

# ArcGIS Maps SDK for JavaScript: Better UX with Popups

Lauren Ballantyne & Lauren Boyd

```
<arcgis-map zoom="4" center="-118,34">
  <arcgis-search position="top-right" />
</arcgis-map>
```

```
view.goTo({
  center: [-126, 49]
})
.catch(function(error) {
  if (error.name != "AbortError") {
    console.error(error);
  }
});
```

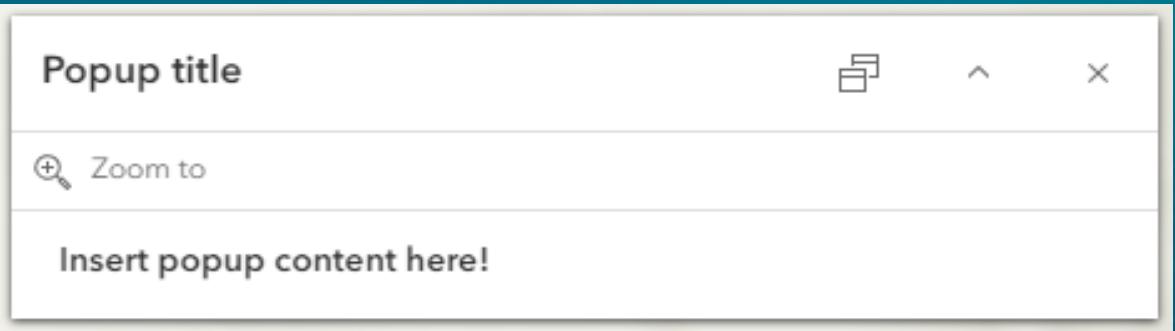
# Agenda

- Intro to popups
- Popup Elements
- Arcade
- What's changed
- Demos
  - Configuring popups in the Map Viewer
  - Adding more to popups using JS Maps SDK

# Popups

What are they?

- Customizable
- Interactive
- Cross-platform feature
- Available for a variety of layer types



# Popup Elements

- Text
- Fields list
- Attachments
- Media (Charts and Images)
- Related records
- Arcade

The image displays three distinct popup elements, each with a title bar and various interactive components.

- Earthquake Alert (Top Left):** Title: "6.6 Magnitude Earthquake". Content includes: "614.46 km depth", "2024-01-20, 4:31 p.m.", "124 km NW of Tarauacá, Brazil (-7.28, -71.47)", "USGS PAGER ALERT: No Response Needed (GREEN)", and "TSUNAMI WARNING: No Alert Issued".
- River Flooding Status (Top Right):** Title: "Station: NWSEVRG1 (NOAA)". Content includes: "Status: MODERATE FLOOD", "Height: 14.23 FEET (4.34 METERS)", and "Flow: 61800 ft³/s (1749.98 m³/s)".
- Line Chart (Bottom Left):** Title: "Ohio County, Kentucky". Content includes: "Unemployment rate for the most recent 14 months of BLS data" and a line graph showing the unemployment rate fluctuating between approximately 4.5% and 6% over time.
- AED Location (Middle Right):** Title: "AED Location: York". Content includes: "Edit", "Get directions", "Zoom to", and a file icon labeled "AED\_Instructions.txt".
- Bird Species Observations (Bottom Right):** Title: "Bird species observations". Content includes: "Edit", "Get directions", "Zoom to", "Canada Goose" (with a detailed description of behavior comments), and a date "July 20, 2022".

# Arcade

What is it?

- ArcGIS's cross-platform expression language
- Create expressions to calculate new values on the fly
  - Use values
  - Use geometries
- Use FeatureSets to access data from other layers hosted in ArcGIS Online
- What Arcade is not:
  - Language for building apps



# Arcade

## Use in pop-ups

- Attribute expressions
- Arcade elements

The screenshot shows the ArcGIS Arcade editor interface. On the left, the 'Run' tab displays the following JavaScript code:

```
1  /*
2  Open the Suggestions tab and choose a template to get started creating different content types for your popups.
3  To learn more about using Arcade to create pop-up content visit:
4  https://developers.arcgis.com/arcade/guide/profiles/#popup-element
5 */
6
7 return {
8   type : 'text',
9   text : 'place your text or html here' //this property supports html tags
10 }
```

On the right, a sidebar titled 'Suggestions' is open, showing a search bar and a list of templates: 'Templates', 'Chart', 'Fields list', and 'Rich text'. There are also links for 'Profile variables', 'Functions', 'Suggestions', and 'Help'.

