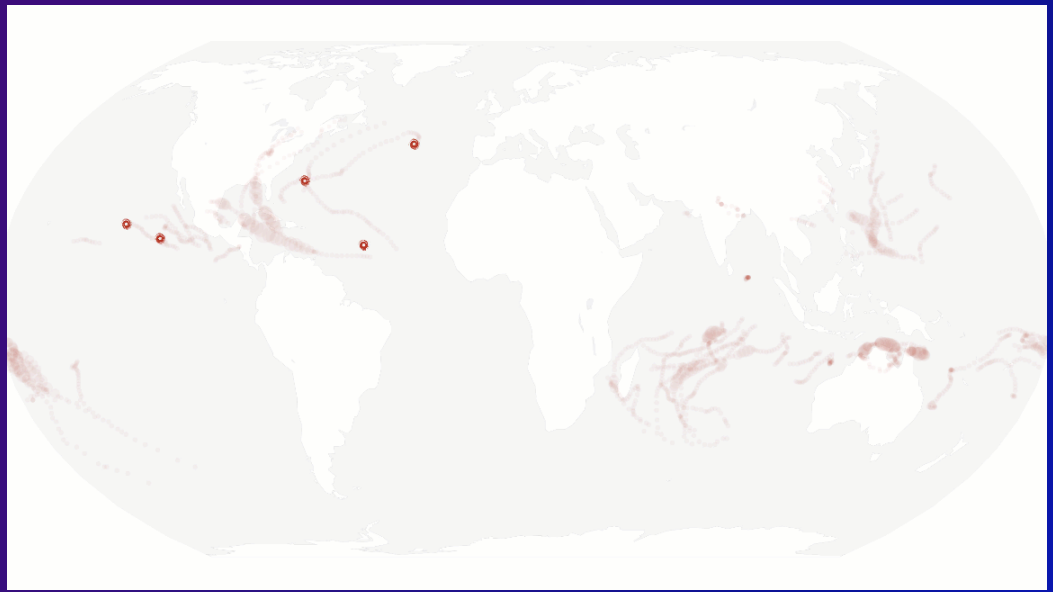
ArcGIS API for JavaScript: Data-Driven Animations

