



ArcGIS API for JavaScript: Photoshop-style Graphics Effects for Your Layers and Data

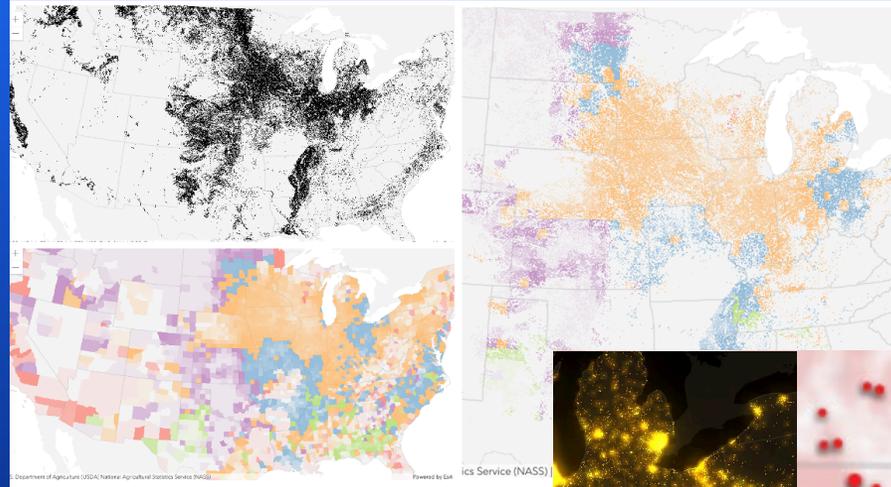
Yann Cabon – ycabon@esri.com

Undral Batsukh – ubatsukh@esri.com

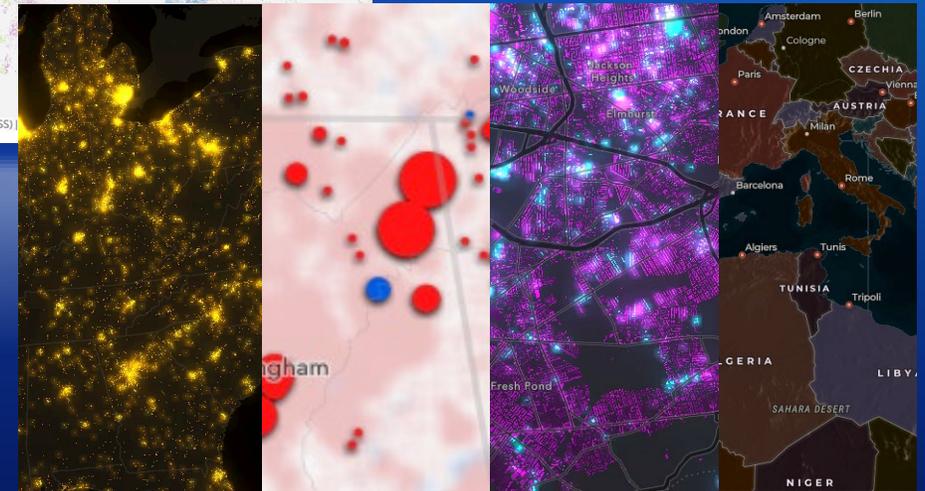
2021 ESRI
DEVELOPER SUMMIT

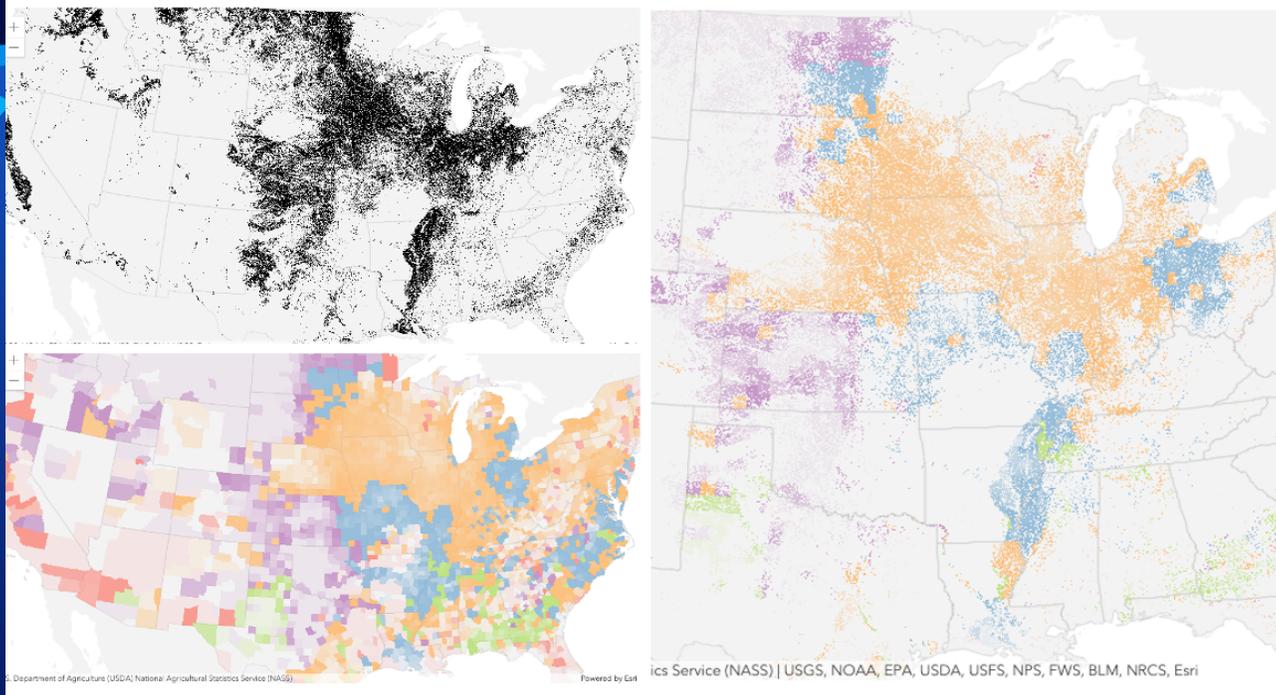
Agenda

- Layer blendMode



- Layer and feature effects





Layer blending

Undral Batsukh

Layer blendMode

- Blend modes are used to blend **layers** together to create an interesting effect in a layer, or even to produce what seems like a new layer.
 - **Top layer** is a layer that has a blend mode applied. All layers underneath the top layer are **background layers**.



VintageLayer.blendMode = "color";



Blend modes categories

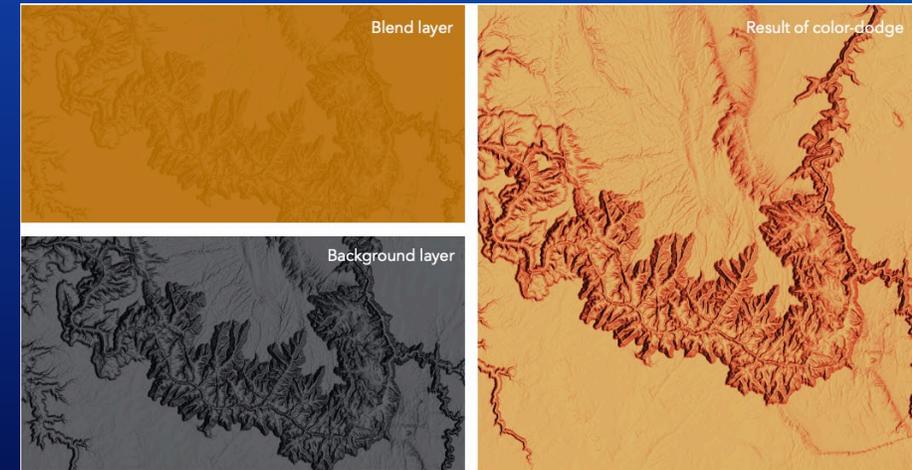
- The **blendMode** is a string property on most layers
- Default blend mode is **normal**
- Lighten blend modes
 - lighten, lighter, screen, plus, color-dodge
- Darken blend modes
 - darken, multiply, color-burn
- Contrast blend modes
 - overlay, soft-light, hard-light, vivid-light
- Component blend modes
 - hue, saturation, luminosity, color
- Composite blend modes
 - destination... and source...
- Invert blend modes
 - difference, excludseion, minus and invert

Lighten blend modes

- Some blend modes create lighter results than all layers.
 - Black becomes transparent, white stays white. Any color that is lighter than black will lighten.
 - Useful when lightening dark colors of the top/blend layer.