# What has changed since we last saw you?

- Design enhancements
- Custom chart colors
- New text editor plugins
- Honoring layer and feature order
- Arcade:
  - M-value support
  - Subtype group layer support
  - Knowledge graph

```
    queryParameters =  
        queryParameters().apply {  
            clause = "price > 200"  
        }  
    viewModelScope.launch {  
  
        featureTable.queryFeatures(queryParameters)  
            .onFailure {  
                showError("Feature query failed")  
            }.onSuccess { featureQueryResult ->  
                val featureResult =  
                    featureQueryResults.firstOrNull()  
            }  
    }  
}
```



# Map Viewer Popups

Lauren Ballantyne

# Going beyond Map Viewer

- Pros of Map Viewer

- Easy UI + Arcade editor
- No coding
- Configuration sticks with web map

- Pros of the JS Maps SDK

- Customize use case per custom application
- More control over popup UI

# Popups & the JS Maps SDK

Lauren Boyd

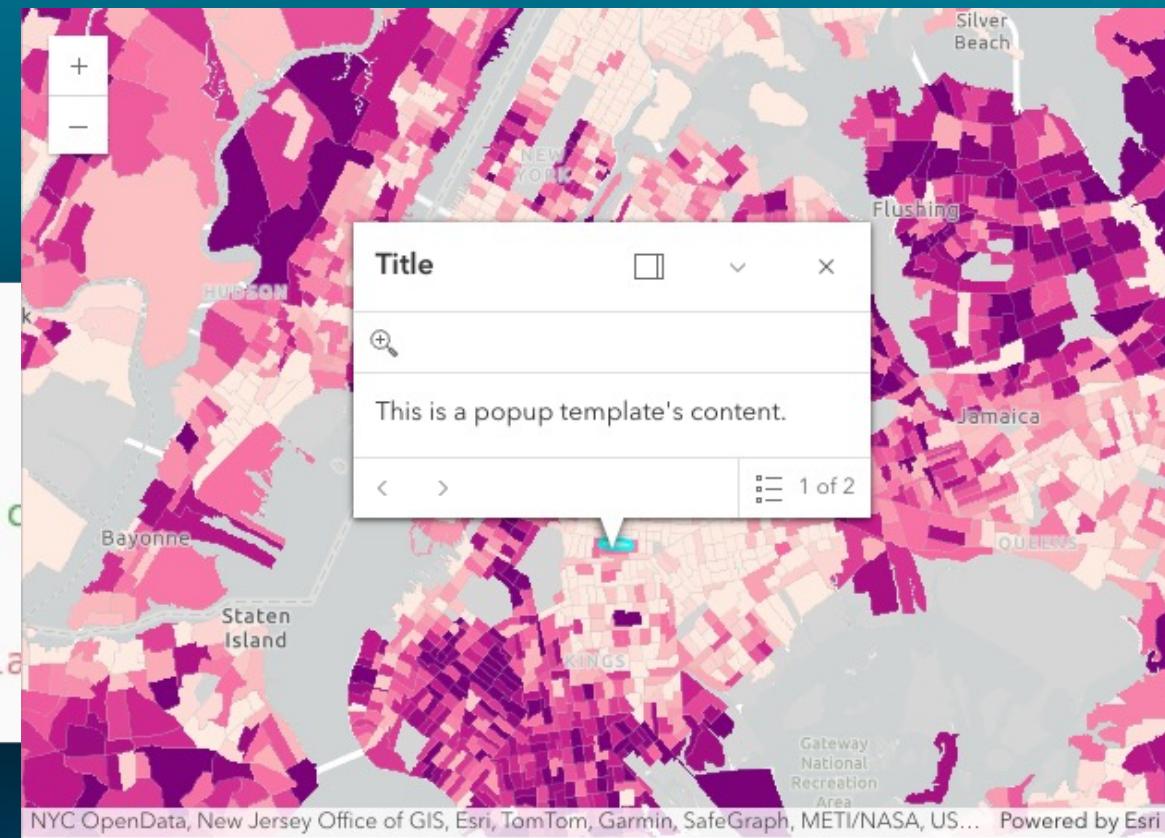
```
view.goTo({  
  center: [-126, 49]  
})  
.catch(function(error) {  
  if (error.name != "AbortError") {  
    console.error(error);  
  }  
});
```

```
queryParameters =  
QueryParameters().apply {  
  whereClause = "price > 200"  
}  
viewModelScope.launch {
```

