



Evan Thibodeau and Patrick Yeo

How to Implement Your Design System in LXC Using Clay

Learn How to Integrate Your Existing Design System into Clay

What You're Going to See Today

- Quick Overview of Design Systems
- Two Ways to Apply Your Design System to Liferay
- How to Use the Clay Sass API to Apply Your Design System
- Demo
- Questions

What is a Design System?

- A set of rules for designers, developers, and others to follow
- Consists of reusable design and code
- Has tokens to change the look and feel of a system



Design System Pros

- Design and Code Consistency Across Your Products
- Transfer of Knowledge Between Departments, Designers, and Developers
- Documentation
- Reduce Bugs
- Minimizing Redundant Components

Design System Cons

- Initial Investment
- Inflexible
- Lack of Creativity

Too stiff? Loosen the reins.

“Rather than approaching design systems like architecture, we should approach it like gardening. A gardener plants seeds and tends to them. They know what they planted, but they don't know exactly what the garden is going to look like. They find out as it grows and intervene when they find weeds, or want to make changes.”

Shopify Designer José Torre

<https://www.figma.com/blog/the-future-of-design-systems-is-complicated/>

Two Ways to Apply Your Design System:

1

Replace Clay
with Your
Design System's CSS

2

Apply Your
Design System
to Clay's CSS

Two Ways to Apply Your Design System:

1 Replace Clay with Your Design System's CSS:

- It seems easier... at least to start with.
- You can use the existing markup from the design system.
 - This means you're on the hook for bugs, UX, and accessibility issues.
- The design system won't restyle any of Liferay's markup.

Two Ways to Apply Your Design System:

2 Apply Your Design System to Clay's CSS:

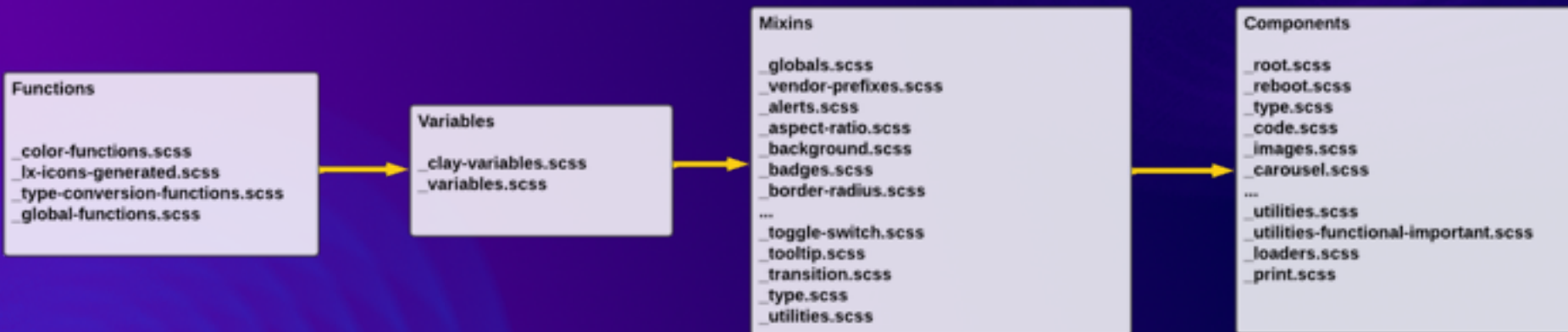
- Applies to all of Liferay.
- You can use Clay Components.
- We will handle bugs and accessibility issues in Clay for you.

How to Apply Your Design System to Clay

- No need to override existing CSS with higher specificity.
- No need to use `!important`.
- Using the Clay Sass API you can change the CSS that is printed out by Clay.

```
.how {  
  // ???  
}
```

Overview of File Structure



Clay CSS Variable Structure





Gray 100 theme

Carbon Design System

Version 11

Carbon Design System Tokens

Name	Value	Example
Blue 10	#edf5ff	
Blue 20	#d0e2ff	
Blue 30	#a6c8ff	
Blue 40	#7ba9ff	
Blue 50	#4589ff	
Blue 60	#0f62fe	
Blue 70	#0043ce	
Blue 80	#002d9c	
Blue 90	#001d6c	
Blue 100	#001141	

Token	Role	Value	Example
<code>\$link-primary</code>	Primary links; Ghost button	Blue 40 = #7ba9ff	
<code>\$link-primary-hover</code>	Hover state color for \$link-primary	Blue 30 = #a6c8ff	
<code>\$link-secondary</code>	Secondary link color for lower contrast backgrounds	Blue 30 = #a6c8ff	
<code>\$link-inverse</code>	Links on \$inverse-02 backgrounds	Blue 60 = #0f62fe	
<code>\$link-visited</code>	Visited links	Purple 40 = #be95ff	