Blend mode	Description
lighten	Compares top and background layers and retains the lighter color in the top layer. Colors in the top layer become transparent if they are darker than the overlapping colors in the background layer allowing the background layer to show through completely. Can be thought of as the opposite of <code>darken</code> blend mode.
lighter	Colors in top and background layers are multiplied by their alphas (layer <code>opacity</code> and layer's <code>data opacity</code>). Then the resulting colors are added together. All overlapping midrange colors are lightened in the top layer. The opacity of layer and layer's data will affect the blend result.
plus	Colors in top and background layers are added together. All overlapping midrange colors are lightened in the top layer. This mode is also known as <code>add</code> or <code>linear-dodge</code> .
screen	Inverts colors in top and background, multiplied, and then inverted again. The resulting colors will be lighter than the original color with less contrast. Screen can produce many different levels of brightening depending on the luminosity values of the top layer. Can be thought of as the opposite of the <code>multiply</code> mode.
color-dodge	Creates a brighter effect by decreasing the contrast between the top and background layers, resulting in saturated mid-tones and bright highlights.

```
topLayer.blendMode= "color-dodge"
```

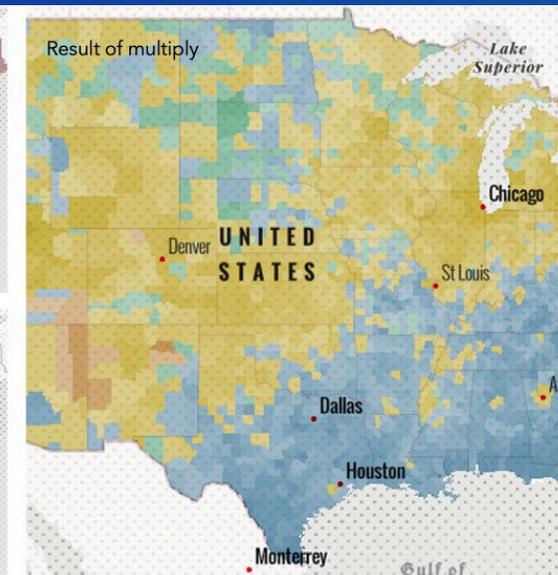
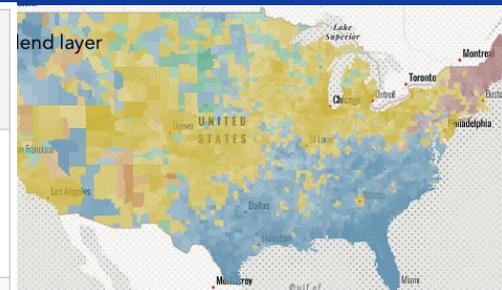


Darken blend modes

- Some blend modes create darker results than all layers.
 - White becomes transparent, black stays black. Any color that is darker than white is going to darken.
 - Can be used to highlight shadows, labels, show contrast, or accentuate an aspect of a map.

```
topLayer.blendMode = "multiply";
```

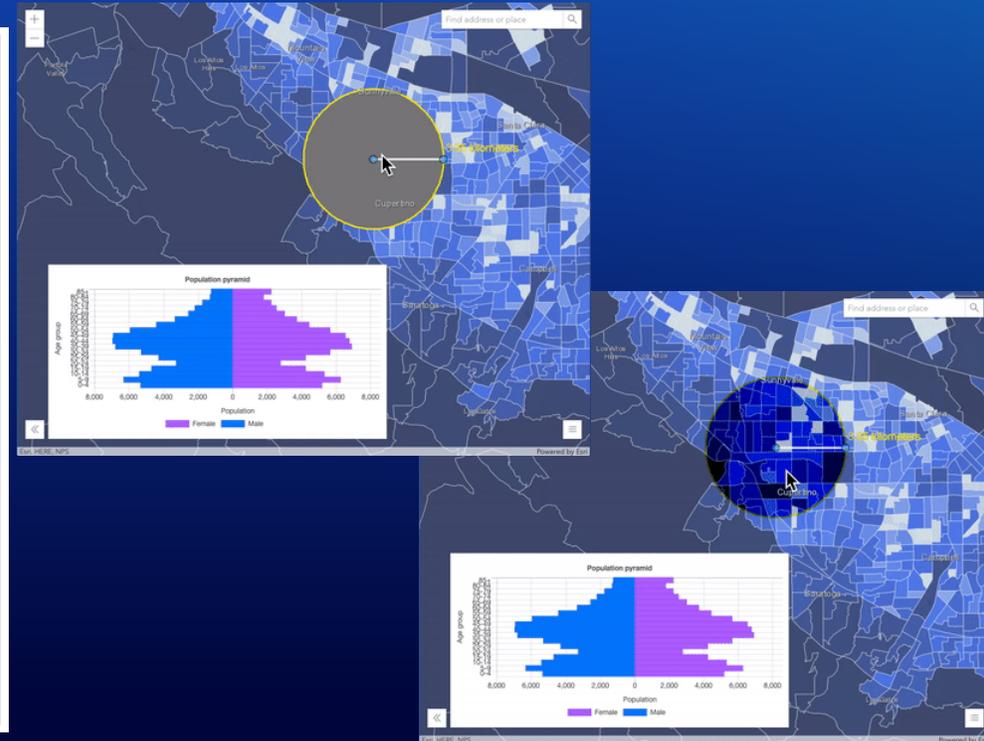
Blend mode	Description
darken	Emphasizes the darkest parts of overlapping layers. Colors in the top layer become transparent if they are lighter than the overlapping colors in the background layer, allowing the background layer to show through completely.
multiply	Emphasizes the darkest parts of overlapping layers by multiplying colors of the top layer and the background layer. Midrange colors from top and background layers are mixed together more evenly.
color-burn	Intensifies the dark areas in all layers. It increases the contrast between top and background layers, by tinting colors in overlapping area towards the top color. To do this it inverts colors of the background layer, divides the result by colors of the top layer, then inverts the results.



Contrast blend modes

- Some blend modes create contrast by using lightening or darkening blend modes to create the blend
 - Lightens the colors lighter than 50% gray, and darkens the colors darker than 50% gray. 50% gray will be transparent.
 - Can be used to increase the contrast and saturation to have more vibrant colors and give a punch to your layers.

Blend mode	Description
overlay	Uses a combination of <code>multiply</code> and <code>screen</code> modes to darken and lighten colors in the top layer with the background layer always shining through. The result is darker color values in the background layer intensify the top layer, while lighter colors in the background layer wash out overlapping areas in the top layer.
soft-light	Applies a half strength <code>screen</code> mode to lighter areas and and half strength <code>multiply</code> mode to darken areas of the top layer. You can think of the <code>soft-light</code> as a softer version of the <code>overlay</code> mode.
hard-light	Multiplies or screens the colors, depending on colors of the top layer. The effect is similar to shining a harsh spotlight on the top layer.
vivid-light	Uses a combination of <code>color-burn</code> or <code>color-dodge</code> by increasing or decreasing the contrast, depending on colors in the top layer.

