

ArcGIS Maps SDK for JavaScript: Create a Custom Visualization Style

Kristian Ekenes

Anne Fitz

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

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

What to expect in this session

- Create innovative mapping styles for your data using public Renderer APIs in the JS Maps SDK.
- You should be familiar with maps, layers, and creating an app with the JS Maps SDK.
- You should be familiar with renderers.
- You should be familiar with Arcade.
- You do not need to be a graphics developer or have extensive WebGL knowledge.

Renderers

The vehicle for data-driven visualization

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

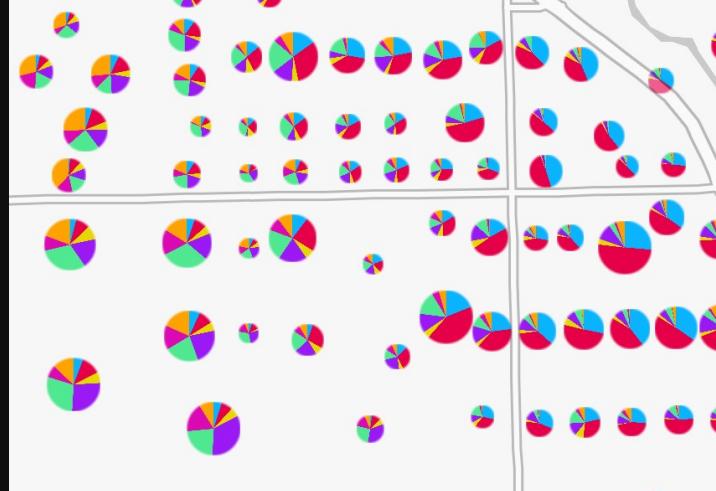
```
    queryParameters =  
    QueryParameters().apply {  
        whereClause = "price > 200"  
    }  
    viewModelScope.launch {
```

Renderers

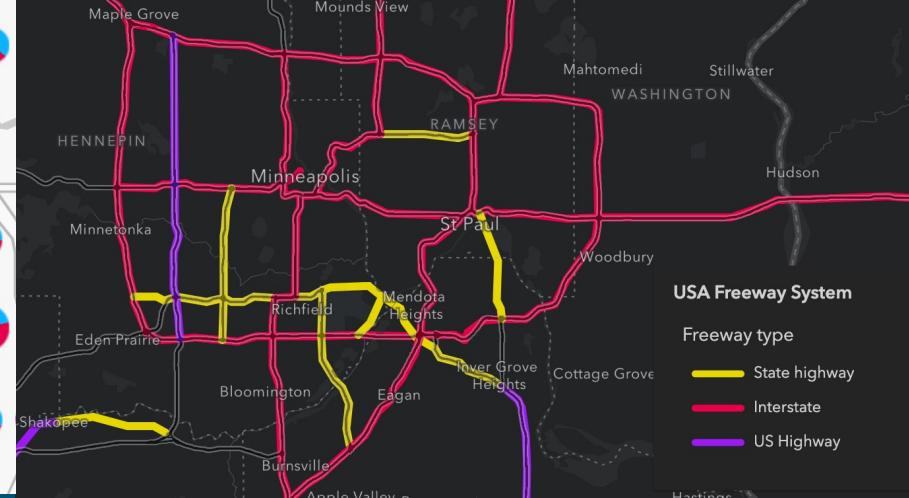
SimpleRenderer



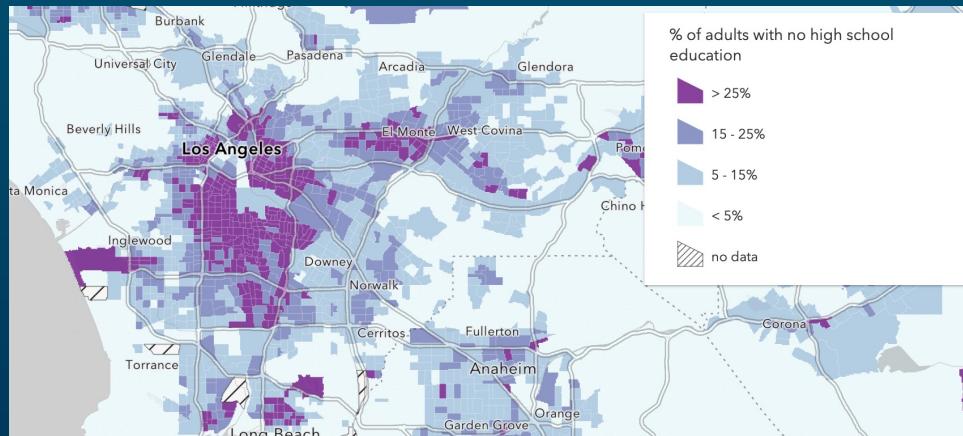
PieChartRenderer



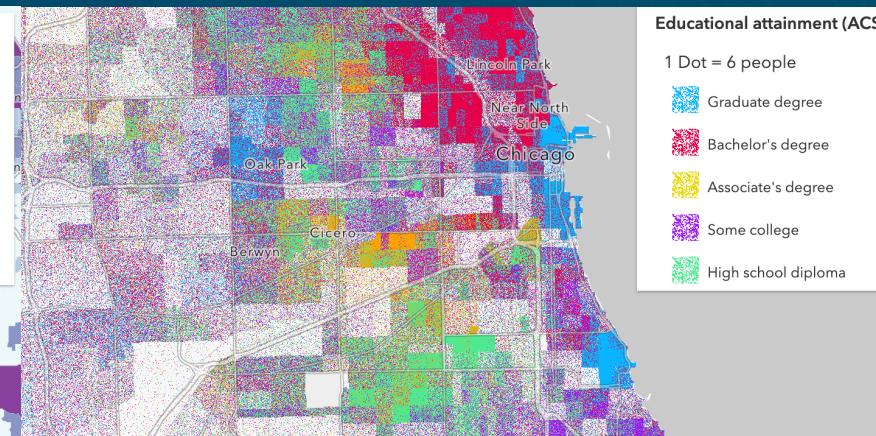
UniqueValueRenderer



ClassBreaksRenderer



DotDensityRenderer



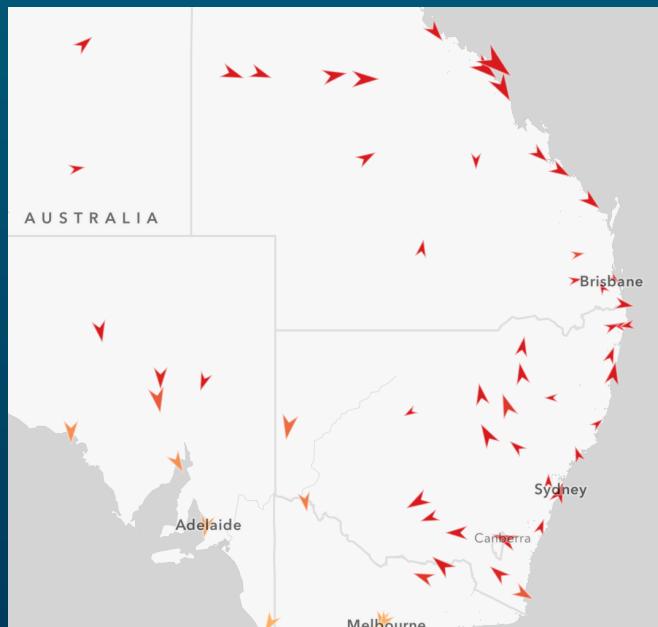
HeatmapRenderer



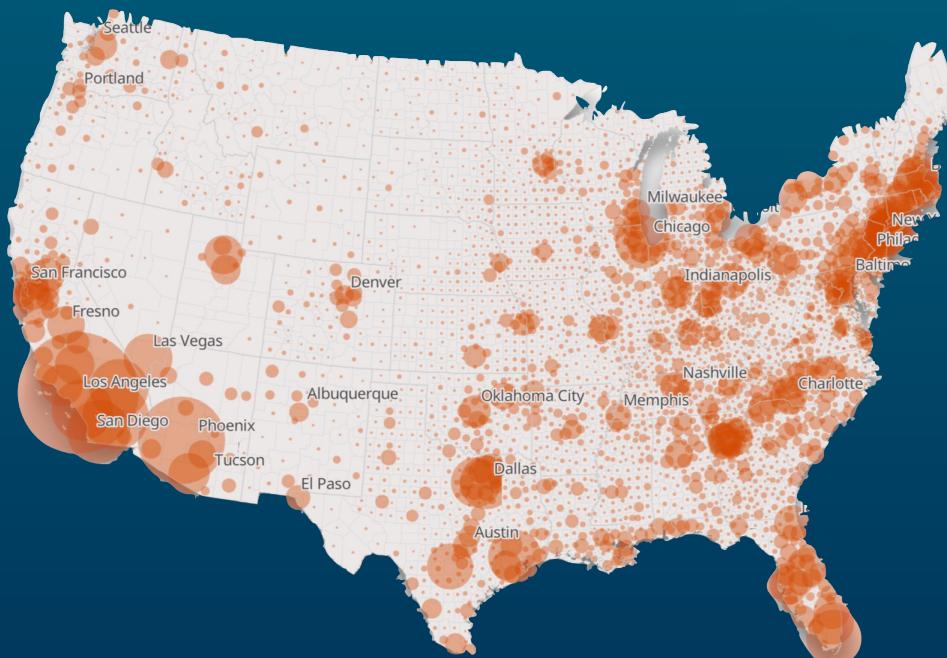
Visual variables

- Override either color, opacity, size, or rotation for **all symbols** in the renderer with a data value (either a field value, or an Arcade expression)

