PatternFly Community Meeting

March 28th, 2023

www.patternfly.org/v4

PatternFly community meeting privacy notice

- Please note: meetings are recorded and made publicly available on YouTube.

- You are welcome to keep your camera off and/or change your display name if you would like to maintain more privacy while on the call.

- Recordings are posted without any attendee or presenter names in the video descriptions.

- Presenters will be introduced using their display name on the call. You are welcome to verbally further identify yourself when presenting if you are comfortable.

- If any attendee wishes the video to be removed from public availability from YouTube for any reason, at any time, please let me know and it will be made private or removed immediately.

Agenda

Topics

- Community topics - Nicole
- Latest release notes links - Nicole
- Patternfly Yew - Jens
- Extension of the month! - Catalog view - Jessie
- PF Logo & Homepage design update - Lucia & Michael
- Preventing style collisions in v5 - Matt

Q&A

Community updates

- Update on [v5 major release timeline](#)
  - We are all hands on deck for PF v5, tested in May and released in June 2023

  - We are about to begin our second round of v5 alpha testing
    - Updated a Koku, Keycloak, OCP, and OCM to use the alpha test candidate versions of PF + others
    - Run the codemods on the code base
    - Evaluate level of effort for making changes that codemods couldn't automatically fix.
  - A summary of our process and results can be found [in this document](#).
  - Contact Nicole (nthoen@redhat.com) if you're interested in aiding with testing in your project.

Latest Release Notes – Milestone 2023.02

- Milestone 2023.02

    - [React release notes](#)

** This was a patch release of PatternFly v4.

We are still currently working towards a major release of the next major version of PatternFly and you can view a summary of the high impact/high priority issues on [this Project Board](#).

PatternFly Yew

PatternFly Yew

- **Yew:** "A framework for creating reliable and efficient web applications."
- Using **Rust**, compiling to **WebAssembly**
- Yew uses concepts similar to ReactJS in the JavaScript world
- **PatternFly Yew** provides Yew components built on top of PatternFly

PatternFly Yew

PatternFly Yew

```rust
#[function_component(Counter)]
pub fn counter() -> Html {
    let counter = use_state_eq(|| 0);

    let onclick = {
        let counter = counter.clone();
        Callback::from(move |_| counter.set(*counter + 1))
    };

    let title = html!("Clicks");

    html! (
        <Gallery gutter=true>
            <Card {title}>
                <p>{ *counter }</p>
                <Button label="Add One"
                    align={Align::Start} icon={Icon::PlusCircle}
                    variant={ButtonVariant::Link}
                    {onclick}
                />
            </Card>
        </Gallery>
    )
}
```
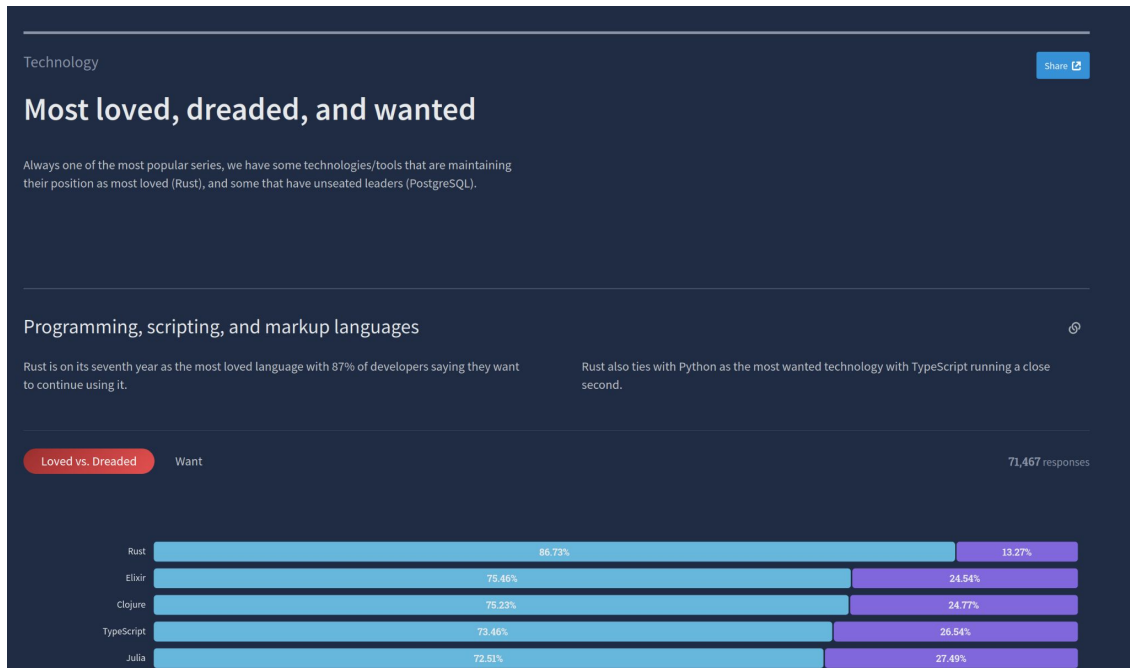
Clicks

