



ArcGIS API for JavaScript: Customizing Widgets

Matt Driscoll – @driskull

JC Franco – @arfnocode

2019 ESRI DEVELOPER SUMMIT
Palm Springs, CA

Agenda



Warning: Stranger Things ahead...

Agenda

- What can be customized
- Customization approaches with demos
- Q & A

Customizing Widgets



Customizing Widgets

- Theming
 - Changing styles: colors, sizing, font, etc.

Customizing Widgets

- Theming
 - Changing styles: colors, sizing, font, etc.
- Altering presentation of a widget
 - New view
 - Recreating a view

Customizing Widgets

- Theming
 - Changing styles: colors, sizing, font, etc.
- Altering presentation of a widget
 - New view
 - Recreating a view
- Extending
 - Overriding, adding functionality

Customization Approaches



Customization Approaches

- Authoring a theme



Customization Approaches

- Authoring a theme
- Recreating a view

Customization Approaches

- Authoring a theme
- Recreating a view
- Extending a view



Act I



Theming

Act I: Theming



Act I: Theming

Why Theme?



Act I: Theming

Why Theme?

- Match branding.
- Match the map.
- Contrast with the map.
- User-specific (e.g. bigger buttons)

Esri Themes

10 themes are provided out-of-the-box:

Using a theme requires only a slight update to the CSS path.

```
<link rel="stylesheet" href="https://js.arcgis.com/4.11/esri/themes/<theme-name>/main.css">
```



DEMO

TIME

Theme Switcher
Out-of-the-box themes

Theming Technology



We use



to create our CSS.

We use



to create our CSS.



nodejs.org | gruntjs.com



is a powerful scripting language for compiling CSS.



is a powerful scripting language for compiling CSS.

- It's modular.
- It's DRY.
- It makes theming easy.

Theming Steps



Theming Steps

1. Get our theme utility.
2. Use the utility.
3. Customize your theme.
4. Host your CSS file.

Theming Setup



Theming Setup

1. Clone the utility *jsapi-styles.git*
2. Run *npm install*
3. Edit *sass/my-theme/main.scss*.
4. See *dist/my-theme/main.css*.

Step 1

Clone the repo.

<https://github.com/jcfranco/jsapi-styles>

```
git clone https://github.com/jcfranco/jsapi-styles.git
```


Step 2

npm install



Step 2

npm install

- Installs the necessary bits.
- Creates a sample theme directory.
- Compiles the CSS from the SCSS.
- Spins up a preview in your default browser.

Step 3

Edit your theme.

sass/my-theme/main.scss

Step 3

Edit your theme.

[sass/my-theme/main.scss](#)

Optionally, edit your app.

[preview/index.html](#)

Step 4

Host your stylesheet and any relevant assets.

Link your stylesheet in your app.

```
<!-- In your app: -->  
<link href="path/to/your/theme/main.css" rel="stylesheet">
```

Goals

Theme Smart

- Avoid adding additional CSS selectors
- Instead, use Sass to your advantage

Theme Structure

Let's look at how the core theme is structured

- Color
- Size
- Type

Theme Structure

Let's look at how the core theme is structured

- Color : *color.scss*
- Size : *sizes.scss*
- Type : *type.scss*

Theme Structure

Default

```
// Inside base/_color.scss  
$background-color: #fff !default;
```

Any value assignment overrides the *!default* value.

```
// Inside sass/my-theme/main.scss  
$background-color: #1e0707;
```

Theme Structure

Default

```
// Inside base/_color.scss  
$background-color: #fff !default;
```

Any value assignment overrides the *!default* value.

```
// Inside sass/my-theme/main.scss  
$background-color: #1e0707;
```

But wait...there's more!

Theme Structure

Override the core color variables...

```
$font-color           : #3a5fe5;  
$interactive-font-color : #ff1515;  
$background-color      : #1e0707  
$button-color         : #ff1515;
```

Theme Structure

Override the core color variables...

```
$font-color           : #3a5fe5;  
$interactive-font-color : #ff1515;  
$background-color      : #1e0707  
$button-color          : #ff1515;
```

...then magic!



Magic



Using *\$button-color* we "automagically" set the hover color.

```
$button-color--hover: darken($button-color, 10%) !default;  
// ...etc
```

API Styling Guide

Act I: Lets make a theme

Stranger Things Theme



DISCLAIMER: *theme has low contrast*



DEMO

TIME

Custom Theme

Preview | Demo Steps

Act I: Theming Recap

- Use the utility for easy theming.
- Theme structure
 - Color
 - Size
 - Typography
- Use the core and override values.