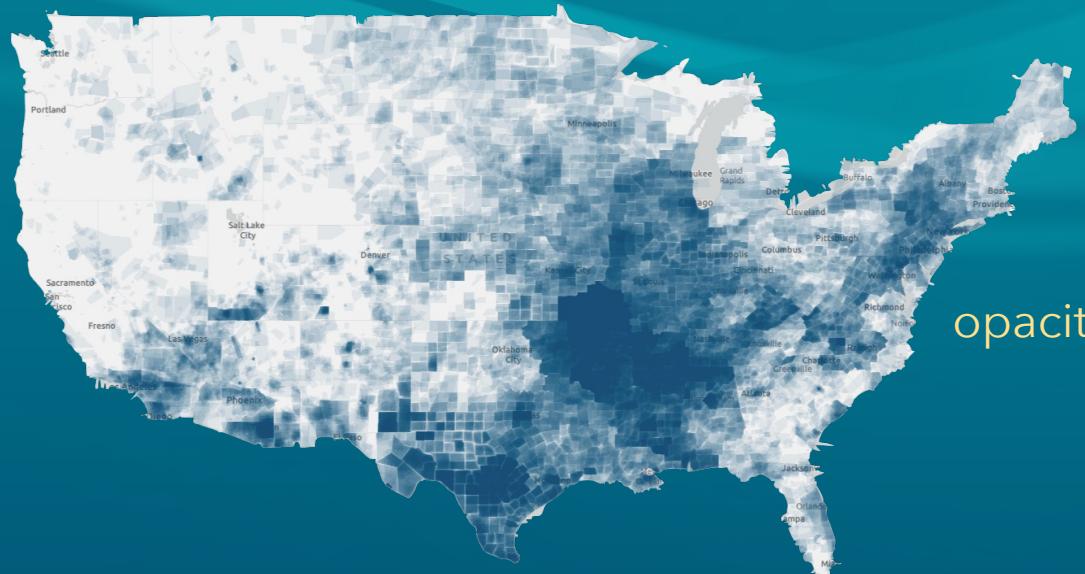
rotation



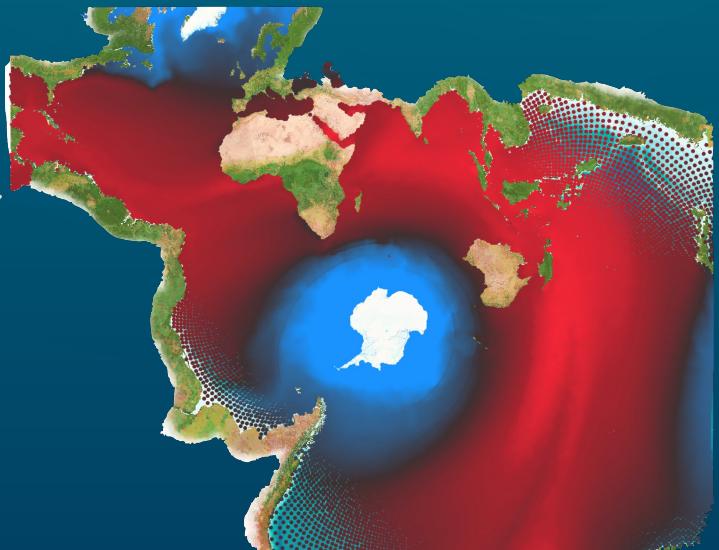
size



opacity

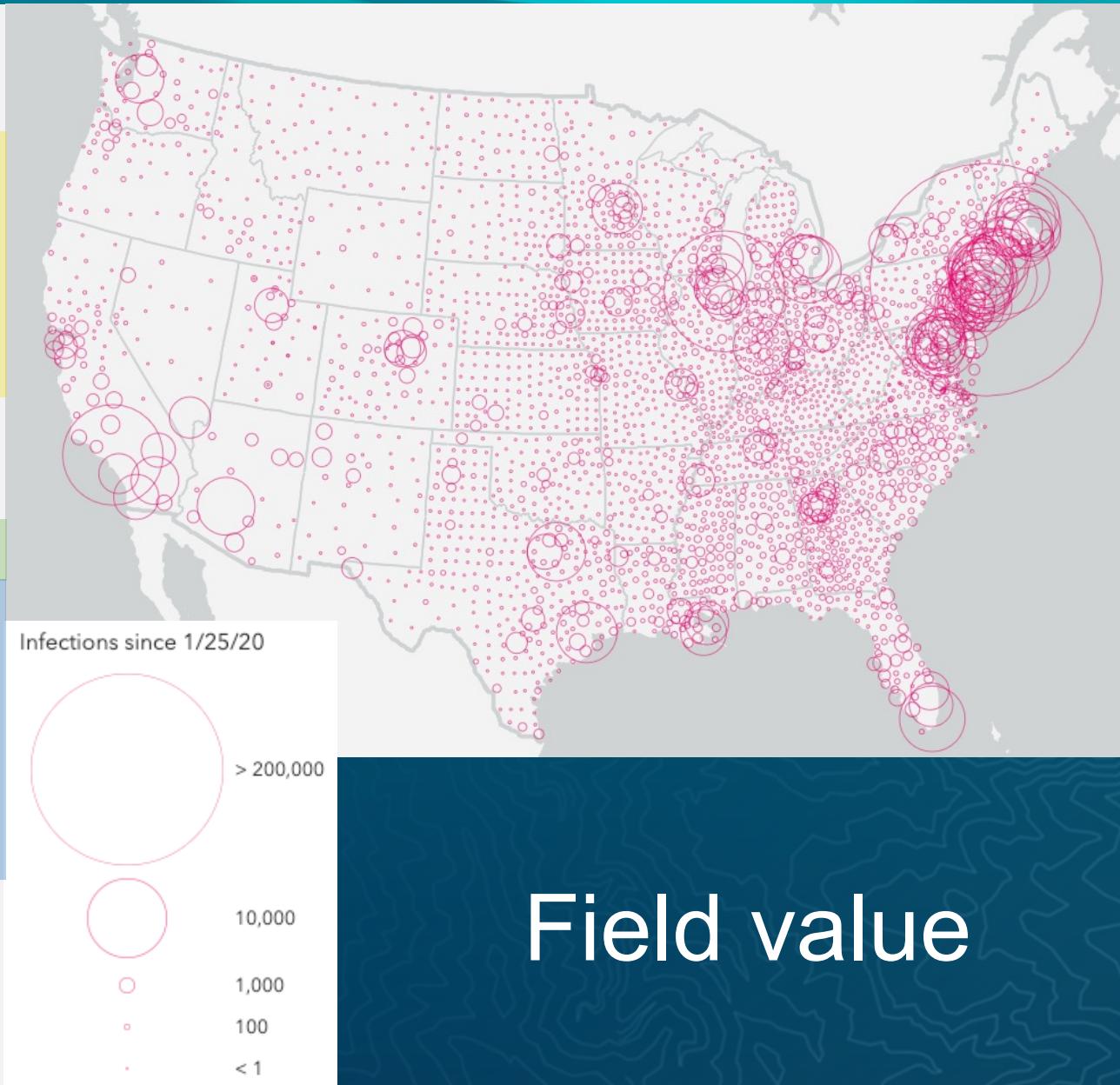


color



Data-driven visualization with Size Visual Variable

```
layer.renderer = new SimpleRenderer({  
    symbol: new SimpleMarkerSymbol({  
        style: "none",  
        outline: new SimpleLineSymbol({  
            color: new Color("rgba(227, 0, 106)"),  
            width: 0.5  
        })  
    }),  
    visualVariables: [  
        new SizeVariable({  
            field: "INFECTIONS_6_1_2020",  
            stops: [  
                { value: 1, size: "2px" },  
                { value: 100, size: "4px" },  
                { value: 1000, size: "10px" },  
                { value: 10000, size: "50px" },  
                { value: 200000, size: "200px" }  
            ]  
        })  
    ]  
});
```



Custom vector symbols

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

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

CIMSymbol

Custom vector symbols

High quality, scalable



Scaled vector symbol



Scaled image

Symbol layers



Symbol



1

layer1



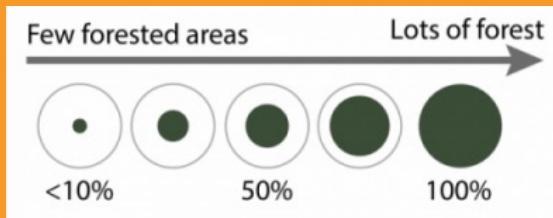
layer2



layer3

Primitive Overrides

Dynamically update a property on an **individual symbol layer** using Arcade



Primitive Overrides

```
const cimSymbol = new CIMSymbol({  
  data: {  
    type: "CIMSymbolReference",  
    symbol: {  
      type: "CIMPointSymbol",  
      symbolLayers: [{ ... }]  
    },  
    primitiveOverrides: [{  
      type: "CIMPrimitiveOverride",  
      primitiveName: "symbol-layer-1", // the name of the symbol layer we want to override  
      propertyName: "Size", // the name of the property on the symbol layer we want to override  
      valueExpressionInfo: {  
        type: "CIMExpressionInfo",  
        title: "Size override",  
        expression: "return $feature.treeHeight;" // the expression to change the size of the symbol  
      }  
    }]  
  }  
});
```

\$feature.treeHeight = 25

```
symbolLayers: [{  
  type: "CIMVectorMarker",  
  primitiveName: "symbol-layer-1",  
  enable: true,  
  size: 40, 25  
  frame: {  
    xmin: 0,  
    ymin: 0,  
    xmax: 17,  
    ymax: 17  
  },  
  markerGraphics: [ ... ]  
}]
```

Results of the **expression** become the
size of the symbol layer

Create custom renderers using Arcade

Renderers

- SimpleRenderer
- ClassBreaksRenderer
- UniqueValueRenderer
- DotDensityRenderer
- HeatmapRenderer
- PieChartRenderer

Symbols

- SimpleMarkerSymbol
- SimpleLineSymbol
- SimpleFillSymbol
- PictureMarkerSymbol
- PictureFillSymbol
- CIMSsymbol

Visual Variables

- Color
- Size
- Opacity
- Rotation

Classes that have a
valueExpression property.

Arcade

Anne

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

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

What is Arcade?

Create flexible content from existing data

- **Expression language** for...
 - calculating and formatting data values,
 - enforcing rules,
 - generating content elements, and
 - Creating any of the above conditionally.
- Works across ArcGIS applications
 - ArcGIS Pro
 - ArcGIS Online
 - ArcGIS Dashboards
 - Developer SDKs
 - ... and more!



Arcade Visualization profile

<https://developers.arcgis.com/arcade/profiles/visualization/>

Profile variables

- \$feature: the feature to be drawn on the map
- \$view.scale: the scale of the map when the expression evaluates

Function bundles

- Core
- Geometry

Return types

- Number
- Text

```
<script type="text/plain" id="winning-party">

var republican = $feature.MP06025a_B;
var democrat = $feature.MP06024a_B;
var independent = $feature.MP06026a_B;
var parties = [republican, democrat, independent];
// Decode() and Max() are built-in Arcade functions
return Decode( Max(parties),
    republican, 'republican',
    democrat, 'democrat',
    independent, 'independent',
    'n/a');

</script>
```

Arcade Visualization Profile

<https://developers.arcgis.com/arcade/profiles/visualization/>

- Look for the `valueExpression` property on the renderer, visual variable, or primitive override

Where is it used?