0

⊕ Add One
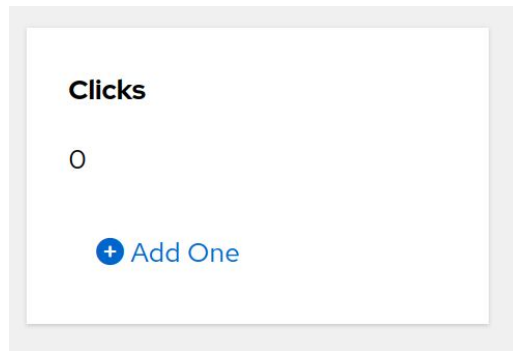
PatternFly Yew – Top 3: Why Rust?

- Find bugs during compile time, not run time
  - Strict type system
  - Lifetimes/borrow checker
  - And much more
- WebAssembly is pretty fast
  - No need to "understand" JavaScript
  - Just execute instructions
- …

PatternFly Yew – Top 3: Why Rust?

Technology

Most loved, dreaded, and wanted

Always one of the most popular series, we have some technologies/tools that are maintaining their position as most loved (Rust), and some that have unseated leaders (PostgreSQL).

Share 🔗

Programming, scripting, and markup languages

Rust is on its seventh year as the most loved language with 87% of developers saying they want to continue using it.

Rust also ties with Python as the most wanted technology with TypeScript running a close second.

| Loved vs. Dreaded | Want | 71,467 responses |

Rust	86.73%	13.27%
Elixir	75.46%	24.54%
Clojure	75.23%	24.77%
TypeScript	73.46%	26.54%
Julia	72.51%	27.49%

Rust is on its **seventh year** as the most loved language with 87% of developers saying they want to continue using it.

PatternFly Yew – Rust Web Toolchain[1]

Rust
Code

cargo →

WASM
Binary

(S)CSS,
Assets,
…
JS(!)