# PopupTemplate

- Formats and defines the popup content
- Set defaultPopupTemplateEnabled

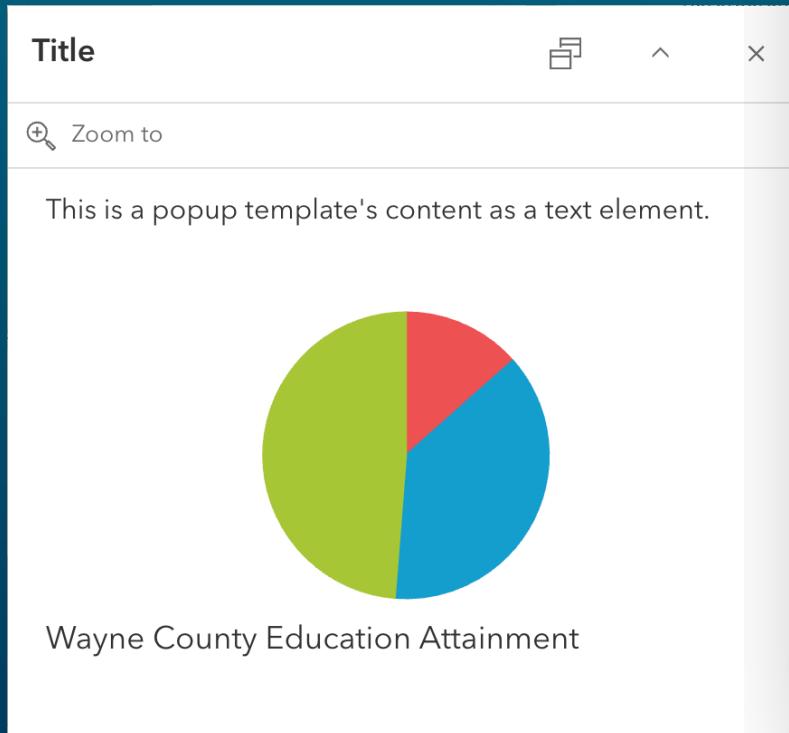
```
1 const popupTemplate = {  
2   // autocasts as new PopupTemplate()  
3   title: "Title",  
4   content: "This is a popup template's content."  
5 };  
6  
7 featureLayer.popupTemplate = popupTemplate;
```



# PopupTemplate

## Content

- String, function, promise, or array of elements



```
1 const popupTemplate = new PopupTemplate({
2   title: 'Title',
3   content: [
4     {
5       type: 'text',
6       text: "This is a popup template's content as a text element.",
7     },
8     {
9       type: 'media',
10      mediaInfos: [
11        {
12          title: '{COUNTY} County Education Attainment',
13          type: 'pie-chart',
14          value: {
15            fields: ['SOMEHS_CY', 'HSGRAD_CY', 'CollegeEducated'],
16          },
17        },
18      ],
19    },
20  ],
21 });
22 featureLayer.popupTemplate = popupTemplate
23
```

# Content elements

- Configured on individual layers' `popupTemplate`

Element	JS Maps SDK Class
Text	TextContent
Fields List	FieldsContent
Attachments	AttachmentContent
Media (charts, images)	MediaContent
Relationship browser	RelationshipContent
Arcade elements	ExpressionContent