- ClassBreaksRenderer
- UniqueValueRenderer
- Visual Variables
- CIM Primitive Overrides

```
const winnerArcade = document.getElementById("winning-party").text;

const renderer = new UniqueValueRenderer({
  valueExpression: winnerArcade,
  valueExpressionTitle: "Counties by dominant party among registered voters",
  uniqueValueInfos: [
    {
      value: "democrat",
      symbol: createSymbol("#00c3ff"),
      label: "Democrat"
    },
    {
      value: "republican",
      symbol: createSymbol("#ff002e"),
      label: "Republican"
    },
    {
      value: "independent",
      symbol: createSymbol("#faff00"),
      label: "Independent/non-affiliated"
    }
});
```

Data-driven visualization

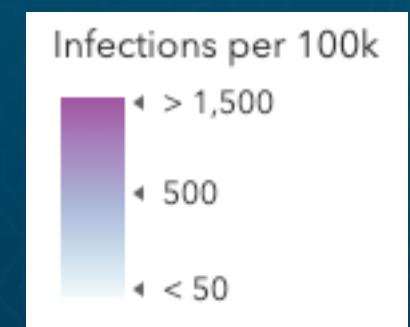
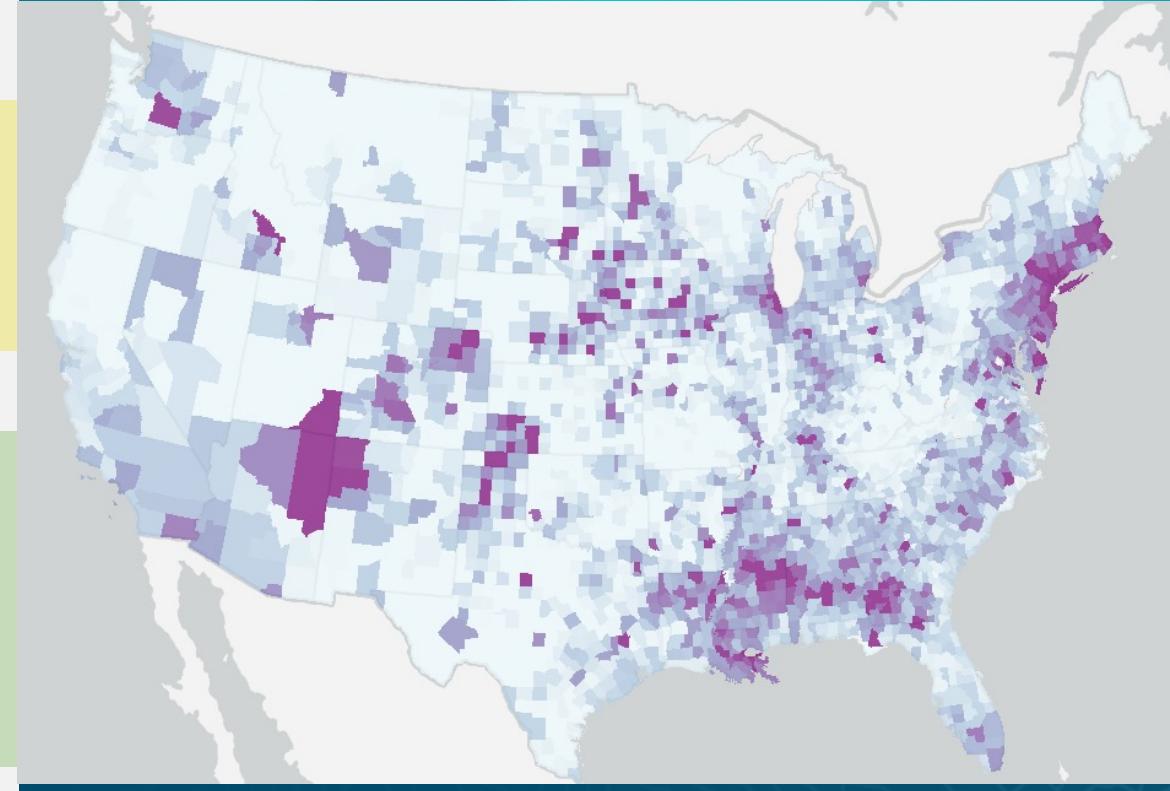
Symbol

```
layer.renderer = new SimpleRenderer({
  symbol: new SimpleFillSymbol({
    outline: new SimpleLineSymbol({
      color: "rgba(128,128,128,0.4)",
      width: 0
    })
  }),
  visualVariables: [
    new ColorVariable({
      valueExpression: `

        var currentDayValue = $feature["DAYSTRING_05_31_2020"];
        var currentDaySplit = Split(currentDayValue, "|");
        var infections = Number(currentDaySplit[0]);
        var deaths = Number(currentDaySplit[1]);
        var population = $feature.Population_1;
        return (infections / population ) * 100000;
      `,
      valueExpressionTitle: `Infections per 100k`,
      stops: [
        { value: 50, color: "#edf8fb" },
        { value: 200, color: "#b3cde3" },
        { value: 500, color: "#8c96c6" },
        { value: 1000, color: "#8856a7" },
        { value: 1500, color: "#810f7c" }
      ]
    })
  ]
});
```

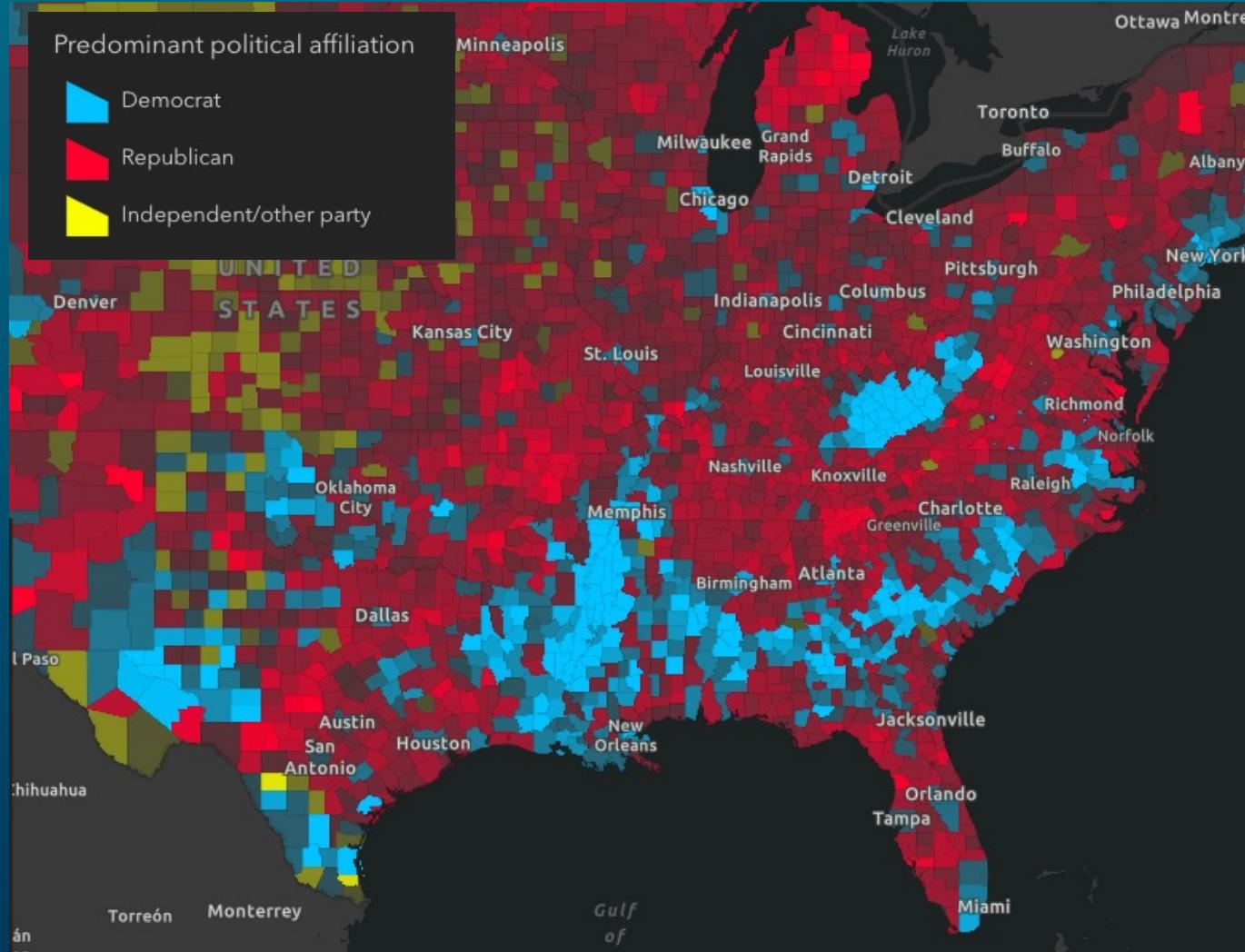
Data

Color
driven
by data



Arcade
expression

Map predominance with Arcade



```
const renderer = {
  type: "unique-value",
  valueExpression: `

    var republican = $feature.MP06025a_B;
    var democrat = $feature.MP06024a_B;
    var independent = $feature.MP06026a_B;
    var parties = [republican, democrat, independent];

    return Decode( Max(parties),
      republican, 'republican',
      democrat, 'democrat',
      independent, 'independent',
      'n/a' );
  `,

  valueExpressionTitle: "Predominant political affiliation",
  uniqueValueInfos: [
    {
      value: "democrat",
      symbol: createSymbol("#00c3ff"),
      label: "Democrat"
    },
    {
      value: "republican",
      symbol: createSymbol("#ff002e"),
      label: "Republican"
    },
    {
      value: "independent",
      symbol: createSymbol("#faff00"),
      label: "Independent/other party"
    }
  ]
};
```