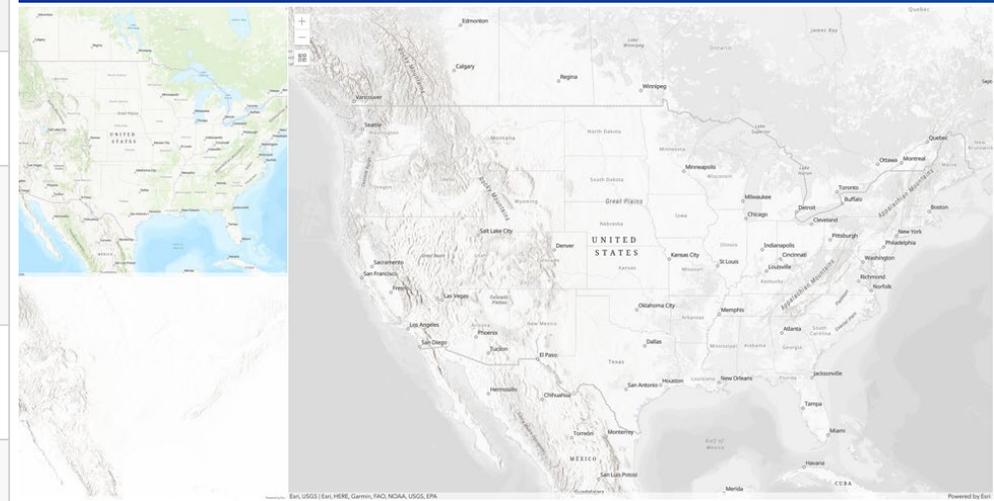


Component blend modes

- Some blend modes use primary color components: hue, saturation and luminosity to blend layers.
 - Use a feature layer with a simple renderer over any layer and set hue, saturation, color, or luminosity blend mode on the layer. With this technique, you create a brand new looking map.

Blend mode	Description
hue	Creates an effect with the hue of the top layer and the luminosity and saturation of the background layer.
saturation	Creates an effect with the saturation of the top layer and the hue and luminosity of the background layer. 50% gray with no saturation in the background layer will not produce any change.
luminosity	Creates effect with the luminosity of the top layer and the hue and saturation of the background layer. Can be thought of as the opposite of <code>color</code> blend mode.
color	Creates an effect with the hue and saturation of the top layer and the luminosity of the background layer. Can be thought of as the opposite of <code>luminosity</code> blend mode.

```
topLayer.blendMode = "luminosity";
```

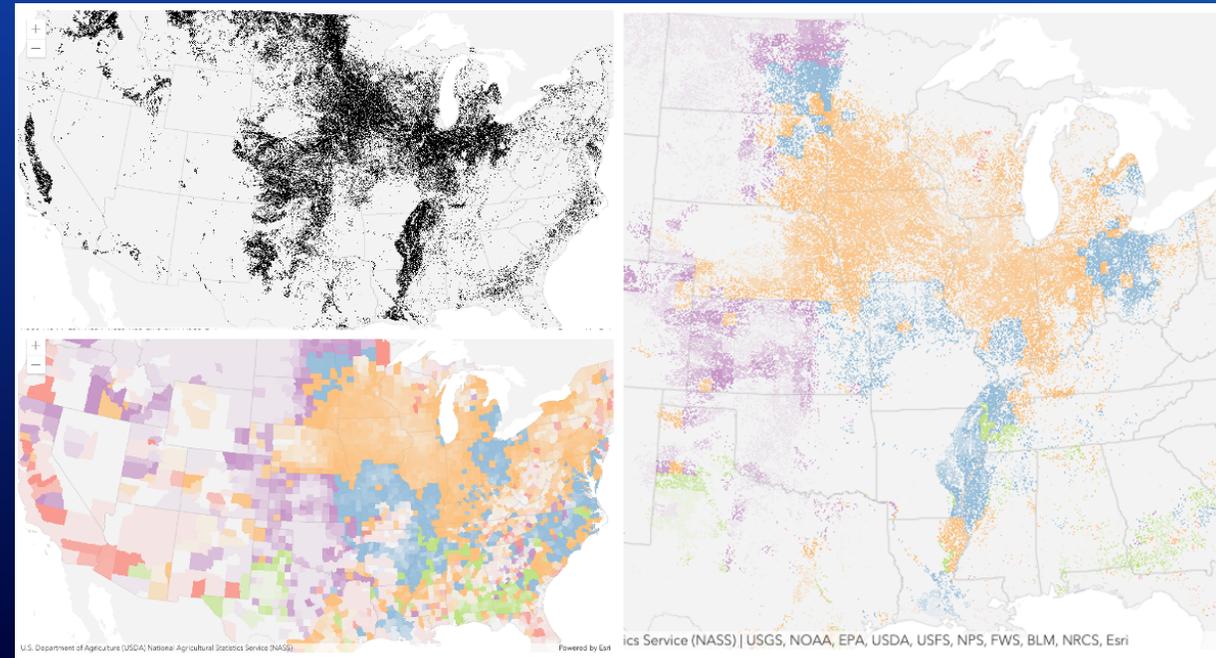


Composite blend modes

- Some blend modes can be used to mask the contents of top, background or both layers.
 - **Destination** modes are used to mask the data of the top layer with the data of the background layer.
 - **Source** modes are used to mask the data of the background layer with the data of the top layer.

Blend mode	Description
destination-over	Destination/background layer covers the top layer. The top layer is drawn underneath the destination layer. You'll see the top layer peek through wherever the background layer is transparent or has no data.
destination-atop	Destination/background layer is drawn only where it overlaps the top layer. The top layer is drawn underneath the background layer. You'll see the top layer peek through wherever the background layer is transparent or has no data.
destination-in	Destination/background layer is drawn only where it overlaps with the top layer. Everything else is made transparent.
destination-out	Destination/background layer is drawn where it doesn't overlap the top layer. Everything else is made transparent.
source-atop	Source/top layer is drawn only where it overlaps the background layer. You will see the background layer peek through where the source layer is transparent or has no data.
source-in	Source/top layer is drawn only where it overlaps with the background layer. Everything else is made transparent.
source-out	Source/top layer is drawn where it doesn't overlap the background layer. Everything else is made transparent.
xor	Top and background layers are made transparent where they overlap. Both layers are drawn normal everywhere else.

```
topLayer.blendMode = "destination-in";
```

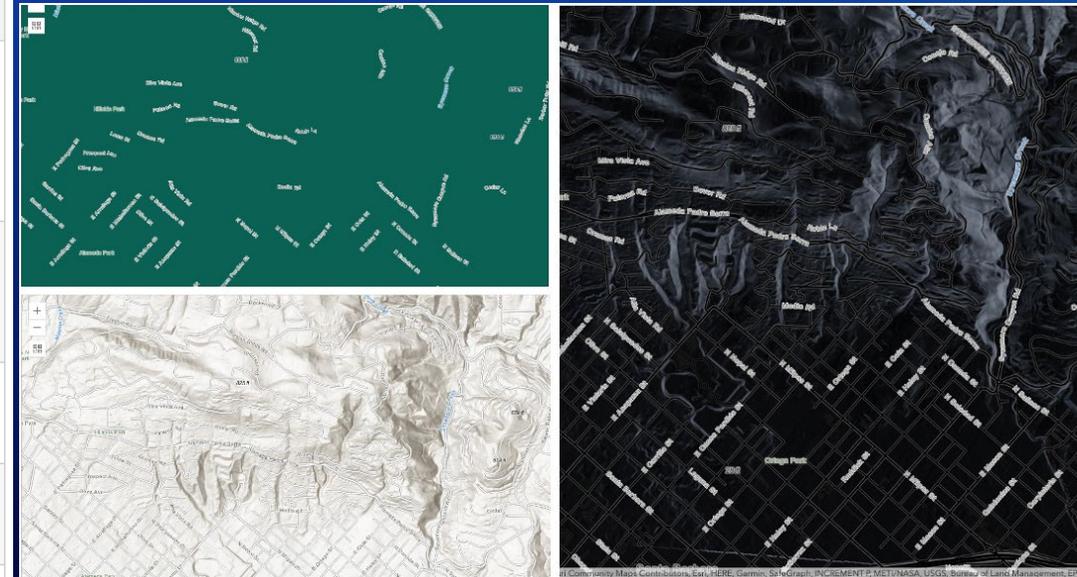


Invert blend modes

- Some blend modes either invert or cancel out colors depending on colors of the background layer.
 - Use **difference** or **exclusion** blend modes on two imagery layers of forest covers to visualize how forest covers changed from one year to another.
 - The **invert** blend mode can be used to turn any light basemap into a dark basemap