trunk →

Distribution
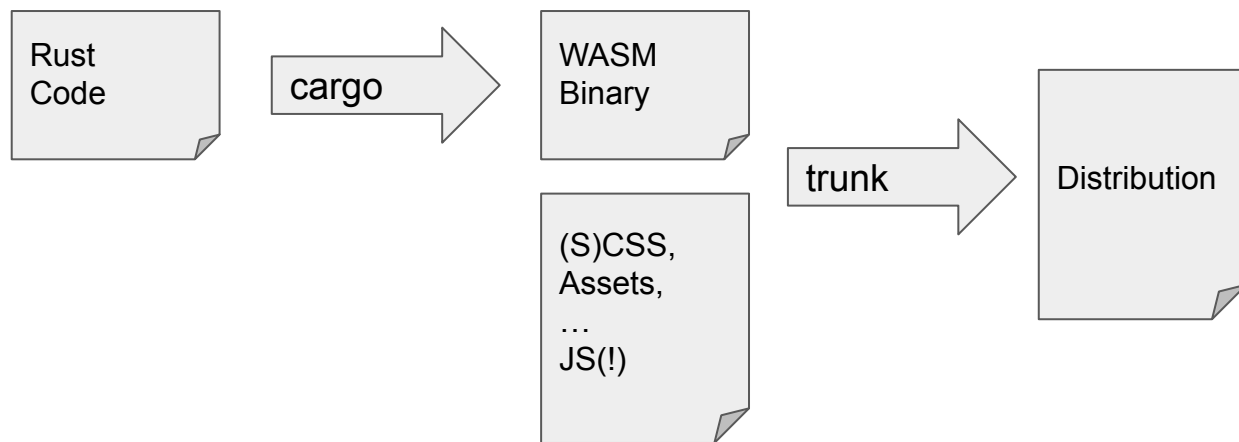
[1] For client-side rendering, there also is server-side rendering

PatternFly Yew – Rust Web Architecture[1]

Rust/WASM Application

DOM

WASM Code ⟷ JS Code

WASM VM   JS VM

Browser / Web APIs
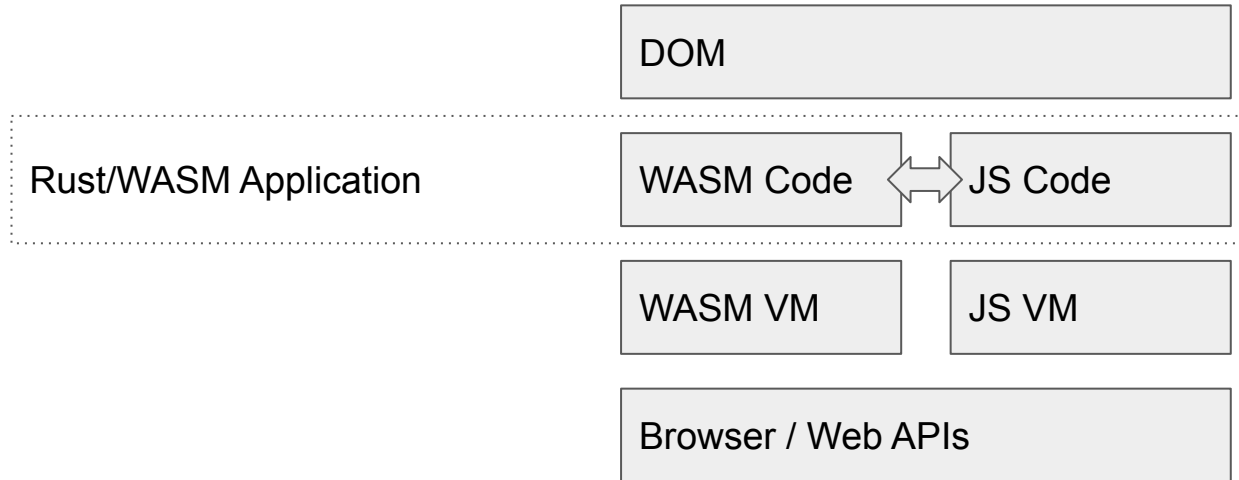
[1] For client-side rendering, there also is server-side rendering

PatternFly Yew

- History
  - Started ~2.5 years ago (Oct 2020)
  - Originating from two Red Hat sponsored upstream projects (Drogue IoT, Seedwing)
- State
  - A good amount of components are available
  - Some features and components are missing
  - Implemented on an "as needed" basis
  - Has hand full of external contributors
- Future?
  - More components & missing futures
  - Keep up with changes

Demo

PatternFly Yew

- PatternFly Yew
  - GitHub Repository: https://github.com/ctron/patternfly-yew
  - Quickstart/Demo: https://ctron.github.io/patternfly-yew-quickstart
- Other examples
  - https://playground.seedwing.io/
  - https://sandbox.drogue.cloud/
- Rust & WebAssembly
  - https://rustwasm.github.io/docs/book/
- Yew – https://yew.rs/
- Trunk – https://trunkrs.dev/

Extension of the Month - Catalog view

Catalog View

- Allows users to create catalog (often used alongside Quick Starts)

Logo and homepage design updates

Why are we refreshing the PF logo?