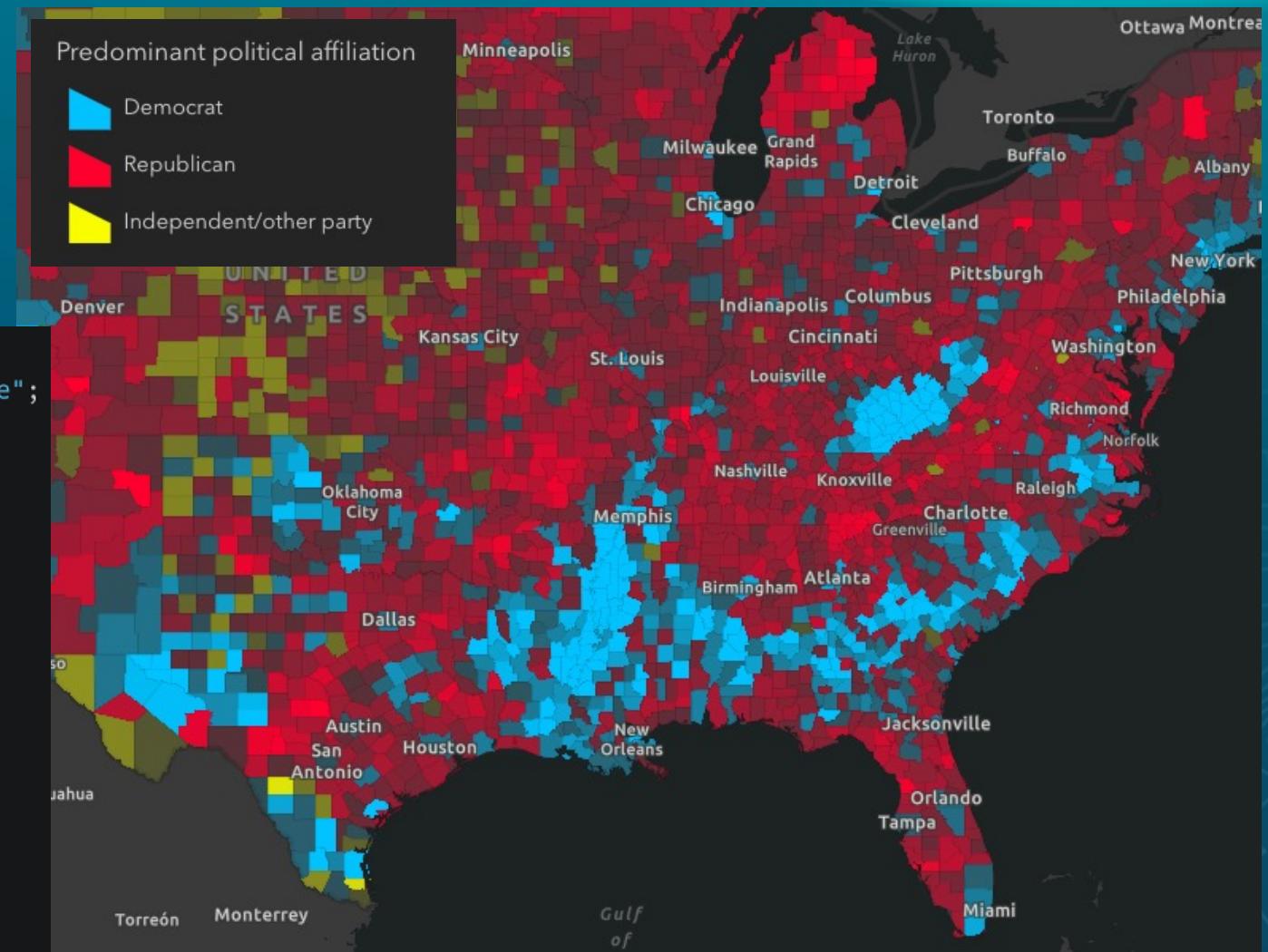
Smart Mapping

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

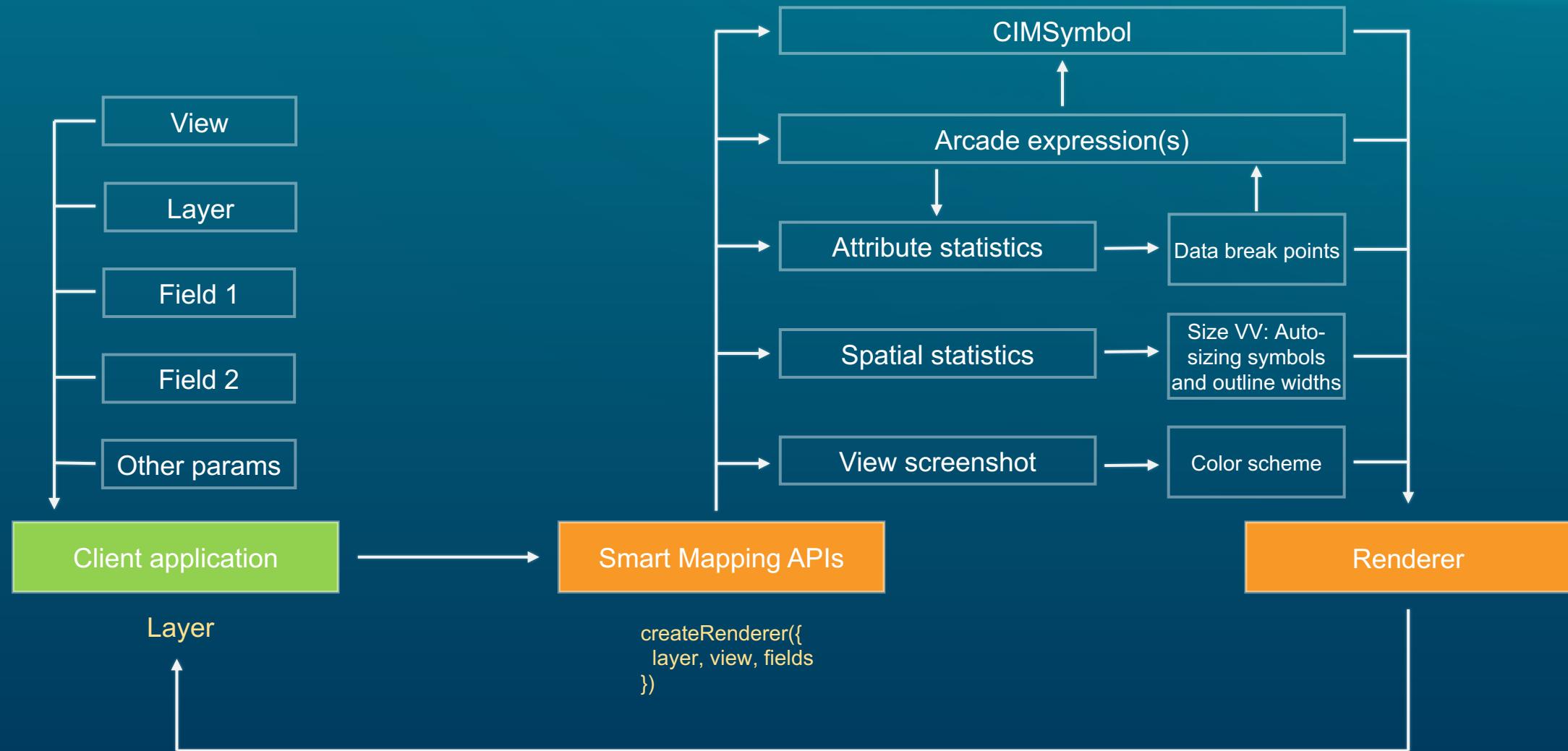
```
    queryParameters =  
    QueryParameters().apply {  
        whereClause = "price > 200"  
    }  
    viewModelScope.launch {
```

Smart mapping - Predominance

- **Statistics** - Which breaks/stop points make sense?
 - **Color** – What colors work best with my basemap?
 - **Arcade** – Which field is largest?



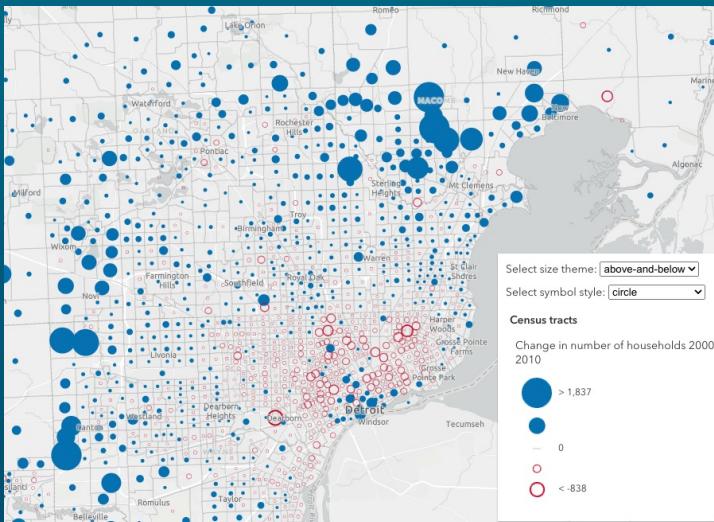
Smart mapping: generating Arcade expressions



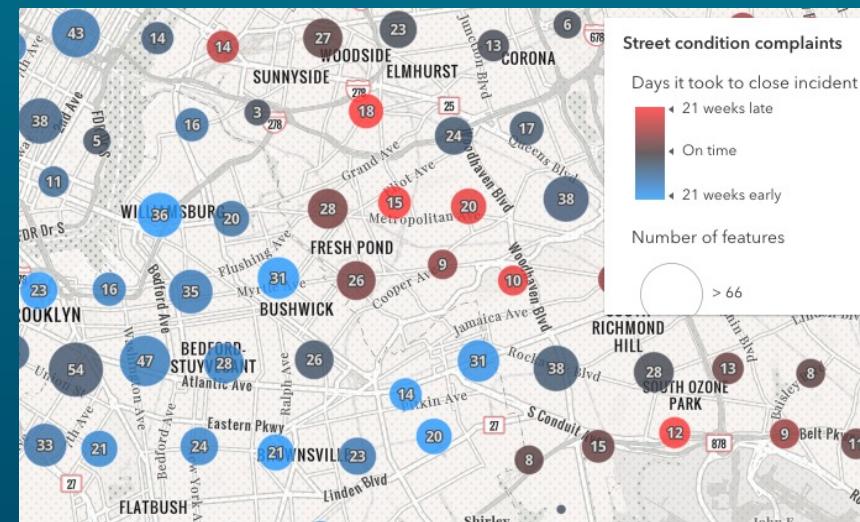
Arcade-based renderers available in Smart Mapping