Blend mode	Description
difference	Subtracts the darker of the overlapping colors from the lighter color. When two pixels with the same value are subtracted, the result is black. Blending with black produces no change. Blending with white inverts the colors. This blending mode is useful for aligning layers with similar content.
exclusion	Similar to the <code>difference</code> blend mode, except that the resulting image is lighter overall. Overlapping areas with lighter color values are lightened, while darker overlapping color values become transparent.
minus	Subtracts colors of the top layer from colors of the background layer making the blend result darker. In the case of negative values, black is displayed.
invert	Inverts the background colors wherever the top and background layers overlap. The invert blend mode inverts the layer similar to a photographic negative.
reflect	This blend mode creates effects as if you added shiny objects or areas of light in the layer. Black pixels in the background layer are ignored as if they were transparent.

```
topLayer.blendMode = "invert";
```

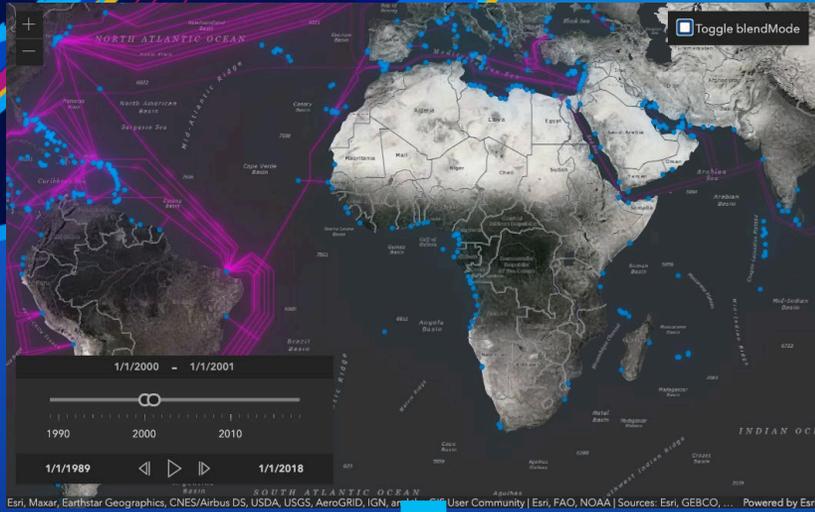


Layer blendMode - tips

- Order of all layers
- Layer opacity
- Opacity of features in layers
- Visibility of layers
- By default, the very bottom layer in a map is drawn on a transparent background. You can change the MapView's background color.

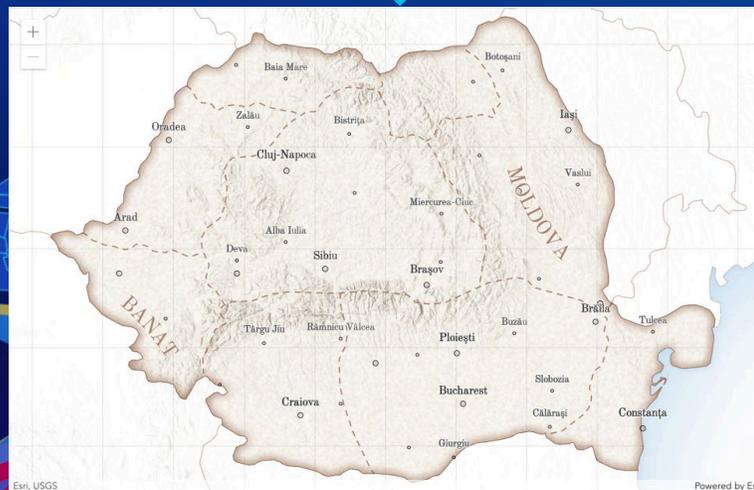
Layer blendMode

- Blend modes can create a variety of very vibrant and intriguing results by blending a layer with the layer(s) below it.
 - All layers underneath the top/blend layer are background layers. Background layer will affect the blend results.
 - <https://www.arcgis.com/apps/mapviewer/index.html?layers=7814c4adee05426caab4c214fe88772a>
- **Examples**
<https://livingatlas.arcgis.com/vessel-traffic/#@=-28.277,48.639,3&time=201701&sublayer=Pleasure>



Submarine cables

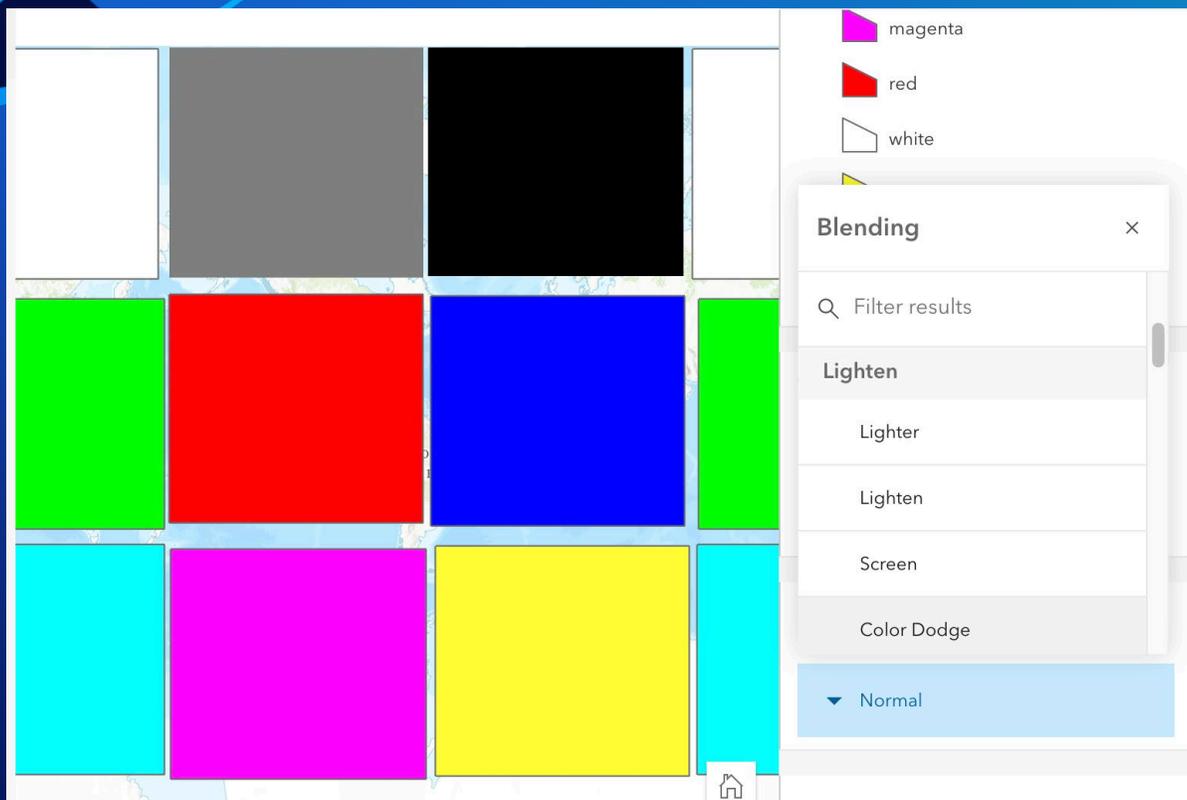
[Demo and source code](#)