PatternFly has been using the same logo since its creation in 2014.

*Read an article **here** about the creation of the original logo.*

PatternFly has been using the same logo since its creation in 2014, back when patternfly.org was hosted on WordPress and the component library was built with Bootstrap. **Since 2014, PF has undergone some huge updates to become more scalable, modern, and accessible.** With the release of PF4 in 2019, and with the ongoing design vision work, our PF design team has been looking for ways to keep PatternFly's design language modern, approachable, ahead of the curve. It's only natural that our (quickly growing) brand also reflects PF's evolving style.
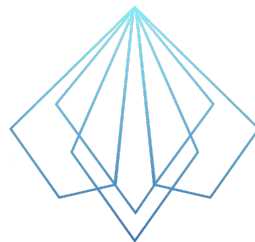
Why are we refreshing the PF logo?

Painpoints of current logo

- The level of detail and thin lines in our current logo doesn't display well at small sizes

- It doesn't match well with the updated Red Hat brand

- More complicated logos are falling behind current design trends, which tend to favor a more simplistic/minimal approach

- The current wordmark in our logo is using Overpass

- The gradient in the logo doesn't match our current blue color palette
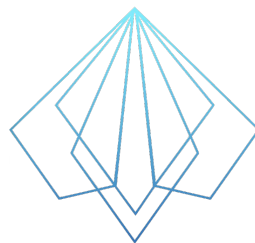
Why are we refreshing the PF logo?
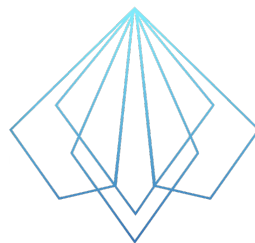
Benefits of updating our new logo

- **Fresh, modern look**
  Modernize our brand, keep up to date with design trends

- **Reflect who we are**
  Align with the updated PF look and feel

- **Better usability and scalability**
  allow for better use at smaller scales (favicon, social media icons) and easier printing (tshirts, printed merch)