Derivative styles created from standard renderer types using Arcade

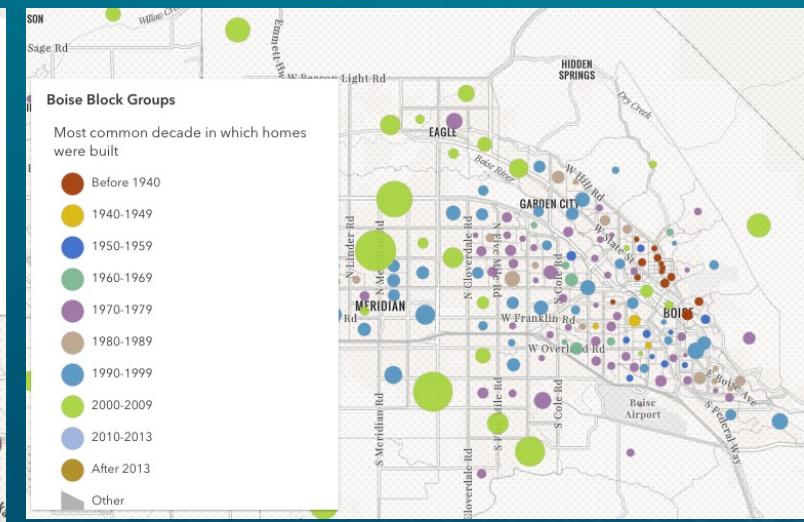
Above and below



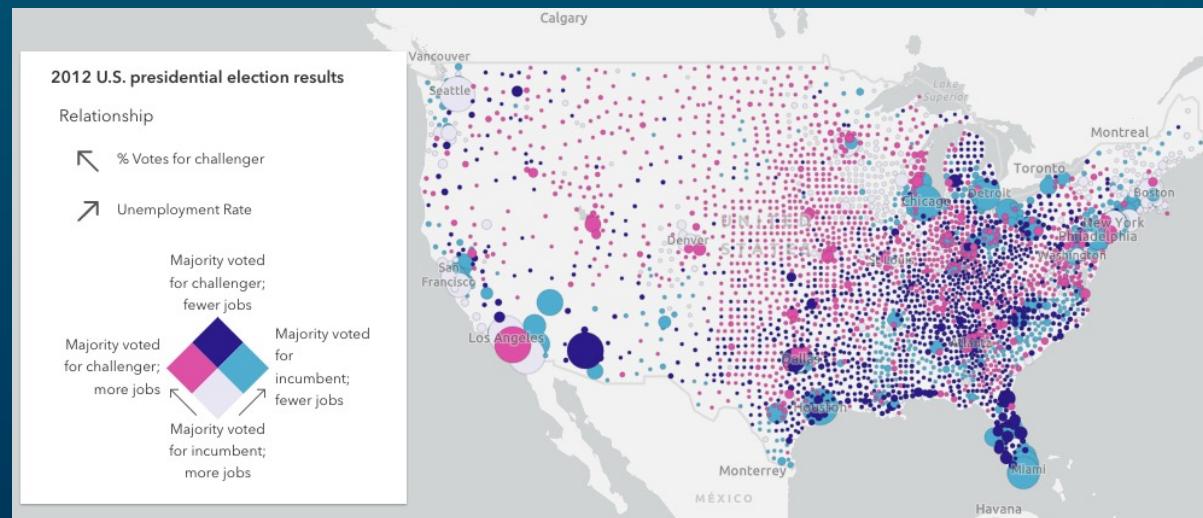
Age



Predominance



Relationship



Each returns an instance
of a renderer class
generated with a function.

Winnipeg

Vancouver

Seattle

Portland

Sacramento

Show descriptive labels

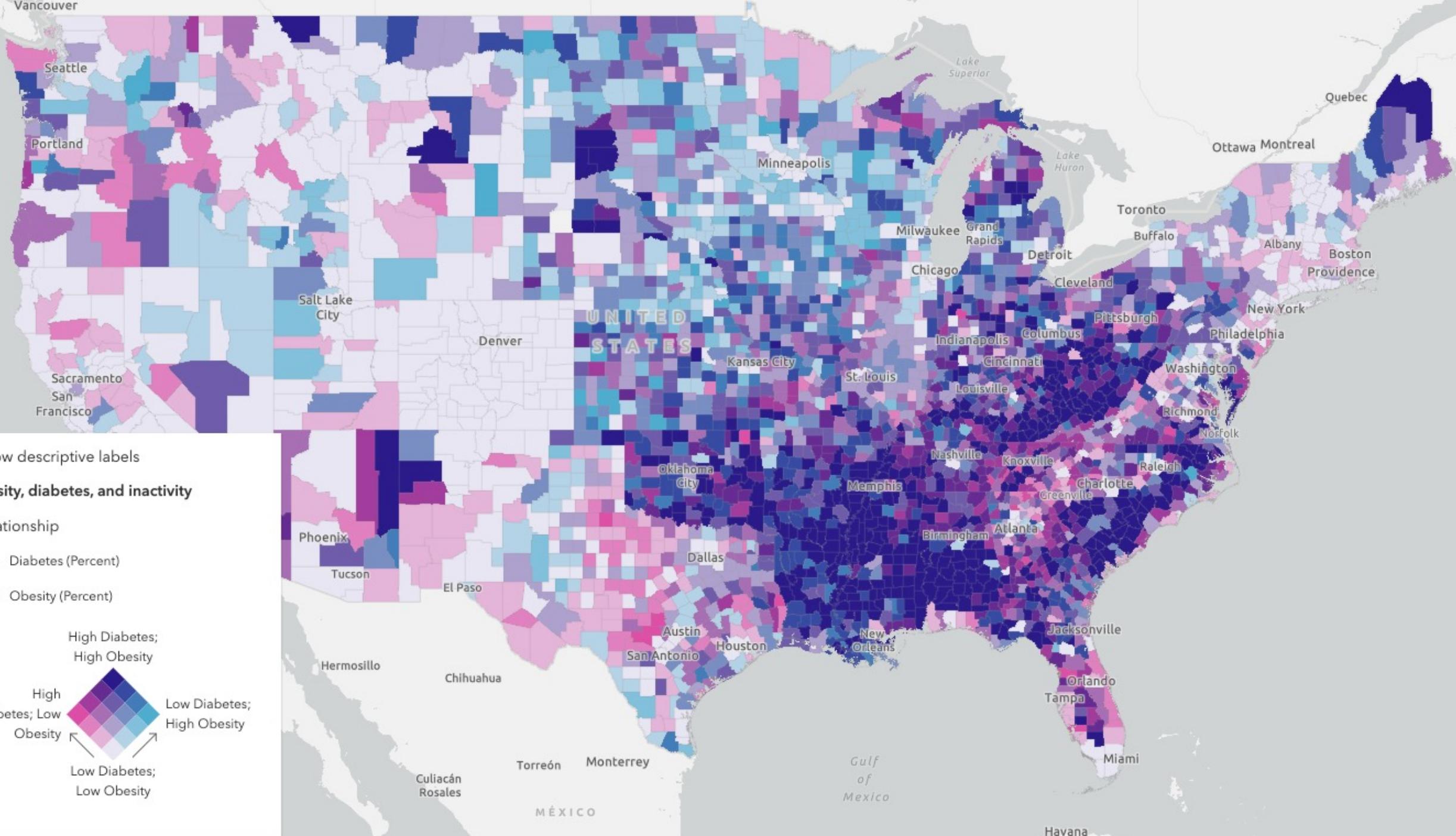
Obesity, diabetes, and inactivity

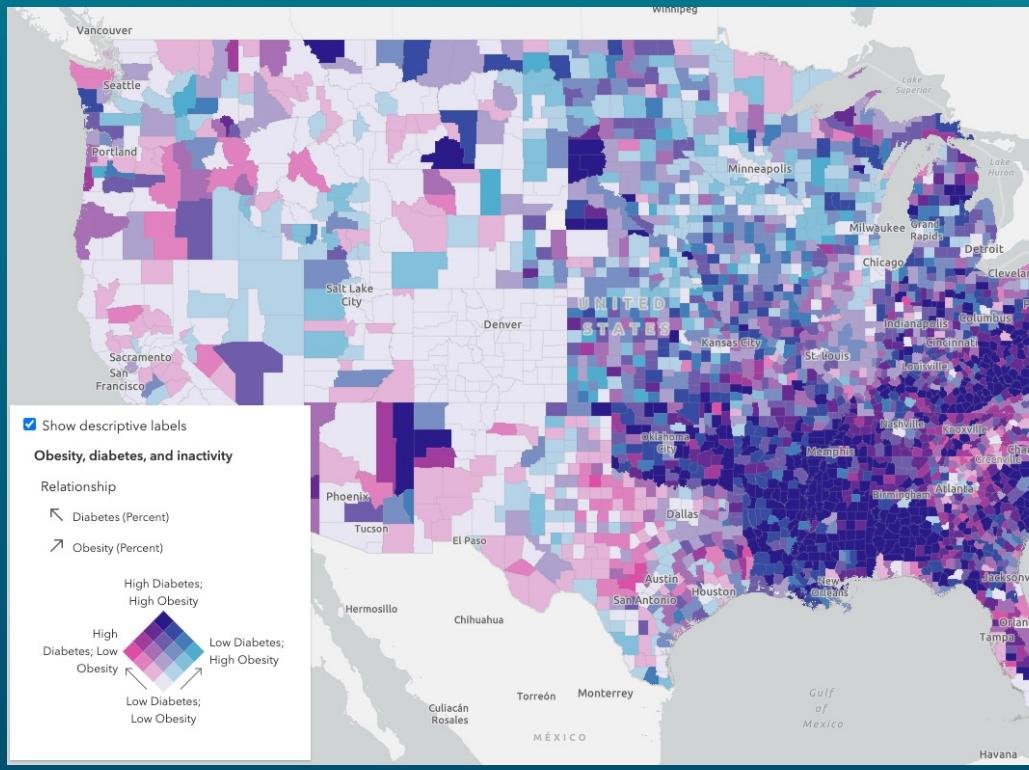
Relationship

↖ Diabetes (Percent)

↗ Obesity (Percent)

High Diabetes;
High Obesity





```

renderer: {,...}
  ▶ authoringInfo: {type: "relationship", ...}
    type: "uniqueValue"
  ▶ uniqueValueInfos: [{value: "HH", symbol: ...}]
    valueExpression: "var field1 = $feature[...]"
    valueExpressionTitle: "Relationship"
  
```



```

authoringInfo:
  ▶ field1: {field: "Relationship", ...}
    classBreakInfo: [0, 1, 2]
    uniqueValueInfos: [{value: "HH", symbol: "triangle-down"}, {value: "HM2", symbol: "triangle-down"}, {value: "HM1", symbol: "triangle-down"}, {value: "HL", symbol: "triangle-down"}, {value: "M2H", symbol: "triangle-down"}, {value: "M2M2", symbol: "triangle-down"}, {value: "M2M1", symbol: "triangle-down"}, {value: "M2L", symbol: "triangle-down"}, {value: "M1H", symbol: "triangle-down"}, {value: "M1M2", symbol: "triangle-down"}, {value: "M1M1", symbol: "triangle-down"}, {value: "M1L", symbol: "triangle-down"}, {value: "LH", symbol: "triangle-down"}, {value: "LM2", symbol: "triangle-down"}, {value: "LM1", symbol: "triangle-down"}, {value: "LL", symbol: "triangle-down"}]
  