Map of Romania

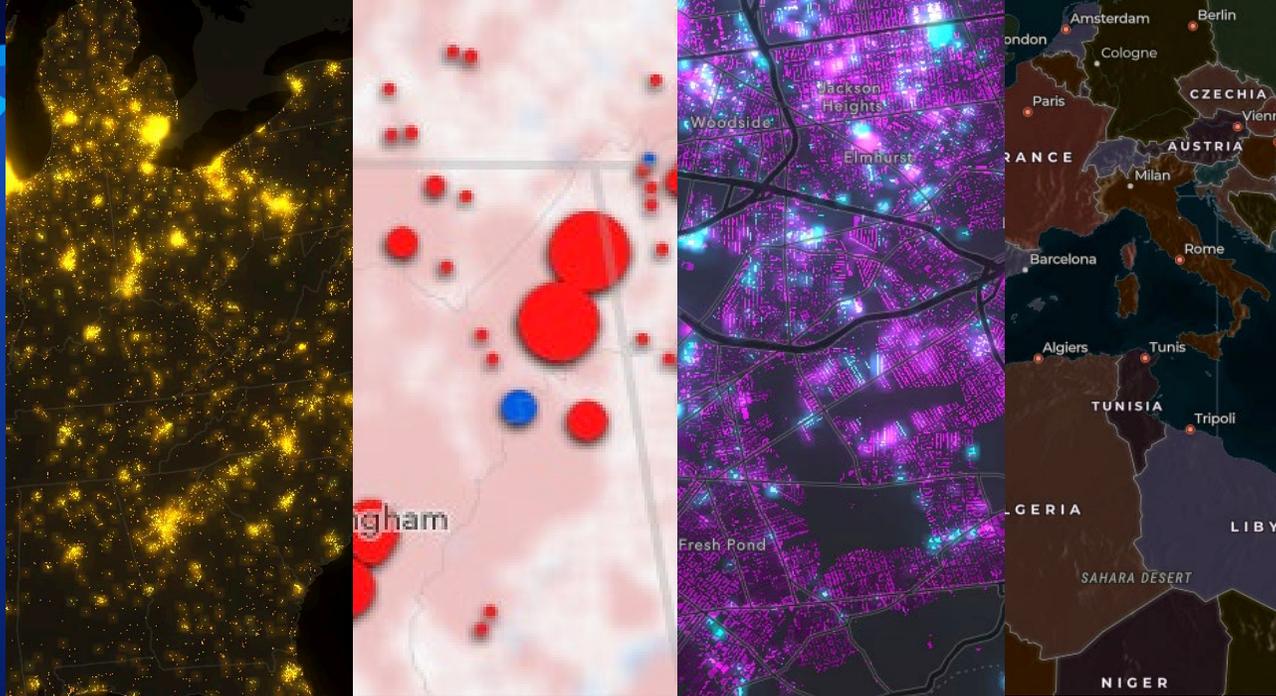
[Source code](#)

[Live app](#)



Test blending in MapViewer Beta

[Use this webmap in MapViewer Beta](#)



Layer and Feature Effects

Yann Cabon

Layer effects

- Various filter functions that can be performed on a layer to achieve different visual effects like in a photo editing apps (Photoshop, GIMP).
- Available on **all layers supported in 2D**.
- How effects work:



Layer effect

Syntax

- An effect is defined like CSS filter
 - If you know about CSS filters you are already proficient with layer filters
- Format
 - A string that describes a list of filter functions with parameters
 - Parameter units: percentage (ratio), lengths, color
- Available filter functions:
 - blur, brightness, contrast, drop-shadow, grayscale, hue-rotate, invert, opacity, saturate, sepia
 - bloom

MDN Web Docs
Technologies | References & Guides | Feedback | Search MDN | Sign in

filter

Web technology for developers > CSS: Cascading Style Sheets > filter

English (US) | Change language

Table of contents

- Syntax
- Functions
- Combining functions
- Formal definition
- Formal syntax
- Examples
- Specifications
- Browser compatibility
- See also

Related Topics

- CSS
- CSS Reference
- Filter Effects

▼ Properties

backdrop-filter

The **filter** CSS property applies graphical effects like blur or color shift to an element. Filters are commonly used to adjust the rendering of images, backgrounds, and borders.

Included in the CSS standard are several functions that achieve predefined effects. You can also reference an SVG filter with a URL to an SVG filter element.

CSS Demo: filter Reset

```
filter: url("../media/examples/shadow.svg#element-id");
filter: blur(5px);
filter: contrast(200%);
filter: grayscale(80%);
filter: hue-rotate(90deg);
filter: drop-shadow(16px 16px 20px red) invert(75%);
```



Syntax

```
layer.effect = "contrast(200%) blur(5px) drop-shadow(1px 2px 5px #000000)";
```

Effects Explorer

- Image correction
 - Adjust brightness, contrast, and saturation
- Color filters
 - Adjust sepia, grayscale, and hue-rotate
- Invert filter
 - Make a light version of a dark basemap
- Blur filter
 - Out of focus
- Drop Shadow
 - Make features stand out
- Bloom filter
 - Make features shine bright



brightness 123

contrast 140

saturate 46

```
layer.effect = 'brightness(123%) contrast(140%) saturate(46%)';
```

Copy to clipboard

Effects Explorer

[Demo](#)

[Source code](#)

Layer effect

- First feedback from cartographers about layer effects:
 - “WOW! Super exciting!”
 - “I want to change the effect at different scales”



Layer effect

Scale dependency

- Effects are interpolated when user zooms
- Format
 - An array of objects with a scale and an effect value
 - Effect list must be compatible between scales

```
layer.effect = [  
  { scale: 80000, value: 'opacity(50%)' },  
  { scale: 20000, value: 'bloom(1 0px 65%)' }  
];
```

Interpolation not possible
between opacity and bloom

```
layer.effect = [  
  { scale: 80000, value: 'bloom(0 0px 65%) opacity(50%)' },  
  { scale: 20000, value: 'bloom(1 0px 65%) opacity(100%)' }  
];
```

```
layer.effect = [  
  { scale: 80000, value: 'bloom(0 0px 65%) opacity(50%)' },  
  { scale: 20000, value: 'bloom(1 0px 65%)' }  
];
```

Scale Dependent Effects

Bloom filter without scale dependency
Too bright at large scale

Scale Dependency
Full control

```
layer.effect = [
  { scale: 80000, value: 'bloom(2 0px 65%)' },
  { scale: 20000, value: 'bloom(2 0px 65%)' }
];
```

Copy to clipboard

NYC: OpenData, State of New Jersey: Esri, HERE, Garmin, SafeGraph, METI/NASA, USGS, EPA, NPS, USDA
Powered by Esri

Scale Dependent Effects

[Demo](#)

[Source code](#)