Carbon Design System Tokens

https://github.com/pat270/devcon2023/blob/main/client-extensions/carbon-gray-100-v11/src/css/v11/_global-colors.scss

```
$cds-black: #000 !default;

$cds-black-100-hover: #212121 !default;

$cds-white: #fff !default;

$cds-gray-10: #f4f4f4 !default;
$cds-gray-20: #e0e0e0 !default;
$cds-gray-30: #c6c6c6 !default;
$cds-gray-40: #a8a8a8 !default;
$cds-gray-50: #8d8d8d !default;
$cds-gray-60: #6f6f6f !default;
$cds-gray-70: #525252 !default;
$cds-gray-80: #393939 !default;
$cds-gray-90: #262626 !default;
$cds-gray-100: #161616 !default;

$cds-gray-10-hover: #e8e8e8 !default;
$cds-gray-20-hover: #d1d1d1 !default;
$cds-gray-30-hover: #b5b5b5 !default;
$cds-gray-40-hover: #999999 !default;
$cds-gray-50-hover: #7a7a7a !default;
$cds-gray-60-hover: #5e5e5e !default;
$cds-gray-70-hover: #636363 !default;
```

Carbon Design System Tokens

https://github.com/pat270/devcon2023/blob/main/client-extensions/carbon-gray-100-v11/src/css/_clay_variables.scss

```
@import 'v11/_global-colors.scss';
@import 'v11/_globals.scss';
@import 'v11/_global-clay.scss';

@import 'v11/_alerts.scss';
@import 'v11/_badges.scss';
@import 'v11/_breadcrumbs.scss';
@import 'v11/_buttons.scss';
@import 'v11/_cards.scss';
@import 'v11/_clay-color.scss';
@import 'v11/_custom-forms.scss';
@import 'v11/_dropdowns.scss';
@import 'v11/_empty-state.scss';
@import 'v11/_forms.scss';
@import 'v11/_labels.scss';
@import 'v11/_links.scss';
@import 'v11/_list-group.scss';
@import 'v11/_loaders.scss';
@import 'v11/_management-bar.scss';
@import 'v11/_menubar.scss';
@import 'v11/_modals.scss';
@import 'v11/_navs.scss';
@import 'v11/_navbar.scss';
```


Carbon Design System Tokens

https://github.com/pat270/devcon2023/blob/main/client-extensions/carbon-gray-100-v11/src/css/v11/_buttons.scss

```
$btn-primary: () !default;
$btn-primary: map-deep-merge(
  (
    merge: false,
    background-color: $cds-button-primary,
    box-shadow: clay-enable-shadows($btn-box-s
    color: $cds-text-on-color,
    visited: (
      color: $cds-text-on-color,
    ),
    hover: (
      background-color: $cds-button-primary-
      background-image: c-unset,
      border-color: c-unset,
      color: $cds-text-on-color,
    ),
  ),
  $btn-primary
);
```

```
$btn-palette: () !default;
$btn-palette: map-deep-merge(
  (
    primary: $btn-primary,
    secondary: $btn-secondary,
    success: $btn-success,
    info: $btn-info,
    warning: $btn-warning,
    danger: $btn-danger,
    light: $btn-light,
    dark: $btn-dark,
    link: $btn-link,
  ),
  $btn-palette
);
```

```
// clay.css

.btn-primary {
  background-color: #0f62fe;
  color: #fff;
}

.btn-primary:visited {
  color: #fff;
}

.btn-primary:hover {
  background-color: #0050e6;
  color: #fff;
}
```

Carbon Design System Tokens

https://github.com/pat270/devcon2023/blob/main/client-extensions/carbon-gray-100-v11/src/css/v11/_buttons.scss

```
$btn-palette: () !default;
$btn-palette: map-deep-merge(
  (
    primary: $btn-primary,
    secondary: $btn-secondary,
    success: $btn-success,
    info: $btn-info,
    warning: $btn-warning,
    danger: $btn-danger,
    light: $btn-light,
    dark: $btn-dark,
    link: $btn-link,
  ),
  $btn-palette
);
```

```
$btn-palette: () !default;
$btn-palette: map-deep-merge(
  (
    '.btn-expressive': (
      min-width: 200px,
      padding-bottom: 20px,
    ),
    '.btn-ghost': (
      background-color: transparent,
      border-color: transparent,
    ),
  ),
  $btn-palette
);
```

Demo of Carbon in DXP

Questions



Links

- Client Extensions Documentation
 - <https://learn.liferay.com/w/dxp/building-applications/client-extensions/front-end-client-extensions/tutorials/using-a-css-client-extension>
- Github Repo
 - <https://github.com/pat270/devcon2023>
- Clay
 - <https://clayui.com/docs/css/scss.html>
- Clay Paver
 - <https://pat270.github.io/clay-paver-2/>



How was this session?

Please share your rating in
the event app. Thank you!