```

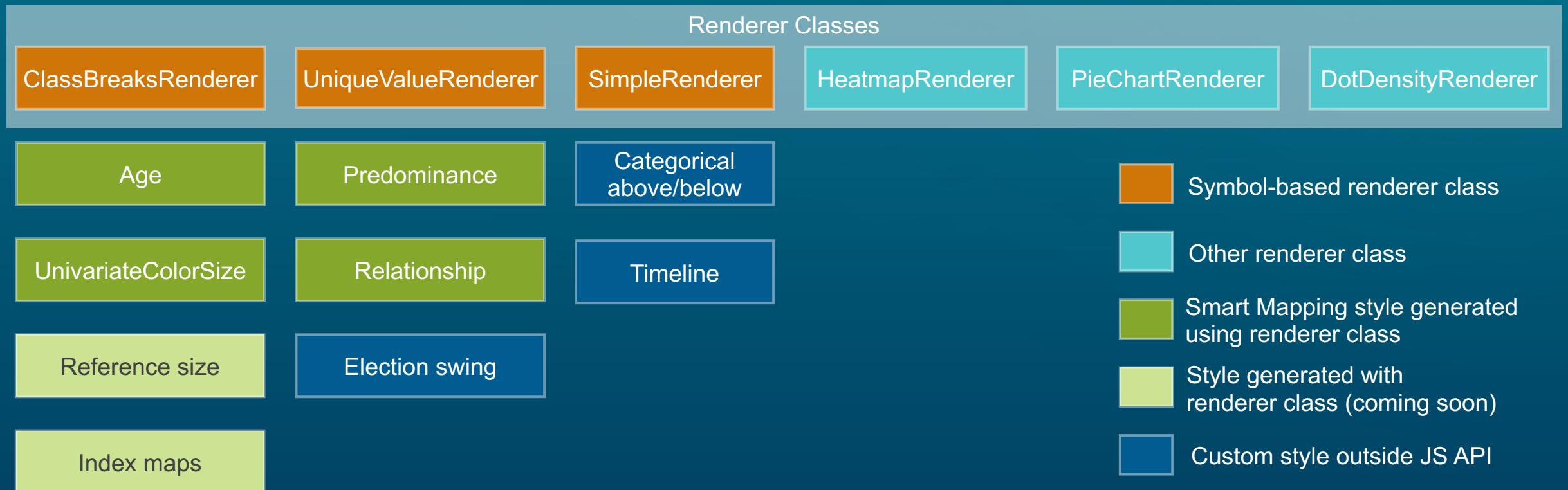
Obesity, diabetes, and inactivity

Relationship

- ▶ High Diabetes; High Obesity
- ▶ High - Medium
- ▶ High - Mild
- ▶ High Diabetes; Low Obesity
- ▶ Medium - High
- ▶ Medium - Medium
- ▶ Medium - Mild

Renderers

Create derivative styles from existing renderer classes



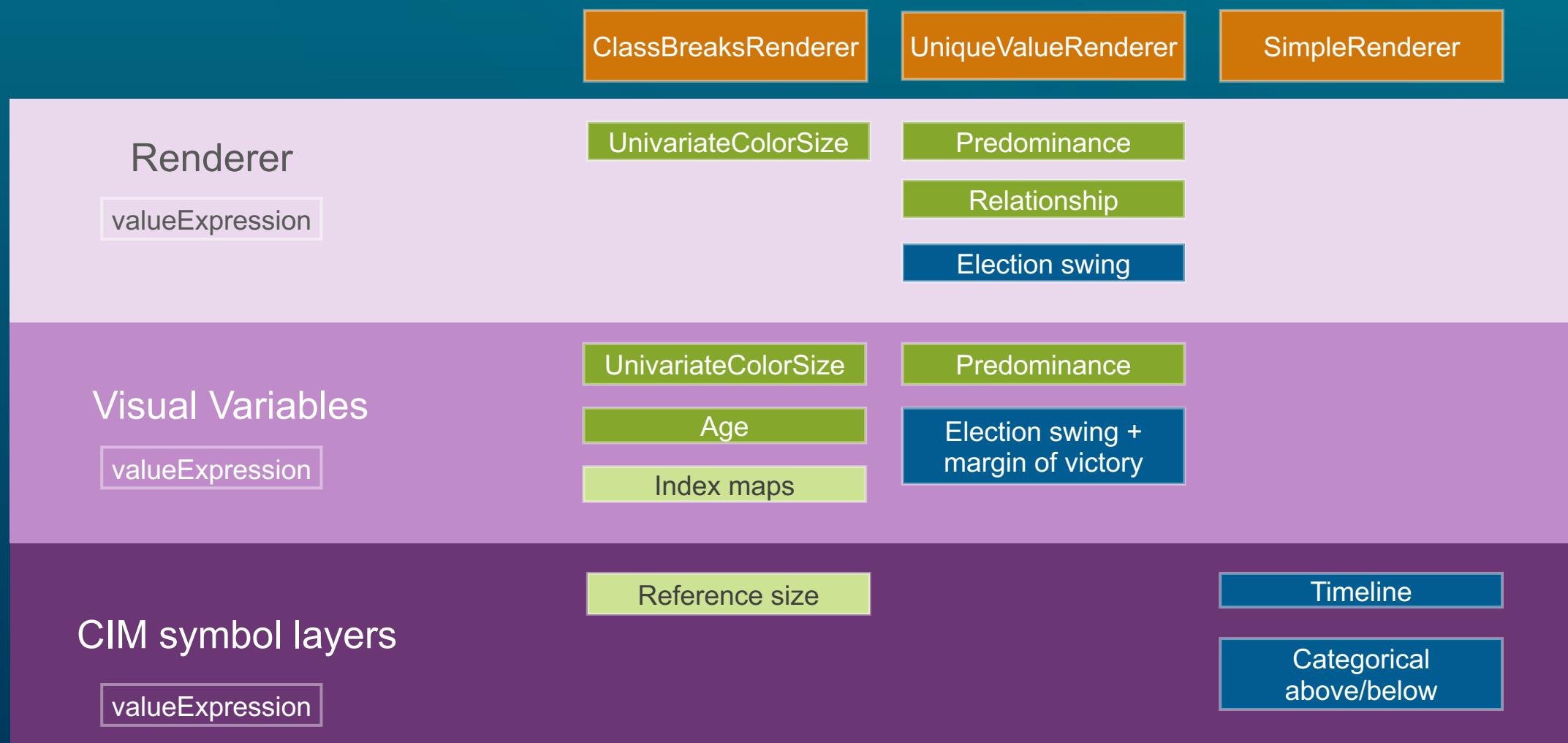
- Simplified UI in Map Viewer
- Legend support

Create your own styles with symbol-dependent renderers

Anywhere Arcade can be defined is an opportunity to be creative

Less complex

More complex

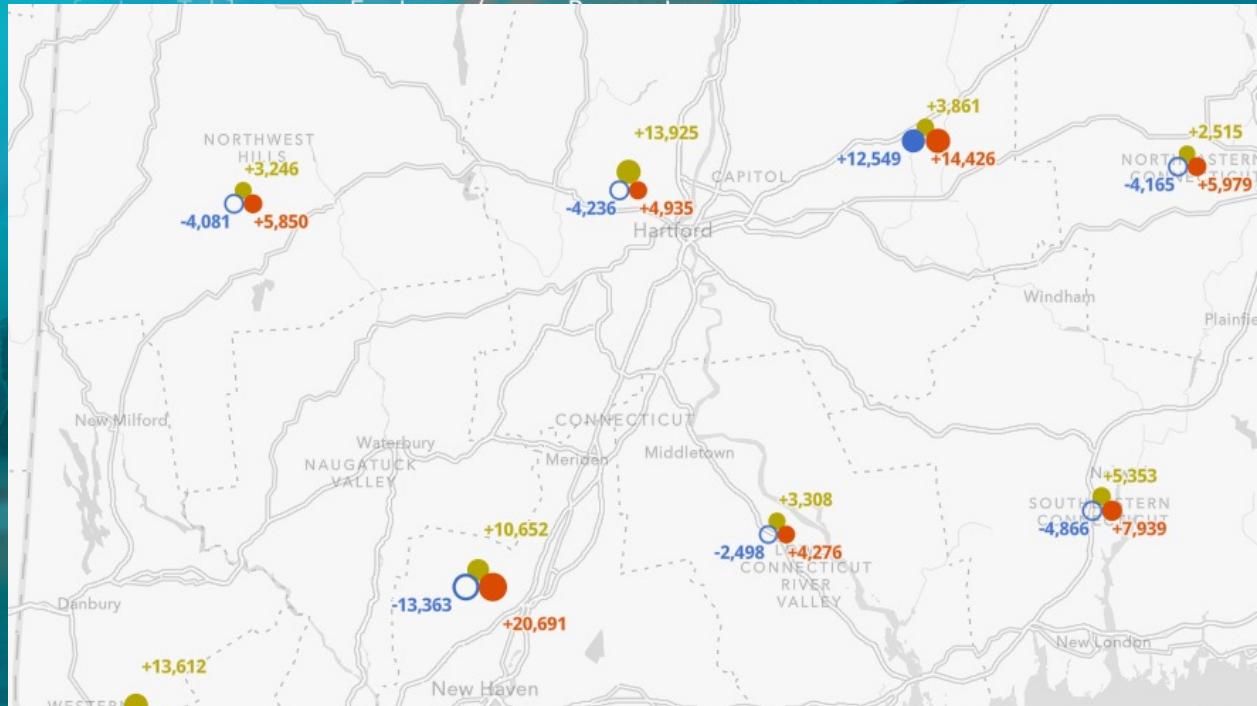


Generate your own style

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

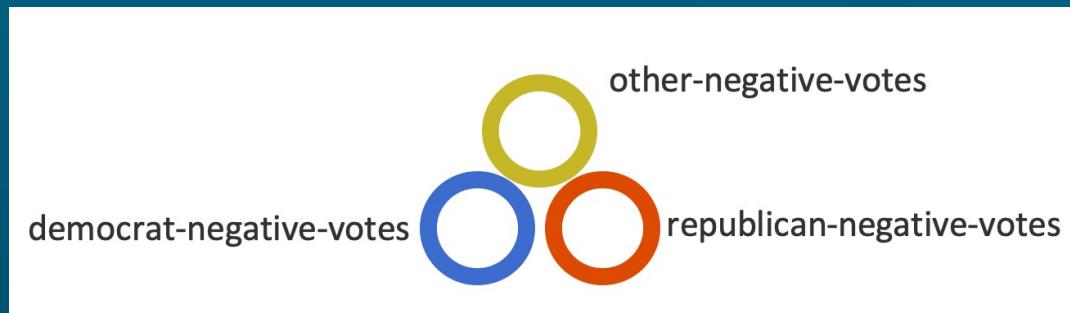
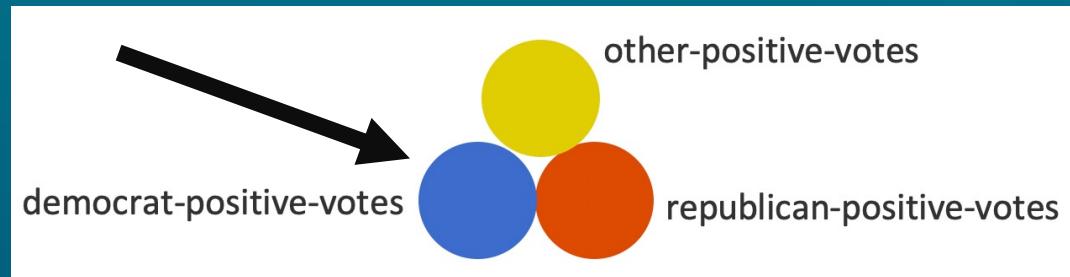
```
    queryParameters =  
    QueryParameters().apply {  
        whereClause = "price > 200"  
    }  
    viewModelScope.launch {
```