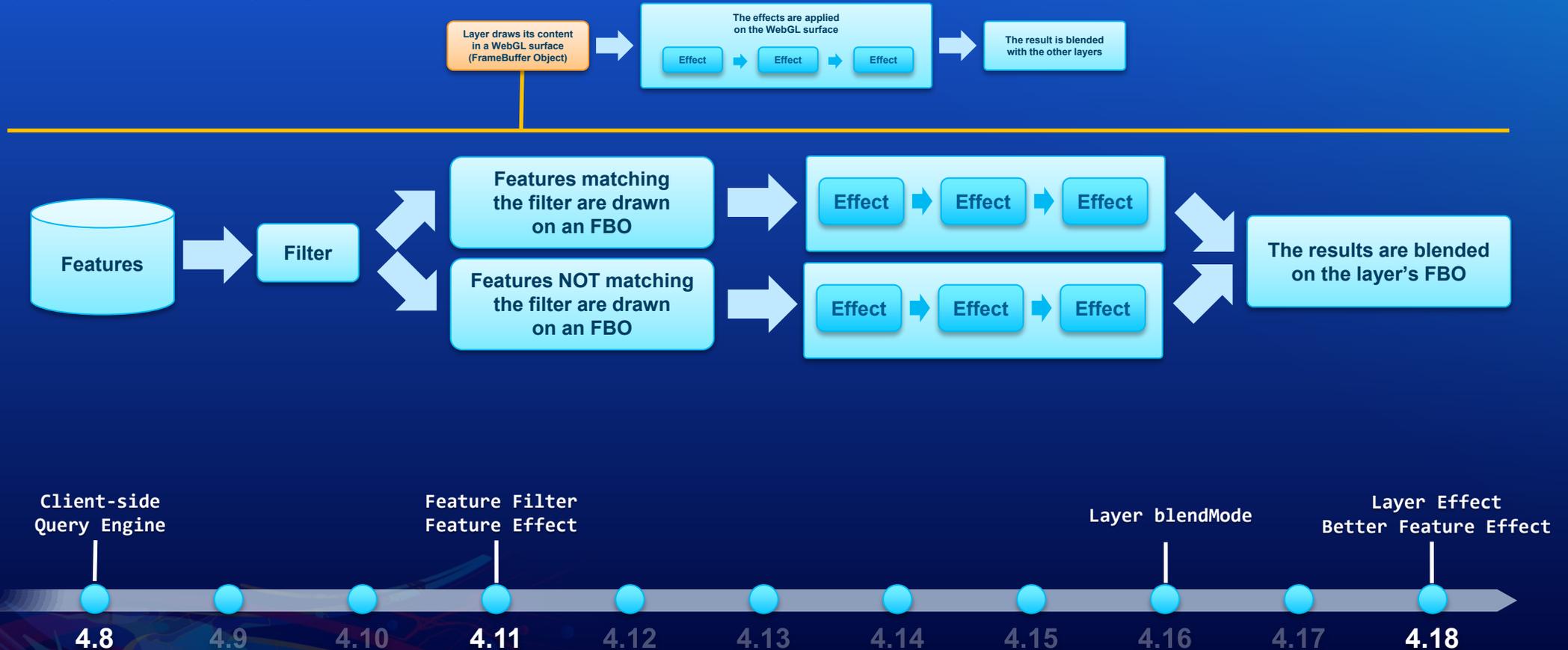
Feature Effects

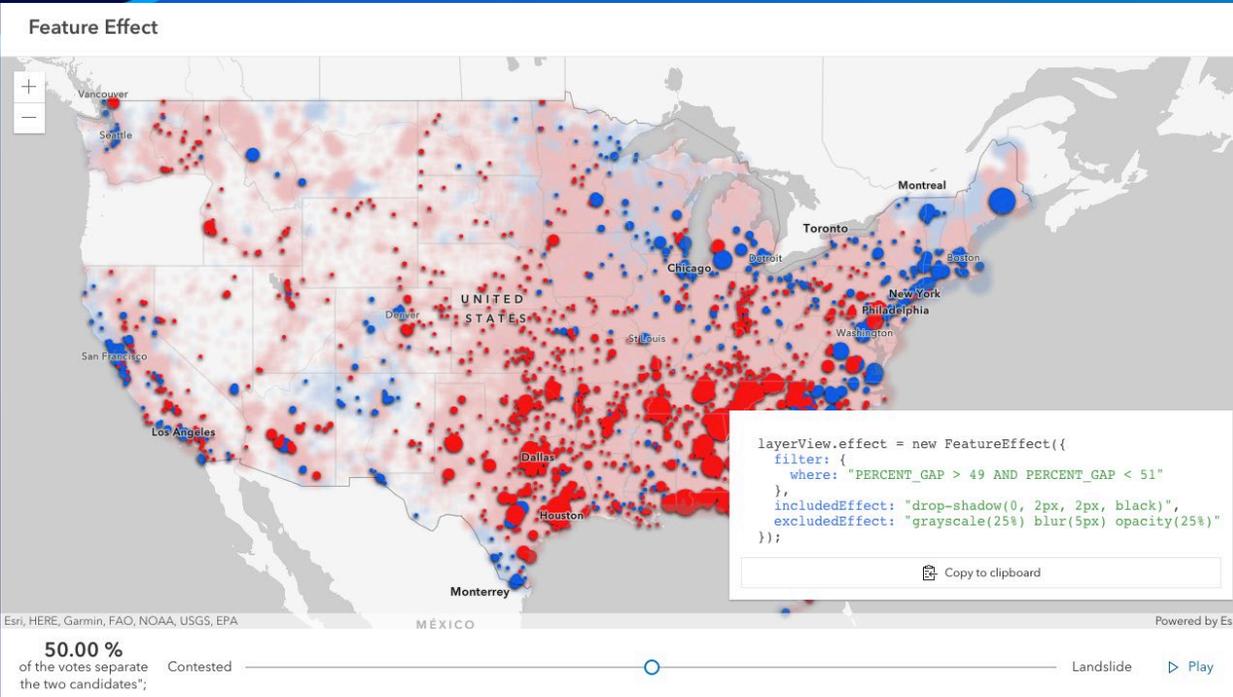
- The effect allows for the selection of features via a **filter**, and an **includedEffect** and **excludedEffect** are applied to those features that respectively pass or fail the filter requirements.
- Available on **feature layerviews in 2D**: FeatureLayerView, GeoJSONLayerView, CSVLayerView, StreamLayerView

```
layerView.effect = new FeatureEffect({
  filter: {
    where: "PERCENT_GAP > 49 AND PERCENT_GAP < 51"
  },
  includedEffect: "drop-shadow(0, 2px, 2px, black)",
  excludedEffect: "grayscale(25%) blur(5px) opacity(25%)"
});
```

Feature Effects

- How feature effects work:

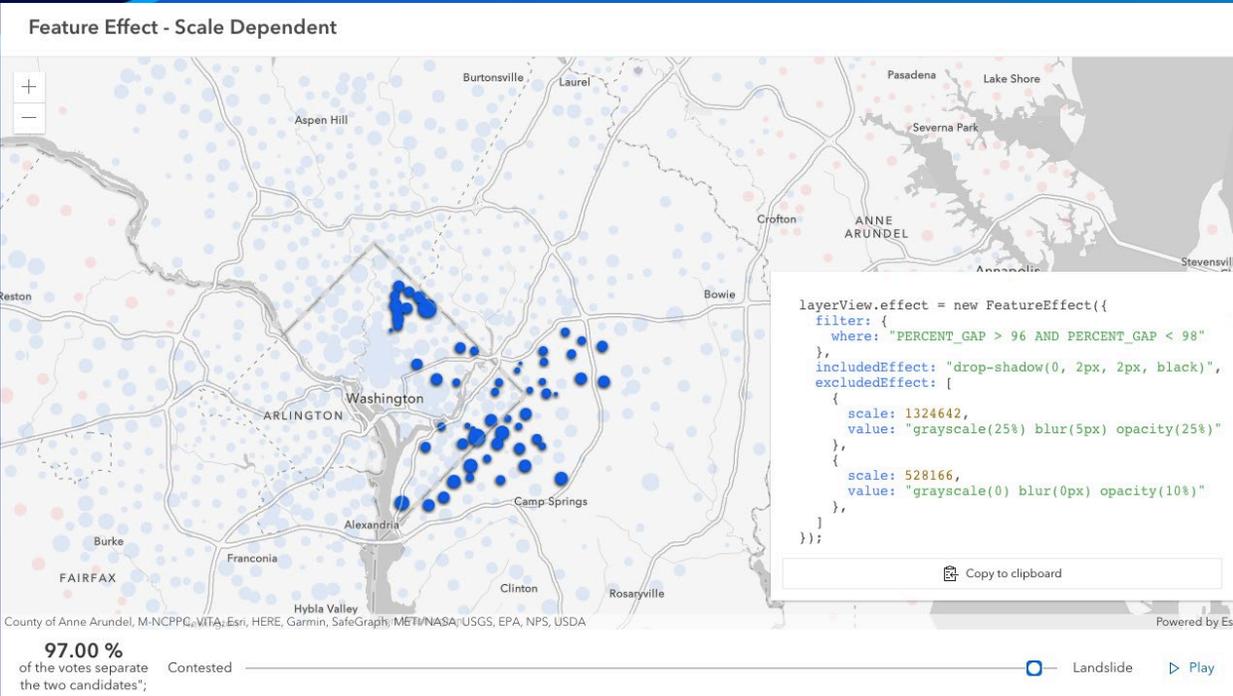




Feature Effects

[Demo](#)

