active. It contains sections for 'Create Customization Package' (with a 'Go to Primo Studio' link), 'Download Package' (with 'Download' buttons for Template View, Current View, and Central Customization Packages), and 'Upload Package'. The 'Upload Package' section features a file input field and an 'Upload' button, which are circled in red. Below this is the 'Upload Logo' section with fields for 'Upload Logo File' and 'Logo Clickable URL' (containing 'https://library.depaul.edu').



Caveats

1. The most difficult part is getting the environment set up initially.
2. You can't "go backwards" and go back to using Primo Studio
3. Things like labels and facets must be changed in Alma (database)



Example Workflow

1. Copy production view to a new view
2. Make changes and test
3. Package up new view and publish to show changes



Questions?

Christine McClure

c.mcclure@depaul.edu

(773) 325-7829

Primo instance:

https://i-share-dpu.primo.exlibrisgroup.com/discovery/search?vid=01CARLI_DPU:CARLI_DPU&lang=en

Code repository:

<https://github.com/dpuwebservices/localDev-primoVE>