End Act I

□ Act 1

Act II

□ Act 2

Views

Act II: Widget Composition

Widgets are composed of Views & ViewModels



Act II: Widget Composition

Widgets are composed of Views & ViewModels

- Logic is separate from presentation
- Reusable
- UI replacement
- Framework integration

Act II: TypeScript

- Widgets written in TypeScript (Typed JavaScript)

Act II: TypeScript

- Widgets written in TypeScript (Typed JavaScript)
- JS of the future, now

Act II: TypeScript

- Widgets written in TypeScript (Typed JavaScript)
- JS of the future, now
- IDE support
 - Visual Studio
 - WebStorm
 - Sublime
 - and more!

Act II: Views



Act II: Views

- Presentation of the Widget
- Uses ViewModel APIs to render the UI
- View-specific logic resides here
- Extends *esri/widgets/Widget*

Act II: Widget Class

esri/widgets/Widget



Act II: Widget Class

esri/widgets/Widget

- Provides lifecycle
- API consistency

Act II: Widget Lifecycle



Act II: Widget Lifecycle


- *constructor*




Act II: Widget Lifecycle

- *constructor*
- *postInitialize*

Act II: Widget Lifecycle

- *constructor*
 - *postInitialize*
 - *render*
- 

Act II: Widget Lifecycle

- *constructor*
 - *postInitialize*
 - *render*
 - *destroy*
- 
- An abstract geometric design in the bottom right corner, featuring various colored rectangles and lines in shades of blue, green, and purple, creating a sense of depth and movement.