Jeremy Bartley

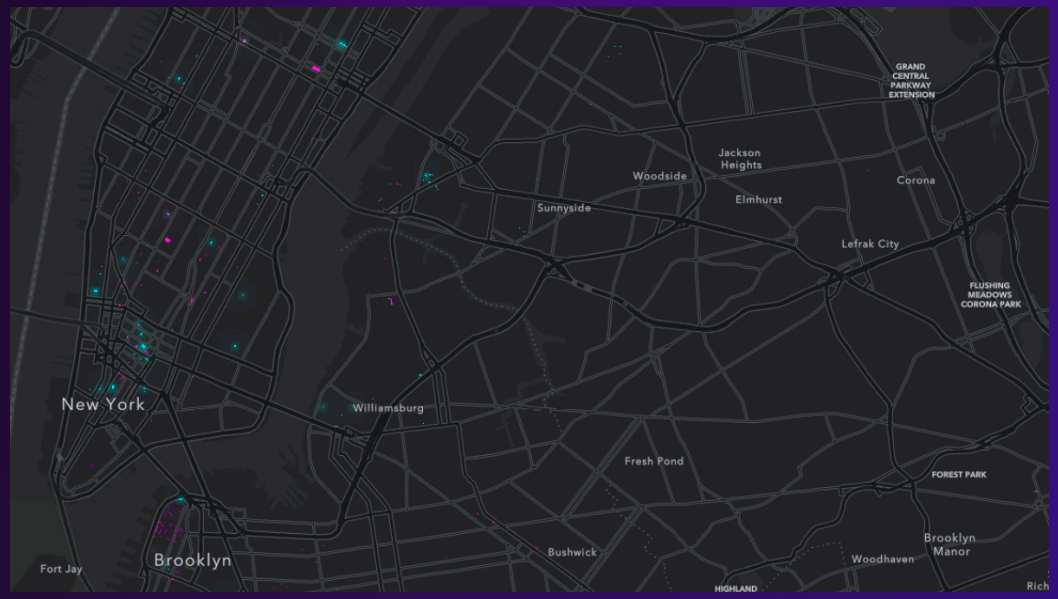
2021 ESRI USER CONFERENCE

Different ways to animate data

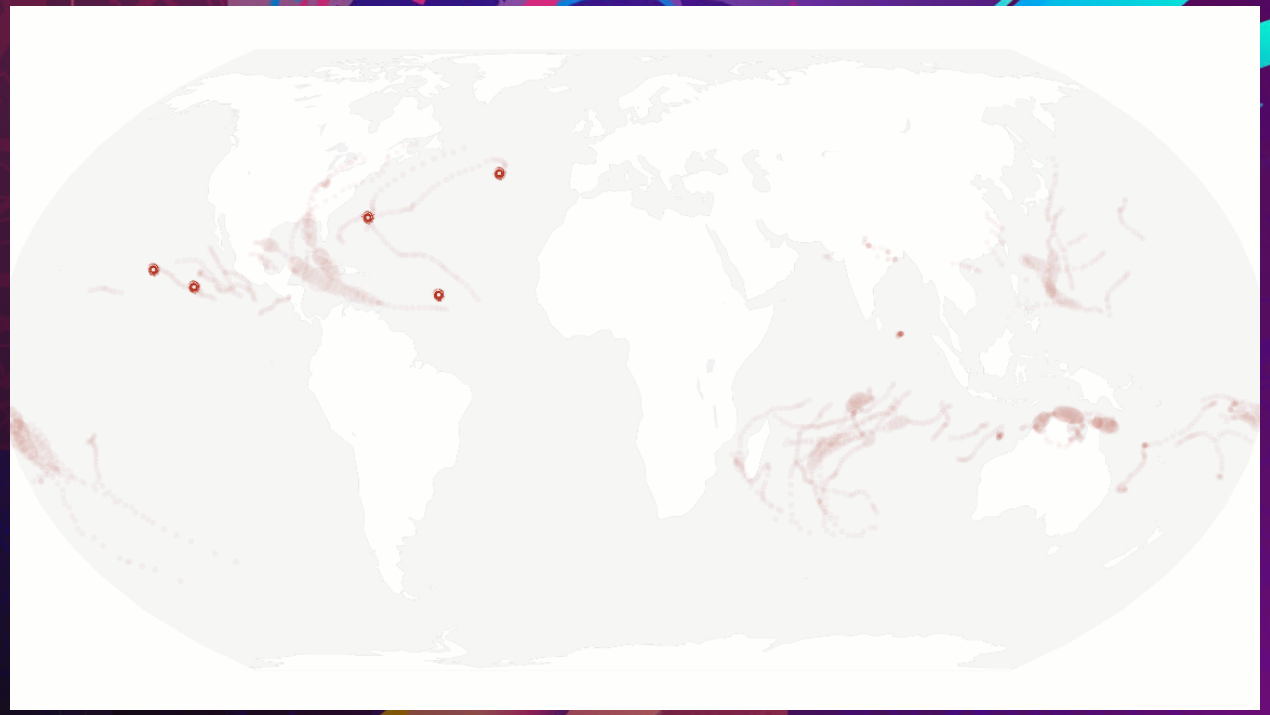
Data filter animation



Visual variable animation



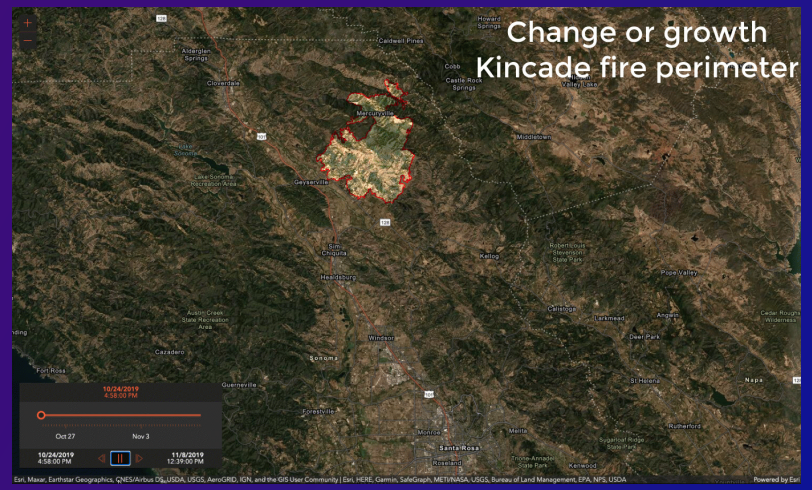
Data variable animation