# Content

## JS Maps SDK Specific

- CustomContent:
  - Displaying Widget

State: California

Zoom to

Wash

Washington

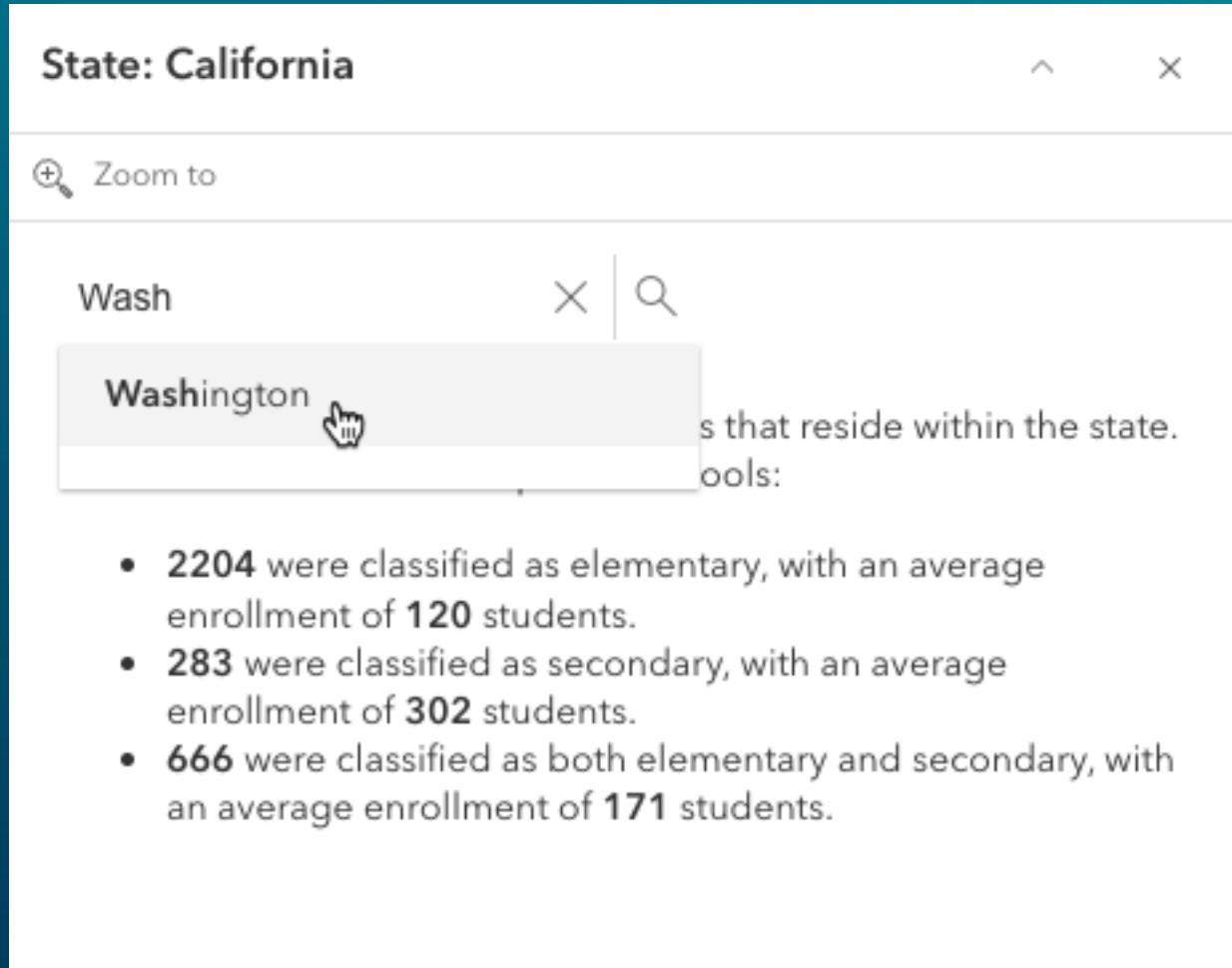
s that reside within the state.  
ools:

- 2204 were classified as elementary, with an average enrollment of 120 students.
- 283 were classified as secondary, with an average enrollment of 302 students.
- 666 were classified as both elementary and secondary, with an average enrollment of 171 students.