Act II: *render*



Act II: *render*

- Defines UI
- Reacts to state
- Uses JSX
- VDOM

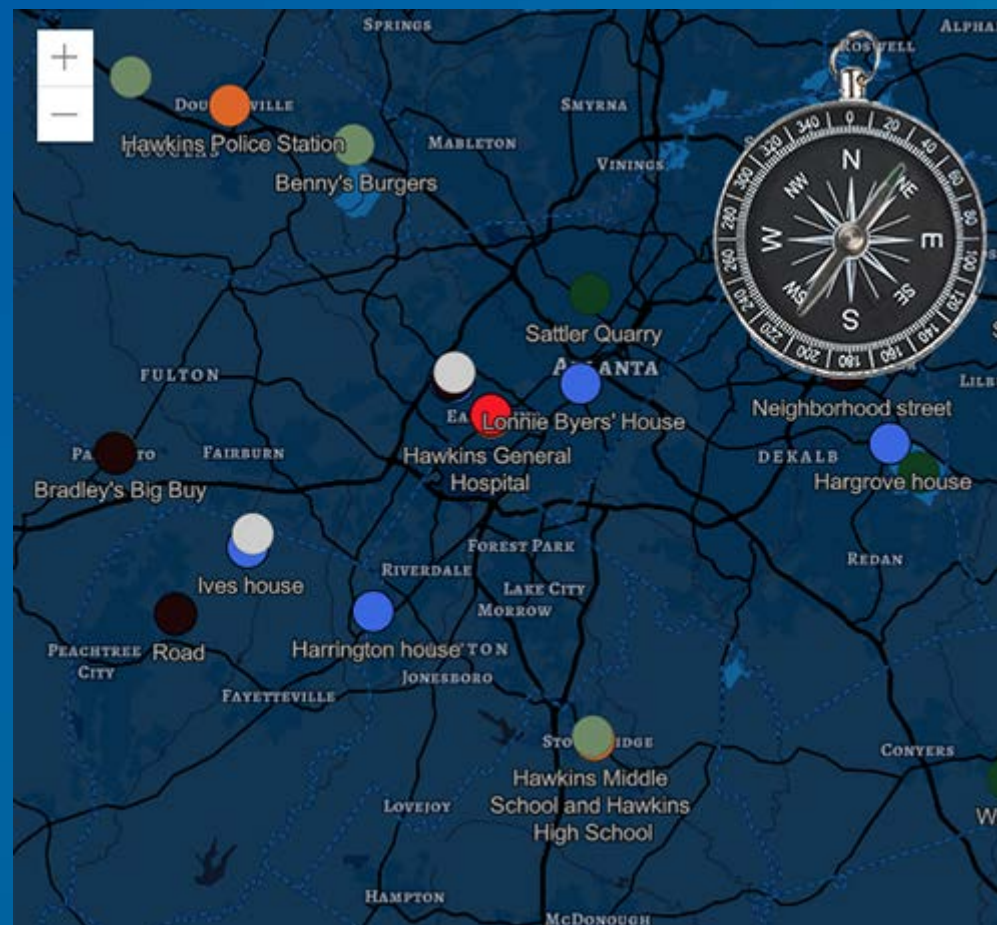
Act II: Working with Views

API Exploration

- Compass Doc
- Compass Sample

Act II: Recreating a view

Custom Stranger Things Compass



Act II: Compass Interface

CustomCompass widget: Same interface as the default Compass widget

```
interface CustomCompass {  
    view: View; //MapView | SceneView  
    viewModel: CompassViewModel;  
    reset(): void; // CompassViewModel.reset()  
}
```



DEMO

Recreating a view

[Demo Start](#) | [Demo Steps](#) | [Compass Doc](#)

TIME

Act II: Views Recap

What have we learned about Widget Views?



Act II: Views Recap

What have we learned about Widget Views?

- Face of the widget
- Present ViewModel logic
- ViewModel separation allows framework integration or custom views
- Downloadable on API docs

End Act II

□ Act 2

Act III

□ Act 3

Extending a View

Act III: Extending



Act III: Extending

- Why?



Act III: Extending

- Why?
 - Reusable

Act III: Extending

- Why?
 - Reusable
 - Same ecosystem

Act III: Extending

- Why?
 - Reusable
 - Same ecosystem
- How?

Act III: Extending

- Why?
 - Reusable
 - Same ecosystem
- How?
 - Leveraging *esri/widgets/Widget*

Act III: Extending


- Why?
 - Reusable
 - Same ecosystem
- How?
 - Leveraging *esri/widgets/Widget*
 - API Widgets



DEMO TIME

CustomCompass (Revisited)

Steps



DEMO

TIME

CustomBasemapGallery

Demo | Steps

Act III: Extending a View Recap



Act III: Extending a View Recap

- Reusable
 - View/ViewModel

Act III: Extending a View Recap

- Reusable
 - View/ViewModel
- Same ecosystem
 - No extra libraries

Act III: Extending a View Recap

- Reusable
 - View/ViewModel
- Same ecosystem
 - No extra libraries
- Extended existing widget
 - Lifecycle
 - TypeScript

End Act III

□ Act 3



The End?



Conclusion



Conclusion

- Authored a theme

Conclusion

- Authored a theme
- Recreated a view

Conclusion

- Authored a theme
- Recreated a view
- Extended a view

Recommended Sessions

- Intro to TypeScript Wed 1:30pm
- JavaScript for Geographers Wed 2:30 pm
- Using TypeScript with ArcGIS API for JavaScript Wed 2:30 pm
- Accessible Web Mapping Apps Thu 9am
- Building Your own Widget with ArcGIS API for JavaScript Thu 4pm