Logo iterations

Logo iterations

Flying into the future

It's all just circles and squares when you think about it, man

PatternFly

PatternFly

PatternFly

PatternFly

PatternFly

PatternFly

Secondary line example

PatternFly
Red Hat Conventions

PatternFly
Red Hat Conventions

Logo iterations

Flying into the future

PatternFly

PatternFly

PatternFly

It's all just circles and squares when you think about it, man

PatternFly

PatternFly

PatternFly

Secondary line example

PatternFly
Red Hat Conventions

PatternFly
Red Hat Conventions

Logo iterations

Flying into the future

PatternFly

It's all just circles and squares when you think about it, man

PatternFly

Final Logo Options – Option A

Flying into the future

| | Transparency variant | Single color/outline variant | Gradient variant |

PatternFly

PatternFly

PatternFly
Red Hat Conventions
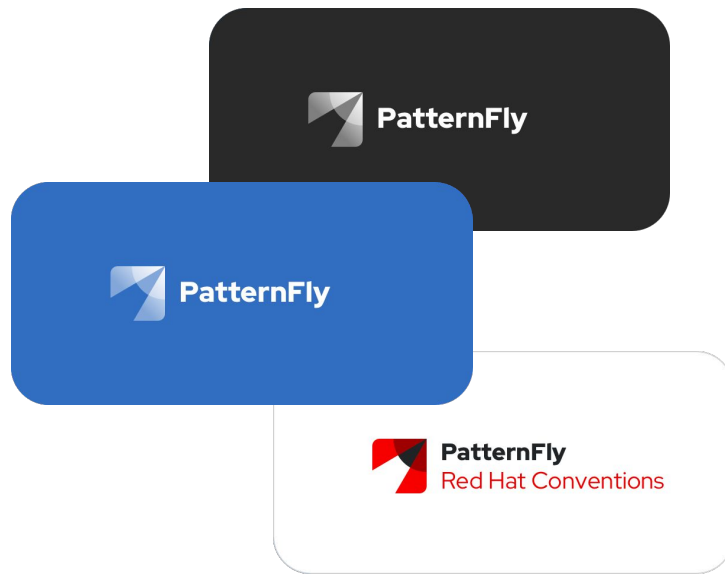
Final Logo Options – Option B

It's all just circles and squares when you think about it, man

Transparency variant

Single color/outline variant

Gradient variant

PatternFly

PatternFly

PatternFly
Red Hat Conventions
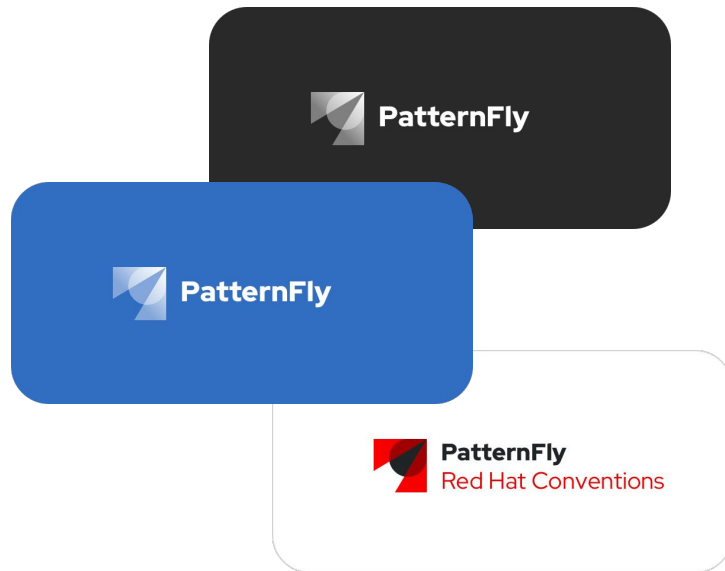
Now to refresh the pf.org homepage ✨

Goals for the new homepage

- Add in more useful content in order to increase engagement with returning users