# Content

## JS Maps SDK Specific

- CustomContent:
  - Displaying Widget
- Promise
  - Useful when calling method/executing query and want to display results



# Content

## JS SDK Specific

- CustomContent:
  - Displaying Widget
- Promise
  - Useful when calling method/executing query and want to display results
- Function
  - Popup may require additional processing

The image shows two overlapping popups from a mobile application. The top popup is titled '#76 Kirkland Creek Backcountry Campsite'. It contains a 'Zoom to' button, a description of the campsite ('#76 Kirkland Creek resides along Lakeshore Trail. This site is currently Open and can be reached by Hiker Only.'), and notes about its location ('There are 6 sites and resides in Swain, North Carolina.') and access ('Site access restriction: Hiker.'). The bottom popup is titled 'Lakeshore Trail'. It also has a 'Zoom to' button and describes the trail ('Lakeshore Trail is a Bridle Path and currently Open. Near Hazel Creek and Unrestricted.'). Both popups have navigation arrows and a page number indicator (1 of 2 or 2 of 2) in the bottom right corner.

```
1 function formatContent
2   // Check if the color is green
3   // the color is red
4   let color =
5     feature.geometry.type === 'Point'
6       ? "green"
7       : "red"
8
9   let textContent
10  // Create the text content for the feature
11  // if the feature is a point
12  if (feature.geometry.type === 'Point') {
13    textContent = <b>{LABEL}</b> resides along {TRAIL}. This site is currently <span style='color: ' +
14      color + '>{ACCESS}</span>':
15  } else {
16    textContent = '{TRAIL} is a {TYPE} and currently {ACCESS}.
17    Near {CAMP} and {UNRESTRICTED}' +
18    color
19  }
20  return [textContent];
21}
22};
```

# Docking

```
1 view.popup = {  
2   dockEnabled: true,  
3   dockOptions:{  
4     // Sets which position to place the popup in the view.  
5     position: "top-right",  
6     // Disables the dock button from the popup.  
7     buttonEnabled: false,  
8     // Ignores the default sizes that triggers popup docking.  
9     // The popup will always be docked when opened.  
10    breakpoint: false  
11  }  
12};
```

# VisibleElements

featureNavigation →

heading →

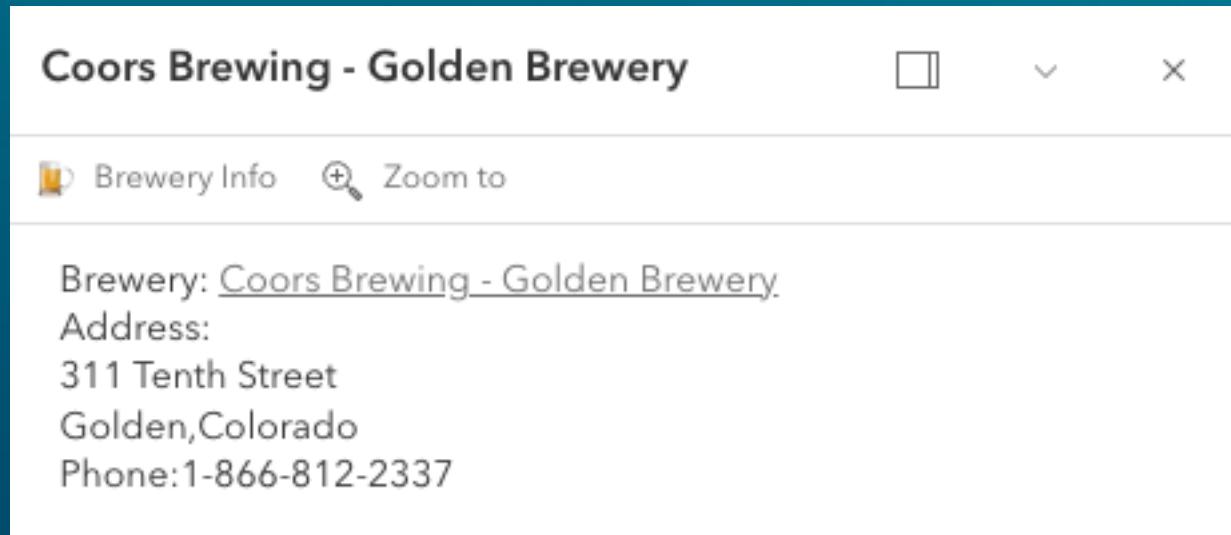
actionBar →

The screenshot shows a modal window with the following elements:

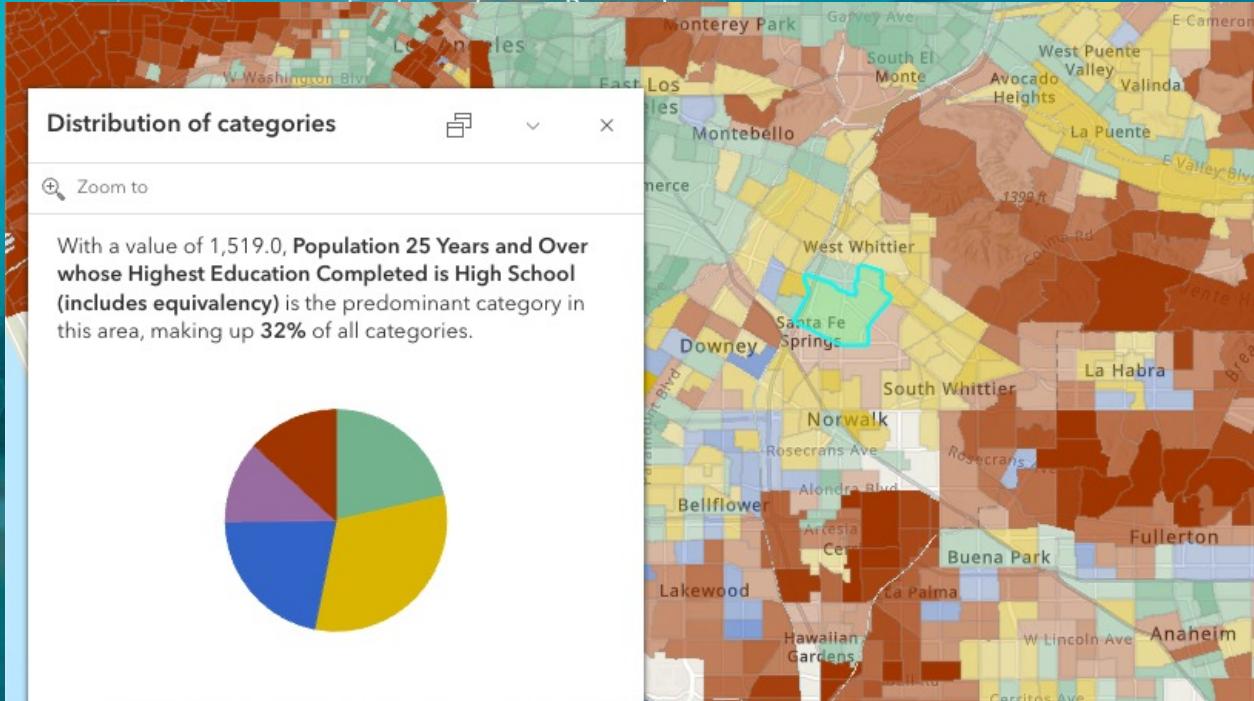
- featureNavigation:** Navigation arrows at the top left.
- heading:** "Union County in New Jersey" at the top center.
- actionBar:** Buttons for "Zoom to" and "Close" at the top right.
- closeButton:** A small "X" button in the top right corner of the modal.
- collapseButton:** A double-headed arrow icon in the top right corner of the modal.
- Text:** "The median household income in this area is estimated to be **\$95,000.00** ( $\pm \$2,210.00$ )."
- Section:** "Median income details for Union County".
- Figure:** A bar chart titled "Income by householder age" showing median household income by age group. The y-axis ranges from 0 to 150,000. The chart contains four bars: one dark green bar at approximately 50,000, two light green bars at approximately 95,000 and 115,000, and one light green bar at approximately 70,000.
- Caption:** "Income by householder age" and "Median household income by householder age."
- Pagination:** "1 of 3" at the bottom right.

# Actions

- Default actions
  - Zoom to
  - Browse features (clustering)
- Custom Actions
  - Styled with image or Calcite icon



```
    queryParameters =  
        queryParameters().apply {  
            clause = "price > 200"  
        }  
    viewModelScope.launch {
```



# Configuring Popups

Lauren Boyd

# What's to come

- Broader related record support
- Popups for vector tile layers
- Attachment improvements
- Utility Network association support