Additional Resources

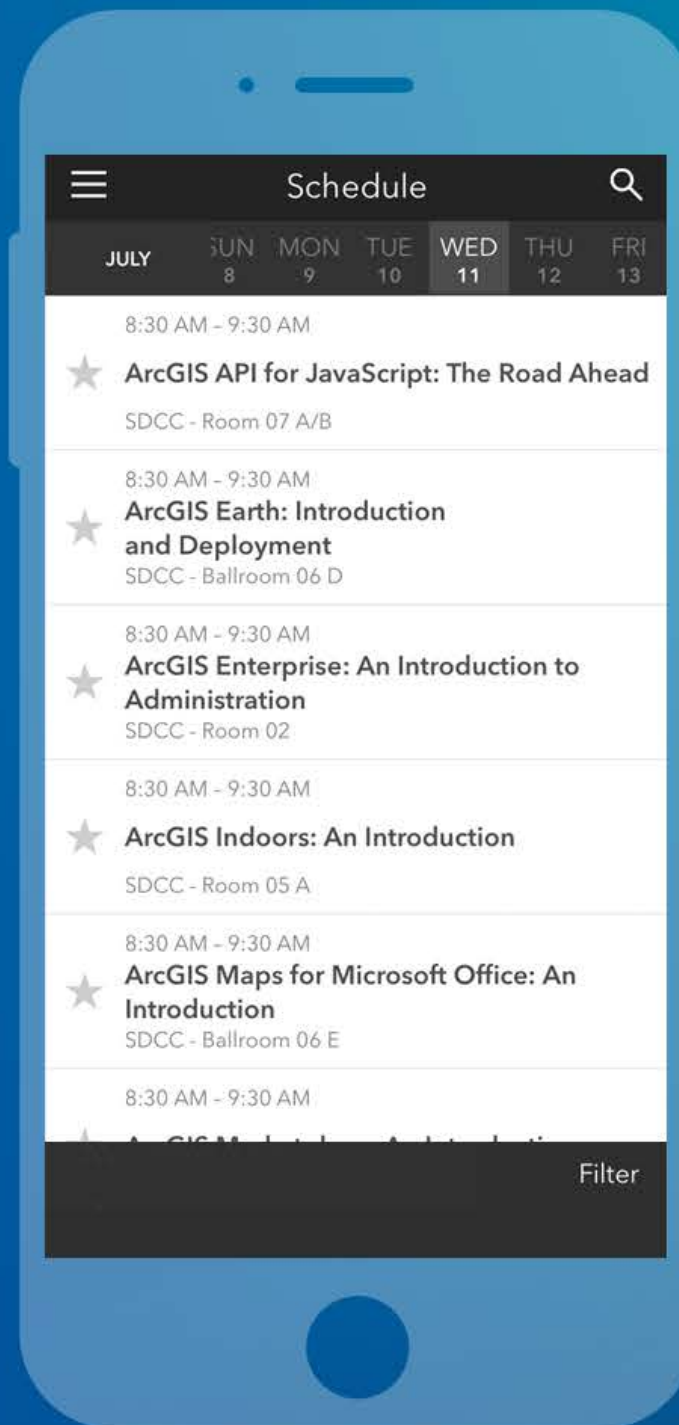
- Implementing Accessor
- Setting up TypeScript
- Widget Development
- JS API SDK

Please Take Our Survey on the App

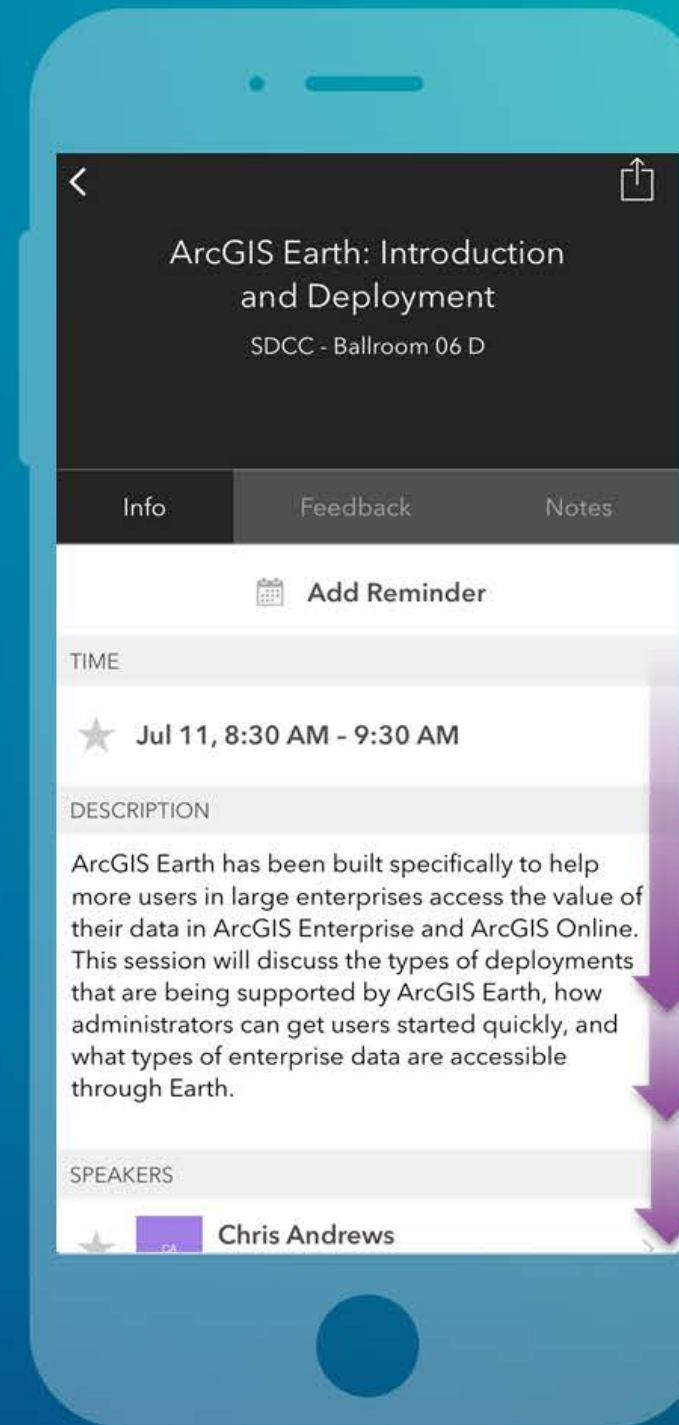
Download the Esri Events app and find your event