[Source code](#)



Feature Effects Scale Dependent

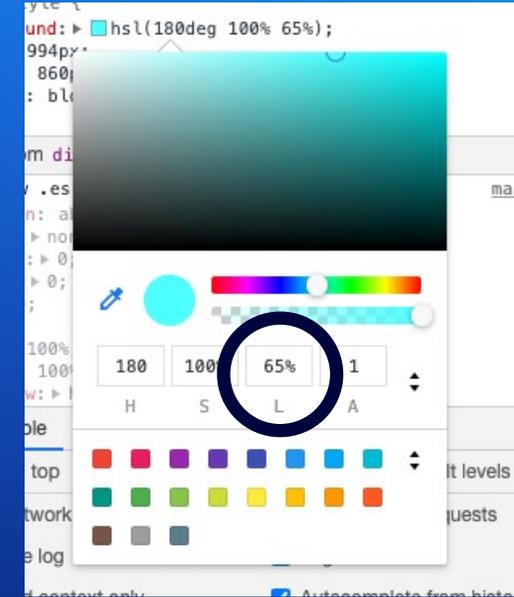
[Demo](#)

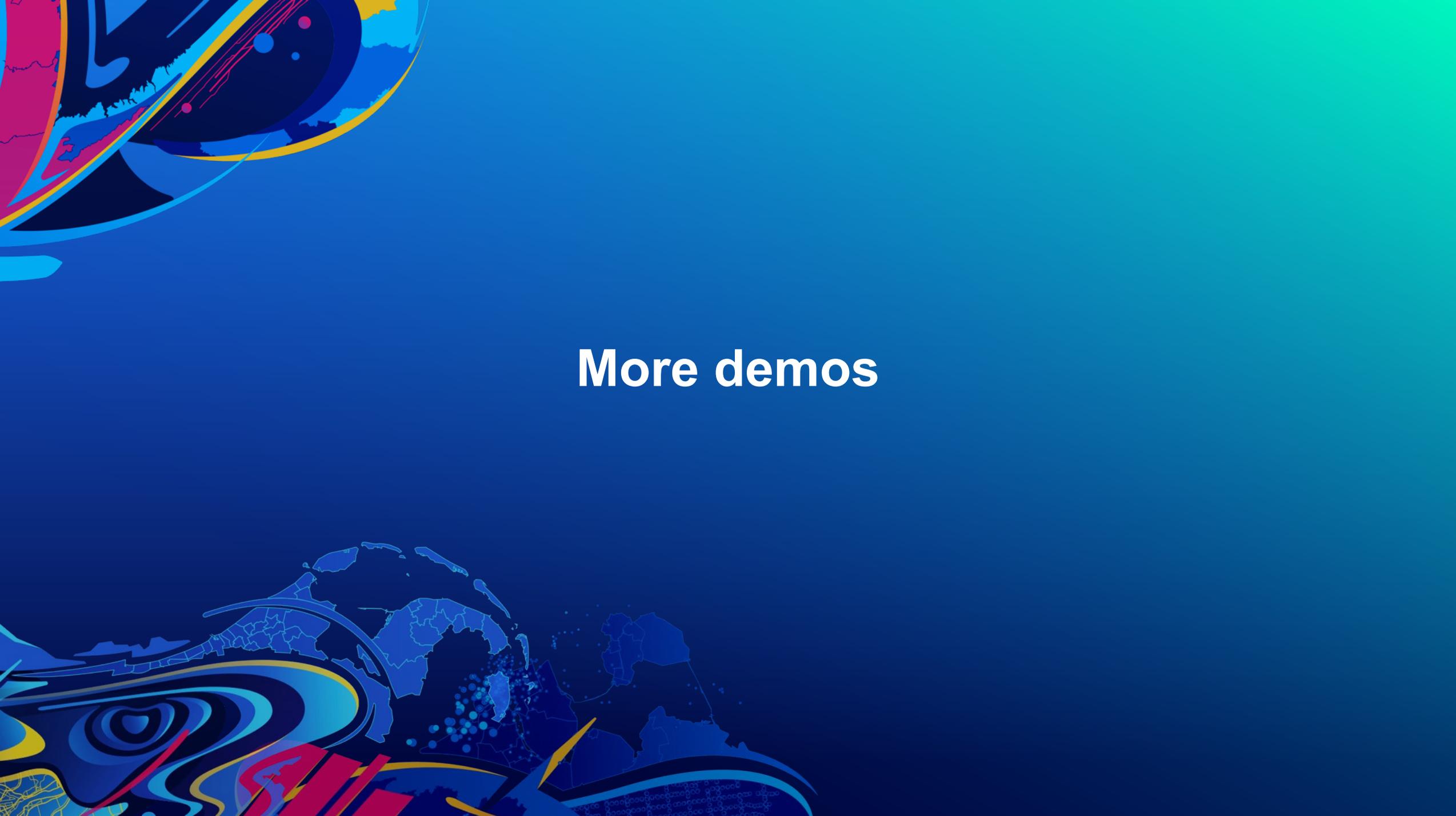
[Source code](#)

Layer and Feature effect

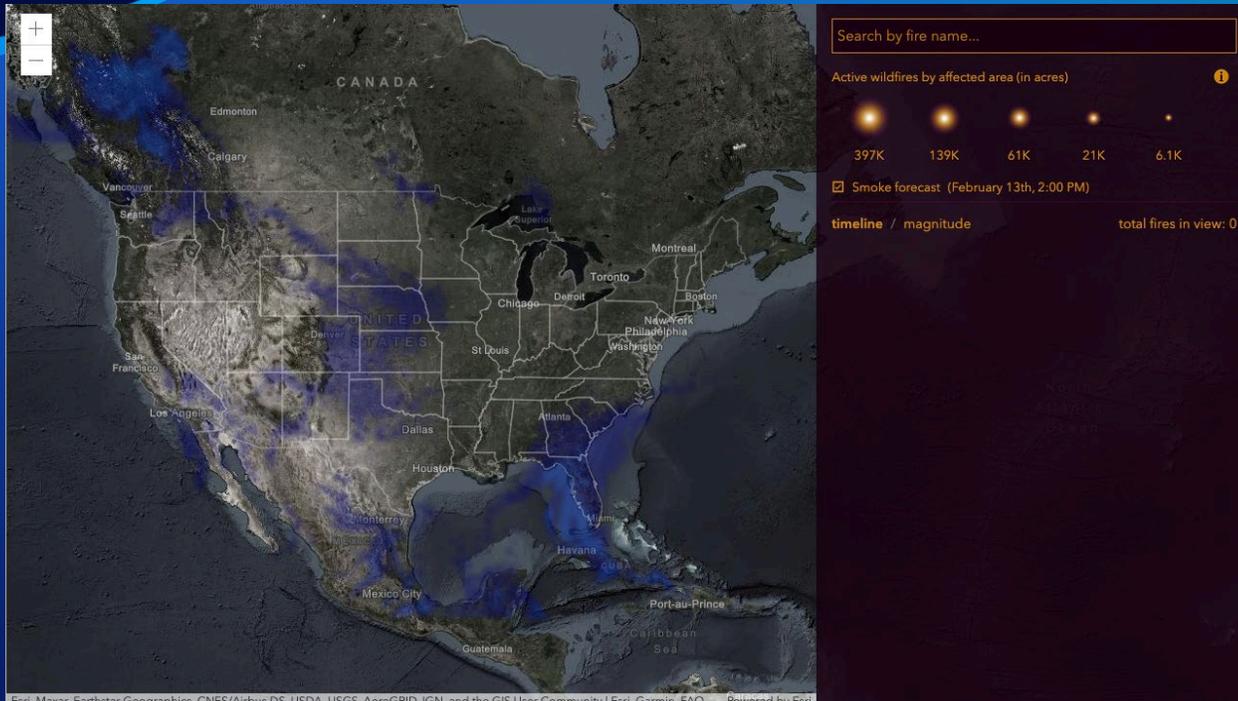
Tips

- To find the right bloom threshold, use the luminosity component of the color in HSL.
- The order of the filter functions in the list.
- Not supported by 3D, Print, and Legend



The image features a vibrant blue gradient background. In the top-left and bottom-left corners, there are abstract, stylized graphic elements. These elements consist of overlapping shapes in shades of blue, red, and yellow, with some areas appearing to have a metallic or glossy texture. The text "More demos" is centered in the middle of the image in a clean, white, sans-serif font.