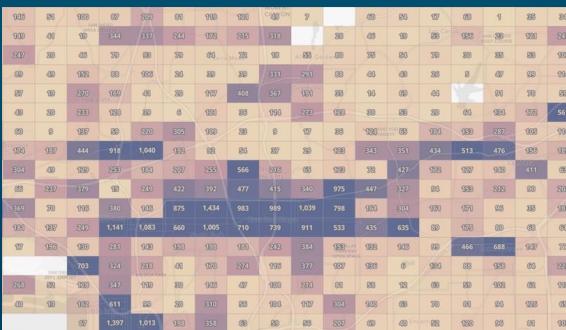
# Resources



[Improving the rich text editing experience in pop-ups](#)



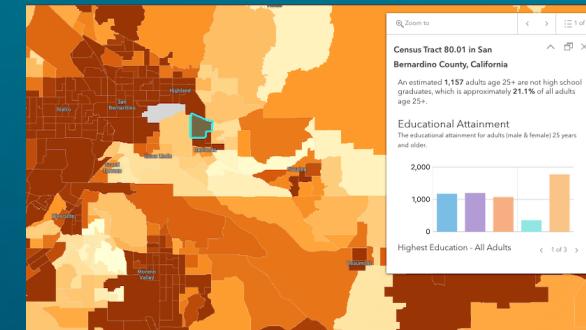
[Introducing custom colors in popup charts](#)



[How to summarize aggregate data using Arcade in popups](#)



[Rolling out related records in Map Viewer](#)



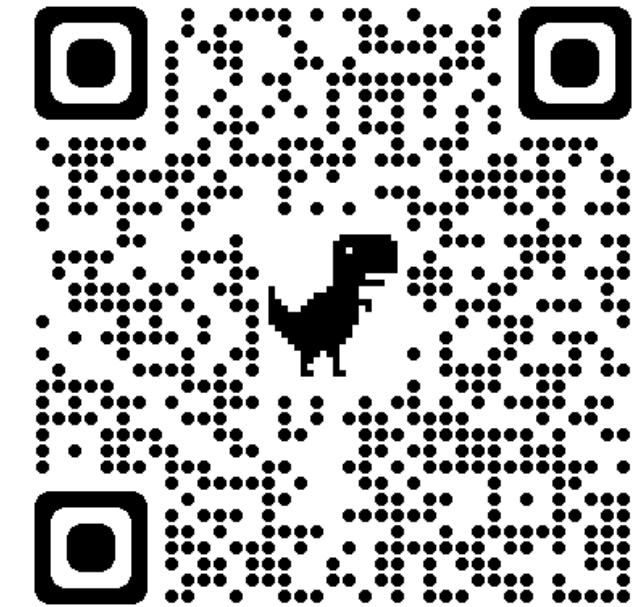
[A fresh new look for popups](#)



[Browsing related records with the ArcGIS API for JavaScript](#)

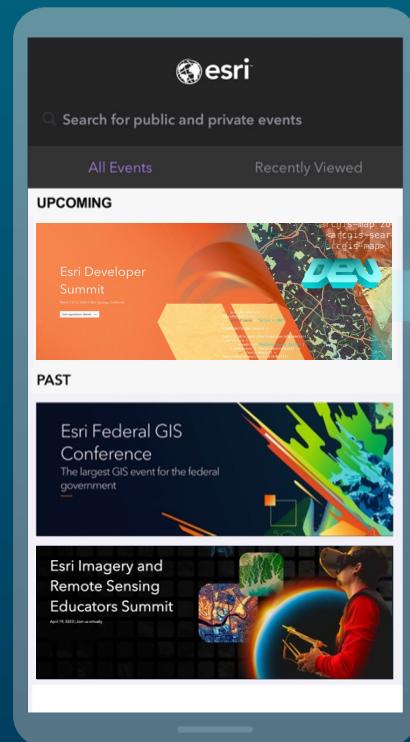
# Link to slides & demos

```
view.goTo({  
  center: [-126, 49]  
})  
.catch(function(error) {  
  if (error.name === "AbortError") {  
    return;  
  }  
});
```