- Update homepage visuals to reflect new PatternFly Penta theme

- Audit existing content, simplify, and identify content gaps
  Shout out to Erin, who has been our content extraordinaire for this effort!
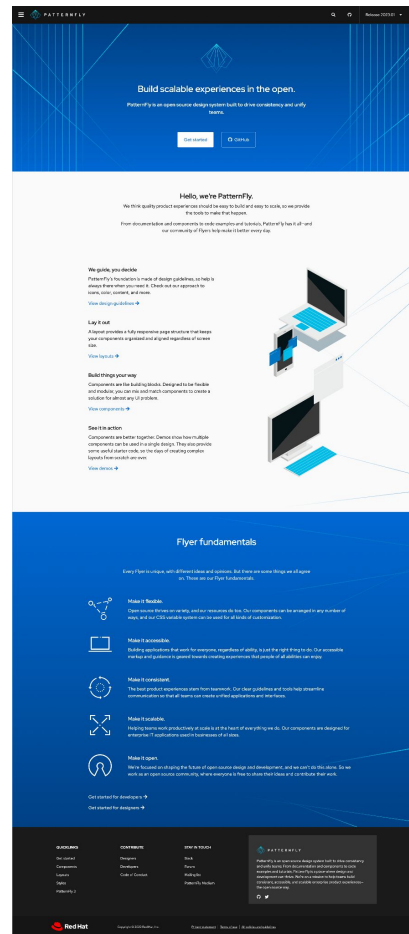
Homepage explorations
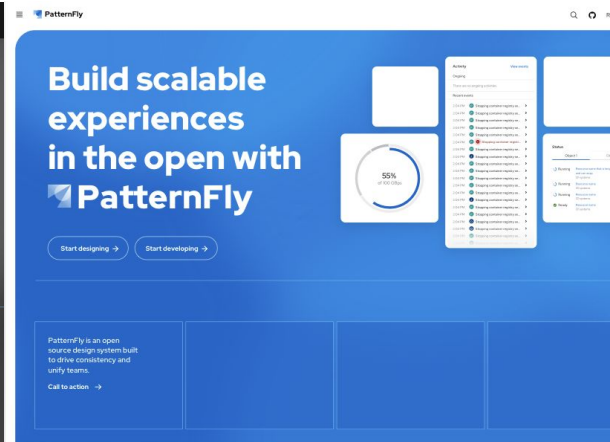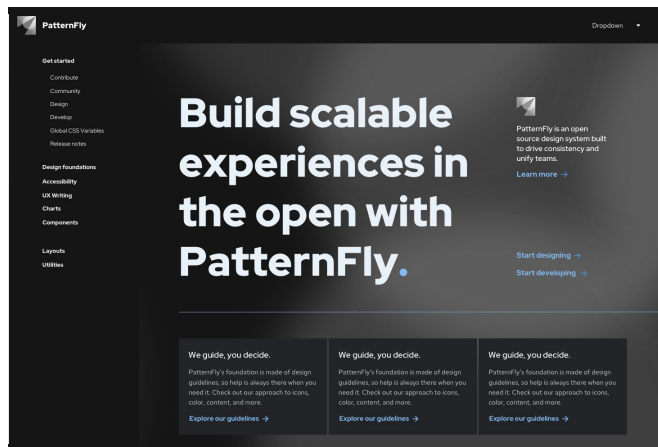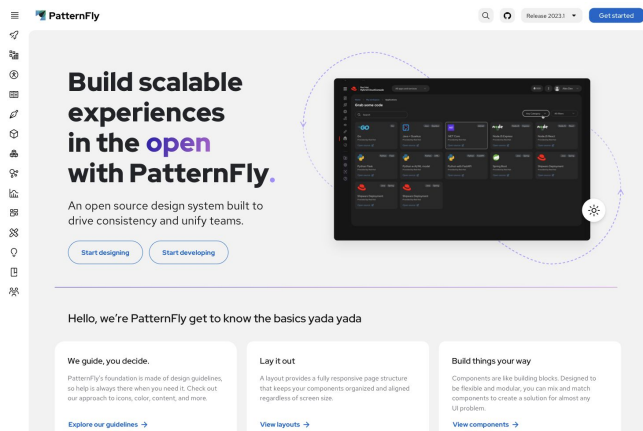
Homepage - updates

🚧 Copy not final

- Updated styles to PatternFly Penta theme
- Updated logo
- Updated tagline and body copy
- Replace "get started" CTA with separate "Start designing" and "Start developing" actions
- Display a component showcase in the header, with ability to toggle between themes
- New "Explore our Blog" section
- New "Stay informed" section
- Proposed move of "flyer fundamentals" to a community page on website
- Proposed new landing pages for design guidelines and layouts