More demos



Wildfire map

[Demo](#)

[Source code](#)

Invert Basemap

Invert

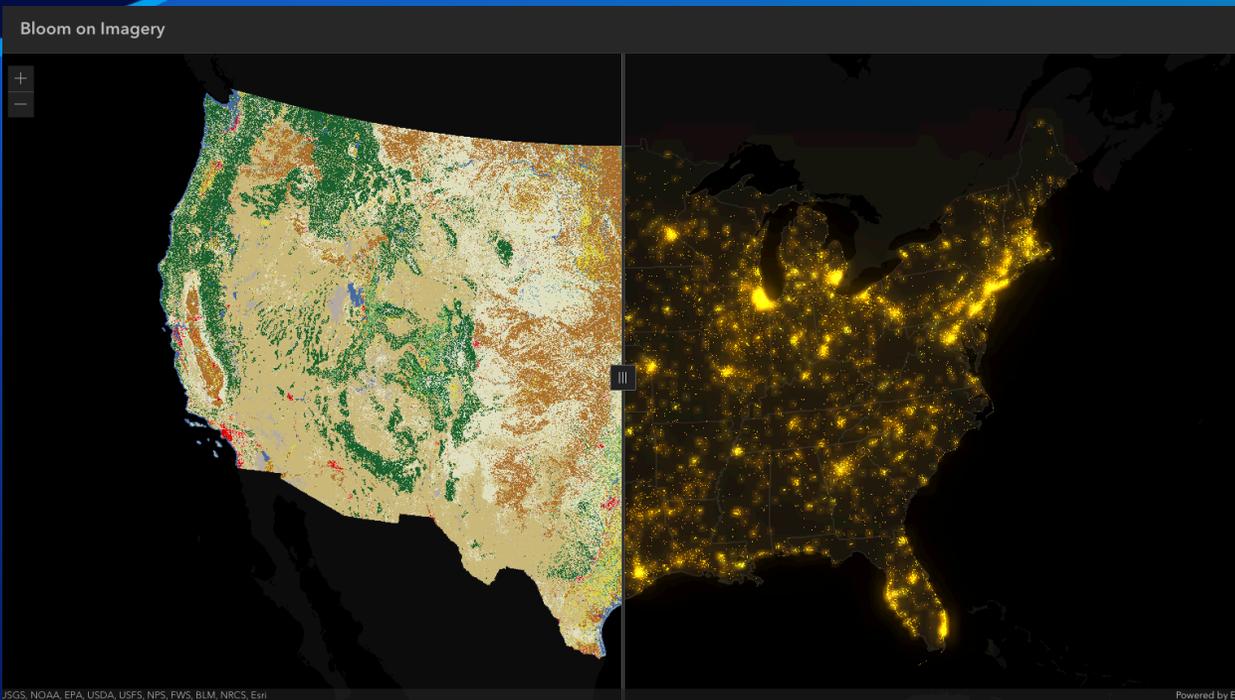
- National Geographic Style Map
- OpenStreetMap
- Charted Territory Map
- Community Map
- Navigation (Dark Mode)
- Newspaper Map
- Human Geography Map
- Human Geography Dark Map
- Modern Antique Map
- Mid-Century Map



Invert Basemap

[Demo](#)

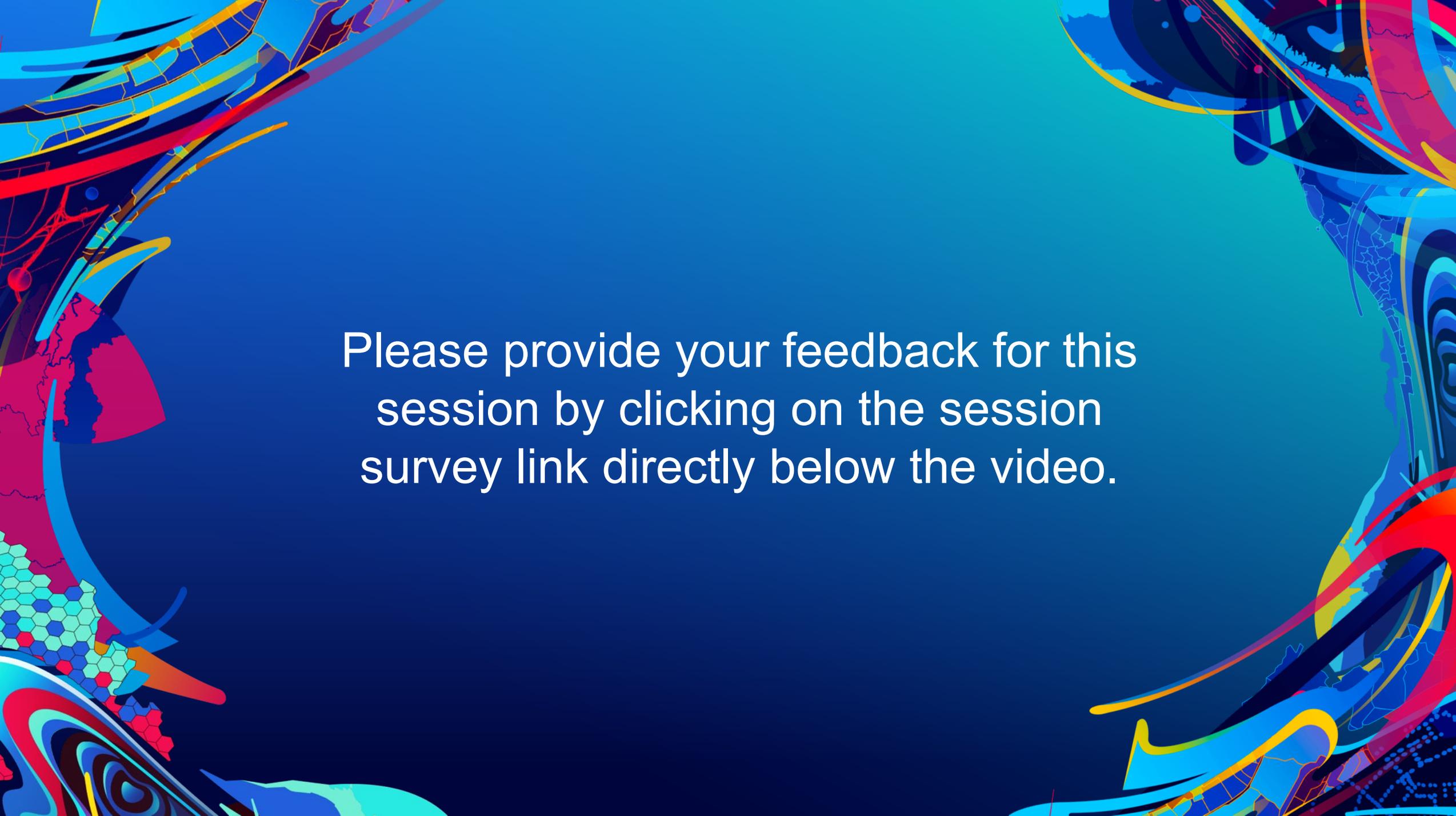
[Source code](#)



Bloom on Imagery

[Demo](#)

[Source code](#)



Please provide your feedback for this session by clicking on the session survey link directly below the video.