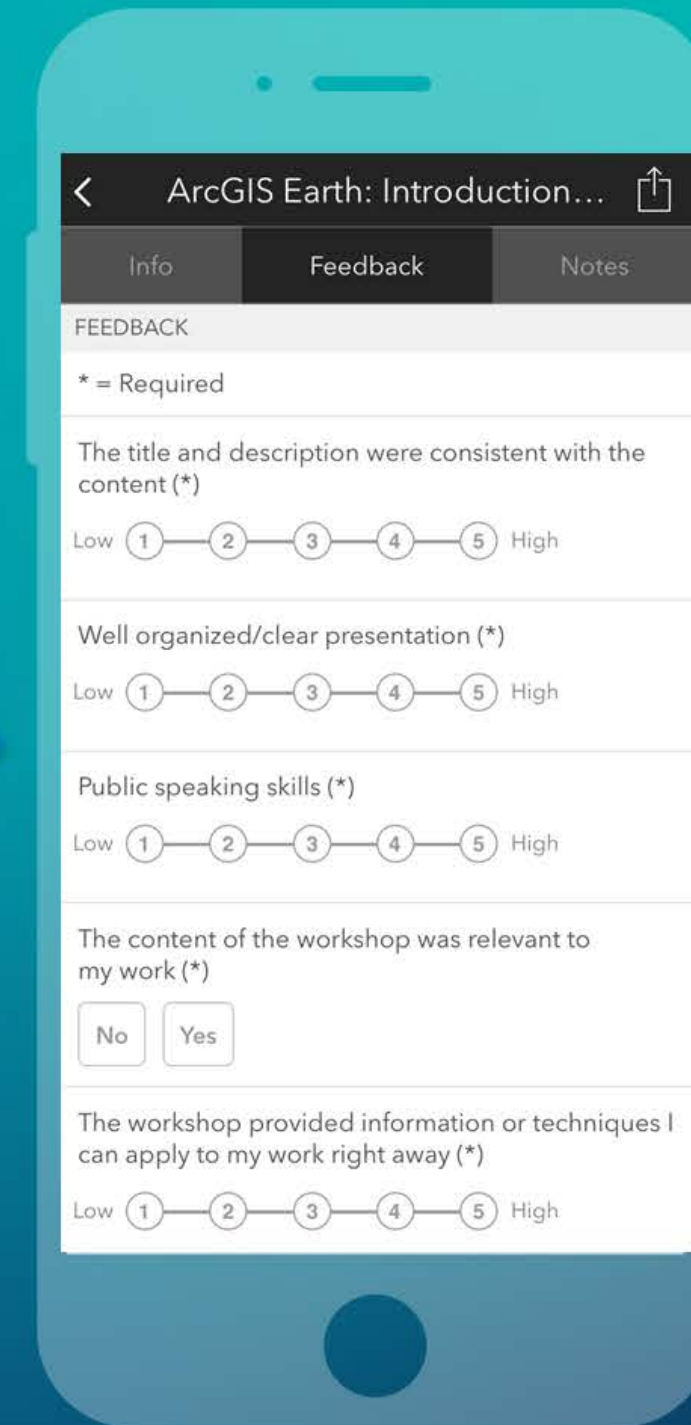
Select the session you attended



Scroll down to find the feedback section



Complete answers and select "Submit"



Questions?

For example

👓 *Where's Barbara?* 😭 🤔 *Where can I find the slides/source?*

👉 bit.ly/customwidgetsds19 👈



esri

THE
SCIENCE
OF
WHERE