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



Categorical above and below

Primitive Overrides



```
const symbolLayer = {  
  type: "CIMVectorMarker",  
  enable: true,  
  primitiveName: "democrat-positive-votes",  
  frame: { xmin: 0.0, ymin: 0.0, xmax: 17.0, ymax: 17.0 },  
  anchorPoint: { x: 0.5, y: 0 },  
  anchorPointUnits: "Relative",  
  markerGraphics: [  
    {  
      type: "CIMMarkerGraphic",  
      geometry: {  
        rings: [  
          [ [ 8.5, 0.2 ],  
            [ 7.06, 0.33 ],  
            [ 5.66, 0.7 ],  
            [ 4.35, 1.31 ],  
            // additional vertices  
            // to complete the circle  
          ]  
        ]  
      }  
    },  
    symbol: {  
      type: "CIMPolygonSymbol",  
      symbolLayers: [  
        {  
          type: "CIMSolidFill",  
          enable: true,  
          color: [60, 108, 204, 255]  
        }  
      ]  
    }  
  ],  
  scaleSymbolsProportionally: true,  
  respectFrame: true  
};
```

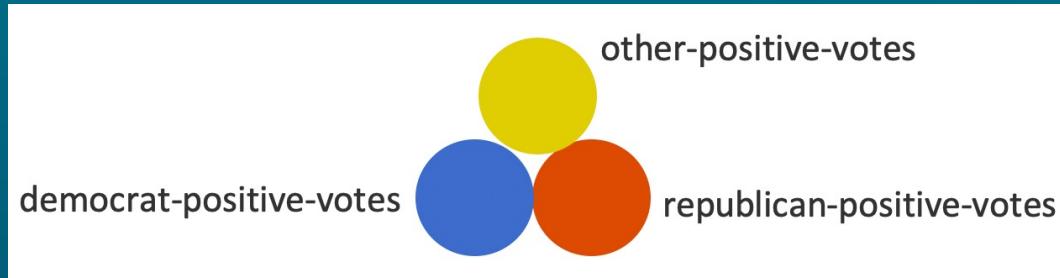
```
const symbolLayer = {  
  type: "CIMVectorMarker",  
  enable: true,  
  primitiveName: "democrat-positive-votes",  
  frame: { xmin: 0.0, ymin: 0.0, xmax: 17.0, ymax: 17.0 },  
  anchorPoint: { x: 0.5, y: 0 },  
  anchorPointUnits: "Relative",  
  markerGraphics: [  
    {  
      type: "CIMMarkerGraphic",  
      geometry: {  
        rings: [  
          [ [ 8.5, 0.2 ],  
            [ 7.06, 0.33 ],  
            [ 5.66, 0.7 ],  
            [ 4.35, 1.31 ],  
            // additional vertices  
            // to complete the circle  
          ]  
        ]  
      }  
    },  
    symbol: {  
      type: "CIMPolygonSymbol",  
      symbolLayers: [  
        {  
          type: "CIMSolidFill",  
          enable: true,  
          color: [60, 108, 204, 255]  
        }  
      ]  
    }  
  ],  
  scaleSymbolsProportionally: true,  
  respectFrame: true  
};
```

Primitive Overrides

```
export const stateChangeRenderer = new SimpleRenderer({  
  symbol: new CIMSymbol({  
    data: {  
      type: 'CIMSymbolReference',  
      symbol: {  
        type: 'CIMPointSymbol',  
        symbolLayers: [ ... ]  
      },  
      primitiveOverrides: [ ... ]  
    }  
  })  
});
```

```
createCircleSymbolLayer({  
  primitiveName: 'other-positive-votes',  
  offsetX: 0,  
  offsetY: 10,  
  color: oColorCIM,  
  donutEnabled: false  
}),  
createCircleSymbolLayer({  
  primitiveName: 'other-negative-votes',  
  offsetX: 0,  
  offsetY: 10,  
  color: oColorCIM,  
  donutEnabled: true  
})  
  
createCircleSymbolLayer({  
  primitiveName: 'democrat-positive-votes',  
  offsetX: -10,  
  offsetY: 0,  
  color: dColorCIM,  
  donutEnabled: false  
}),  
createCircleSymbolLayer({  
  primitiveName: 'democrat-negative-votes',  
  offsetX: -10,  
  offsetY: 0,  
  color: dColorCIM,  
  donutEnabled: true  
})  
  
createCircleSymbolLayer({  
  primitiveName: 'republican-positive-votes',  
  offsetX: 10,  
  offsetY: 0,  
  color: rColorCIM,  
  donutEnabled: false  
}),  
createCircleSymbolLayer({  
  primitiveName: 'republican-negative-votes',  
  offsetX: 10,  
  offsetY: 0,  
  color: rColorCIM,  
  donutEnabled: true  
}),
```

Primitive Overrides



```
var votesNext = $feature.PRSDEM_16;
var votesPrevious = $feature.PRSDEM_12;
var change = votesNext - votesPrevious;
var value = IIF( change > 0, change, 0);

// assign sizes to data values
// interpolate between stops
var sizeFactor = When(
  value >= 500000, 40,
  value >= 100000, 30 + ((40-30)/(500000-100000)) * (value - 100000),
  value >= 50000, 5 + ((30-5)/(100000-50000)) * (value - 1)),
  value >= 10000, 1 + ((5-1)/(50000-10000)) * (value - 0.5)),
  value > 0, 0.5 + ((1-0.5)/(10000) * value),
  0
);
```

```
const democratPositiveOverride2 = {
  type: "CIMPrimitiveOverride",
  primitiveName: "democrat-positive-votes",
  propertyName: "Size",
  valueExpressionInfo: {
    type: "CIMEExpressionInfo",
    title: "Increase in Democrat votes",
    expression: `
      var votesNext = $feature.PRSDEM_16;
      var votesPrevious = $feature.PRSDEM_12;
      var change = votesNext - votesPrevious;
      var value = IIF( change > 0, change, 0);

      // assign sizes to data values
      // interpolate between stops
      var sizeFactor = When(
        value >= 500000, 40,
        value >= 100000, 30 + ((40-30)/(500000-100000)) * (value - 100000),
        value >= 50000, 5 + ((30-5)/(100000-50000)) * (value - 1)),
        value >= 10000, 1 + ((5-1)/(50000-10000)) * (value - 0.5)),
        value > 0, 0.5 + ((1-0.5)/(10000) * value),
        0
      );

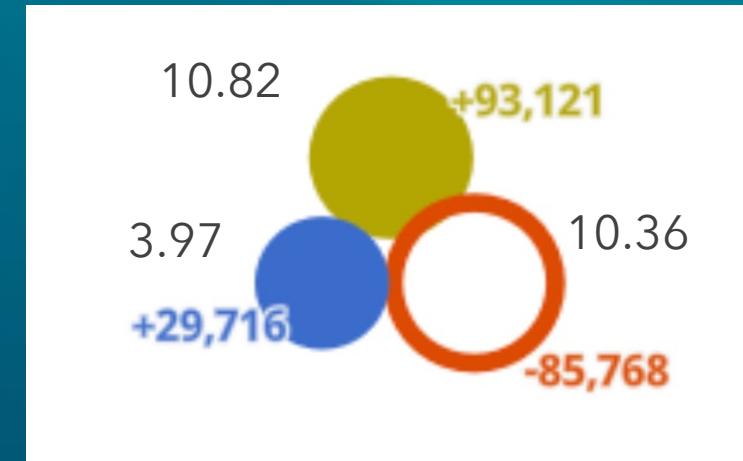
      // adjust size based on scale
      // symbols grow as you zoom in
      // symbols shrink as you zoom out
      var scaleFactorBase = ( ${referenceScale} / $view.scale );
      var scaleFactor = When(
        scaleFactorBase >= 1, 1, // 1
        scaleFactorBase >= 0.5, scaleFactorBase * 1, // 0.6
        scaleFactorBase >= 0.25, scaleFactorBase * 1, // 0.45
        scaleFactorBase >= 0.125, scaleFactorBase * 1, // 0.3125
        scaleFactorBase * 1 // 0.1875
      );

      return sizeFactor * scaleFactor;
    `,
    returnType: "Default"
  }
}
```