Homepage – Final look/feel 🚧 Copy not final

Homepage - Final look/feel 🚧 Copy not final

PatternFly's **updated** visual identity

Functional artifacts   🚧 Work in Progress
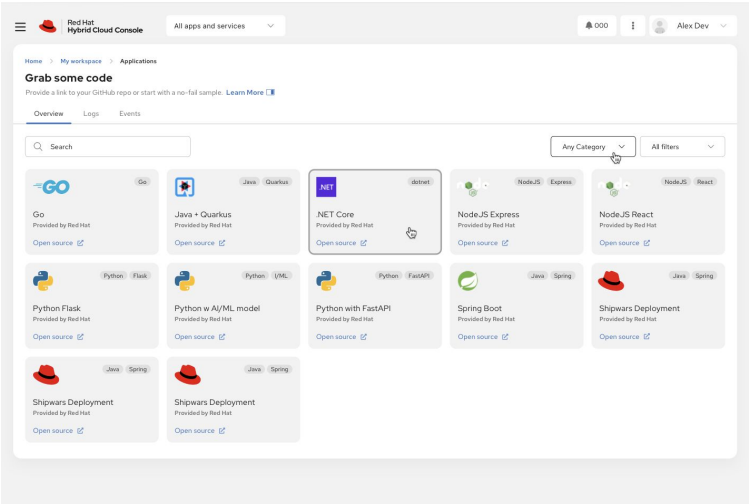
Marketing artifacts   🚧 Work in Progress
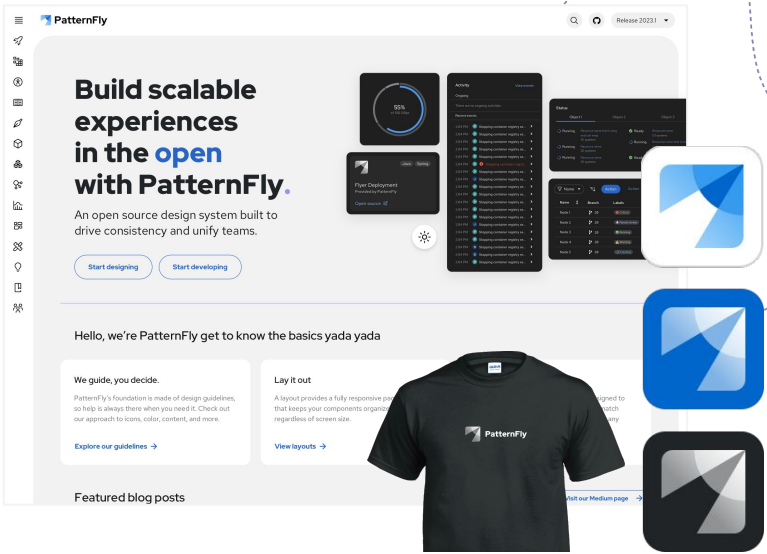
v4/v5 Style collision solution

v4/v5 Style Collisions

Objective
Why do we need a solution for this?

- Allow consumers to use multiple versions of PatternFly in a single app

- Avoid default styling collisions from one version to another

- Add customizability with this solution

v4/v5 Style Collisions - Solution

Version prefixing

Solution: Set default variables for namespace and version

```
$pf-v5-global--namespace: 'pf-' !default;
```

```
$pf-v5-global--version: 'v5-' !default;
```

```
#{$pf-prefix} : pf-v5-
```

v4/v5 Style Collisions - Configuration

```
$card:

#{$pf-prefix}c-card;



Selector: #{$card}
Variable: --#{$card}--Color
```

Selector

`.pf-v5-c-card`

Variable

`--pf-v5-c-card--Color`

v4/v5 Style Collisions - Benefits

Benefits of namespacing selectors and variables

- Style inheritance is no longer an issue

- Avoid custom styling collisions from one version to another

- The prefix is customizable by the consumer

v4/v5 Style Collisions - Benefits

```
$pf-global: #{$pf-prefix} + '-global';


-#{$card--Color}: var(--#{$pf-global}--Color-100);
```

Global variable

–pf-v5—global-Color-100 / #{$pf-v5-global-Color-100};

Card variable

–pf-v5-c-card-Color: var(–pf-v5—global-Color-100)

Q & A

Links

To join the PatternFly mailing list: https://listman.redhat.com/mailman/listinfo/patternfly

Import the PatternFly Community calendar: calendar

Community meeting coordinator contact: Nicole Thoen - nthoen@redhat.com

Next office hours: April 5th

Thank you!