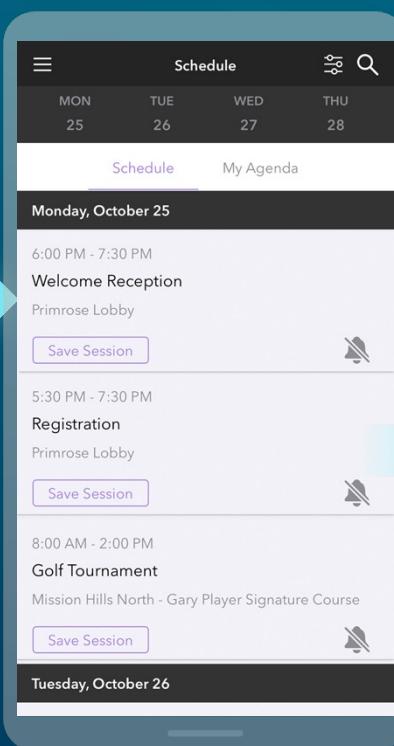


# Please Share Your Feedback in the App

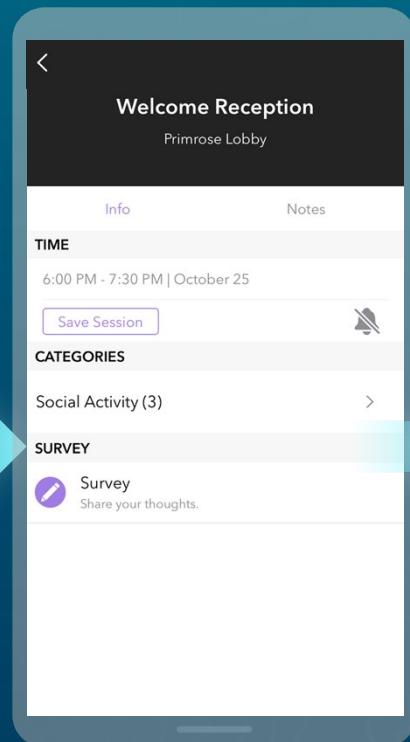
Download the Esri Events app and find your event



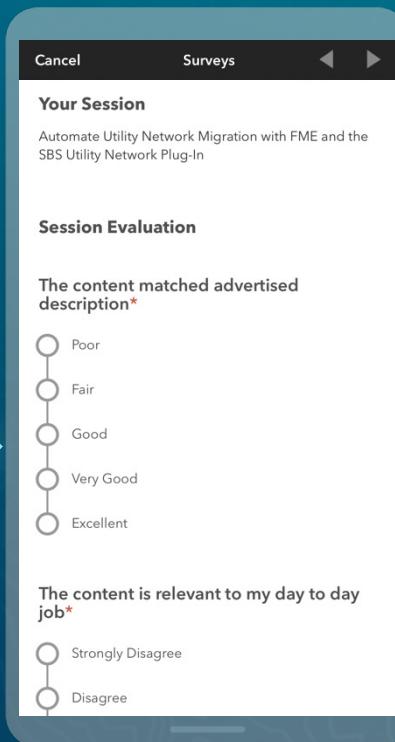
Select the session you attended



Scroll down to "Survey"



Log in to access the survey



# Connect With Us On Social

And Join the Conversation Using #EsriDevSummit2024

-  [twitter.com/EsriDevs](https://twitter.com/EsriDevs) #esridevsummit2024
-  [twitter.com/EsriDevEvents](https://twitter.com/EsriDevEvents)
-  [youtube.com/c/EsriDevelopers](https://youtube.com/c/EsriDevelopers)
-  [links.esri.com/DevVideos](https://links.esri.com/DevVideos)
-  [github.com/Esri](https://github.com/Esri)
-  [github.com/EsriDevEvents](https://github.com/EsriDevEvents)
-  [links.esri.com/EsriDevCommunity](https://links.esri.com/EsriDevCommunity)

```
<arcgis-map zoom="4" center="-118,34" />  
  
view.goTo({  
  center: [-126, 49]  
})  
.catch(function(error) {  
  if (error.name != "AbortError") {  
    console.error(error);  
  }  
});
```

```
// show the compass and pass the  
mapRotation state data  
Compass(rotation = mapRotation)  
    // reset the ComposableMapView's viewpoint  
rotation to point north using the  
mapViewModel  
    mapViewModel.setViewpointRotation(0.0)  
}
```



esri®

THE  
SCIENCE  
OF  
WHERE®

```
const layerList = new LayerList({  
    view: view  
});  
  
// Add widget to the top right corner  
// of the view  
view.ui.add(layerList, "top-right")  
  
<arcgis-map zoom="4" center="-118,34"
```