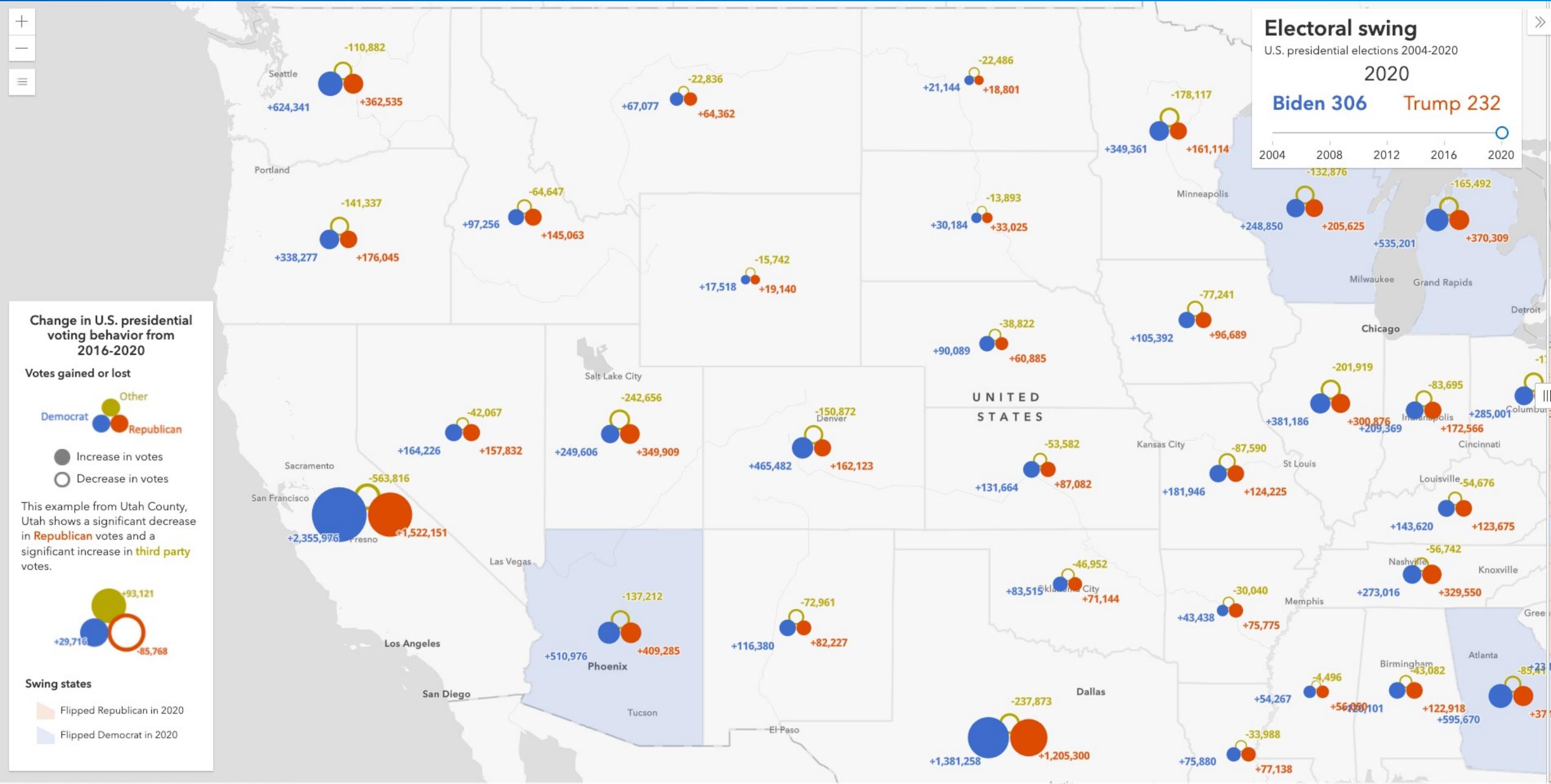
Primitive Overrides

```
var votesNext = $feature.PRS_DEM_16;  
var votesPrevious = $feature.PRS_DEM_12;  
var change = votesNext - votesPrevious;  
var value = IIF( change > 0, change, 0);  
  
// assign sizes to data values  
// interpolate between stops  
var sizeFactor = When(  
    value >= 500000, 40,  
    value >= 100000, 30 + ((40-30)/(500000-100000)) * (value - 100000),  
    value >= 50000, 5 + ((30-5)/(100000-50000)) * (value - 1),  
    value >= 10000, 1 + ((5-1)/(50000-10000)) * (value - 0.5),  
    value > 0, 0.5 + ((1-0.5)/(10000) * value),  
    0  
);  
  
var change = votesNext - votesPrevious;  
var value = IIF( change < 0, Abs(change), 0);
```

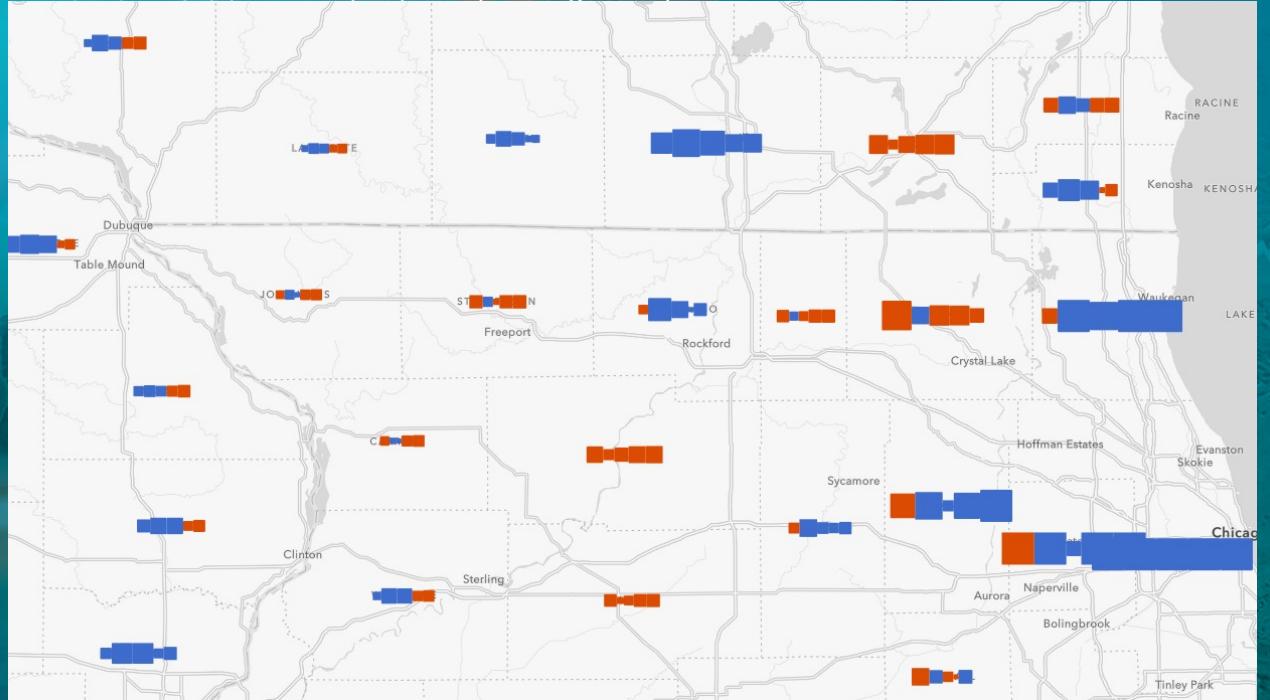
Computed sizes



All other symbol layers are assigned a size of zero, and therefore aren't drawn.



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



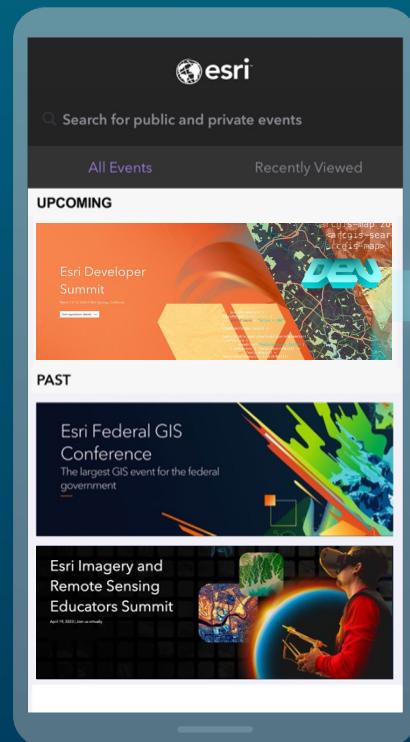
Timeline style

Final thoughts

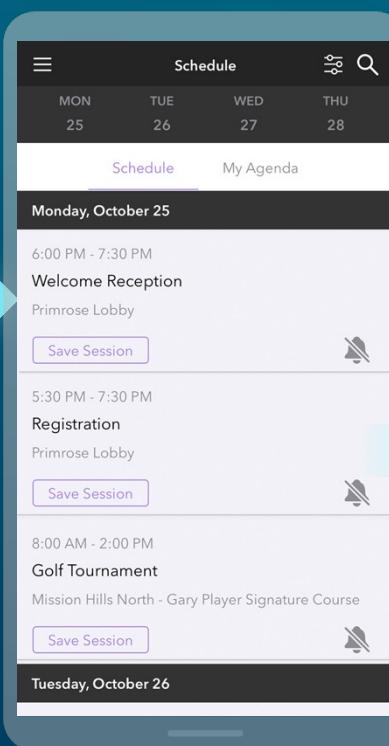
- More complexity isn't always a good thing, but it can be!
- Just because you can, doesn't mean you should.
- Know your audience
 - What works for specialists likely won't work for the general public
- Simplify as much as possible.
- Provide alternative styles/visualizations to show perspective.

Please Share Your Feedback in the App

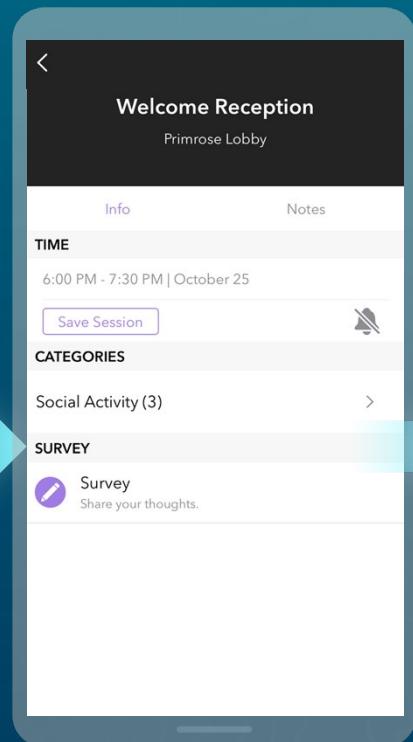
Download the Esri Events app and find your event



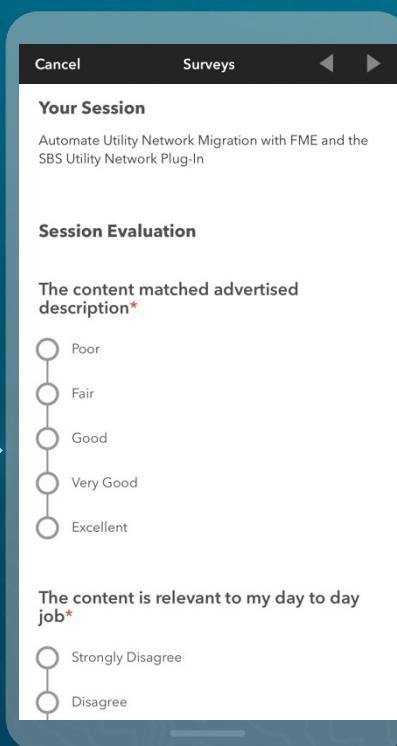
Select the session you attended



Scroll down to "Survey"



Log in to access the survey



Connect With Us On Social

And Join the Conversation Using #EsriDevSummit2024

-  twitter.com/EsriDevs #EsriDevSummit2024
-  twitter.com/EsriDevEvents
-  youtube.com/c/EsriDevelopers
-  links.esri.com/DevVideos
-  github.com/Esri
-  github.com/EsriDevSummit
-  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"
```

```
<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);
  }
});
```

Final thoughts

- More complexity isn't always a good thing
- Just because you can, doesn't mean you should.



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

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

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



```
// 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"
```