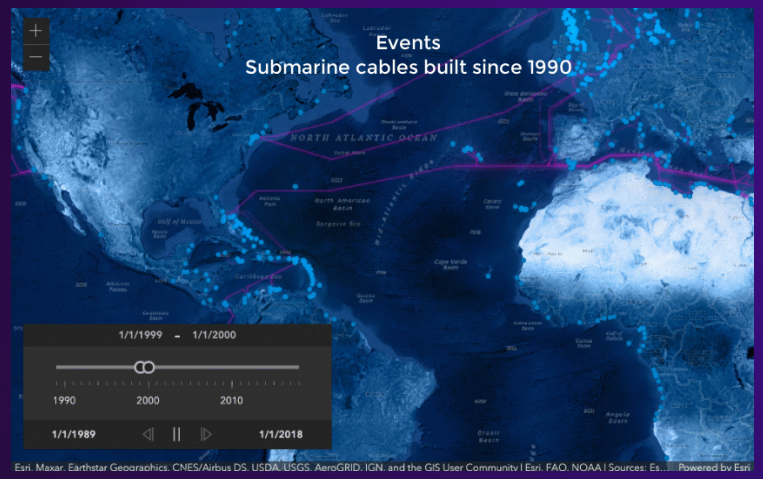
Data filter animation

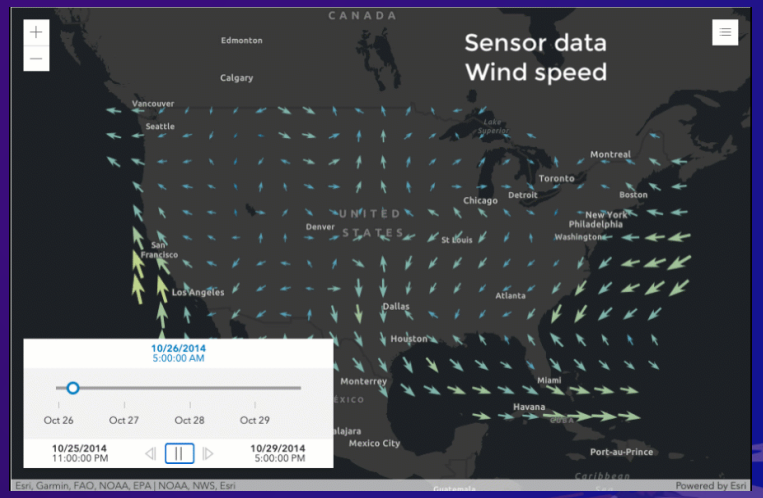
Animating temporal data

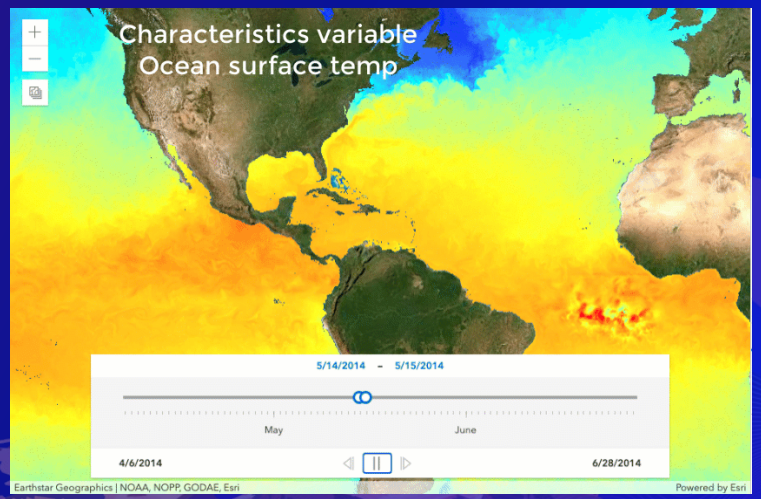
- Detect change or pattern over time
- Location/geometry changes over time



- Stationary objects



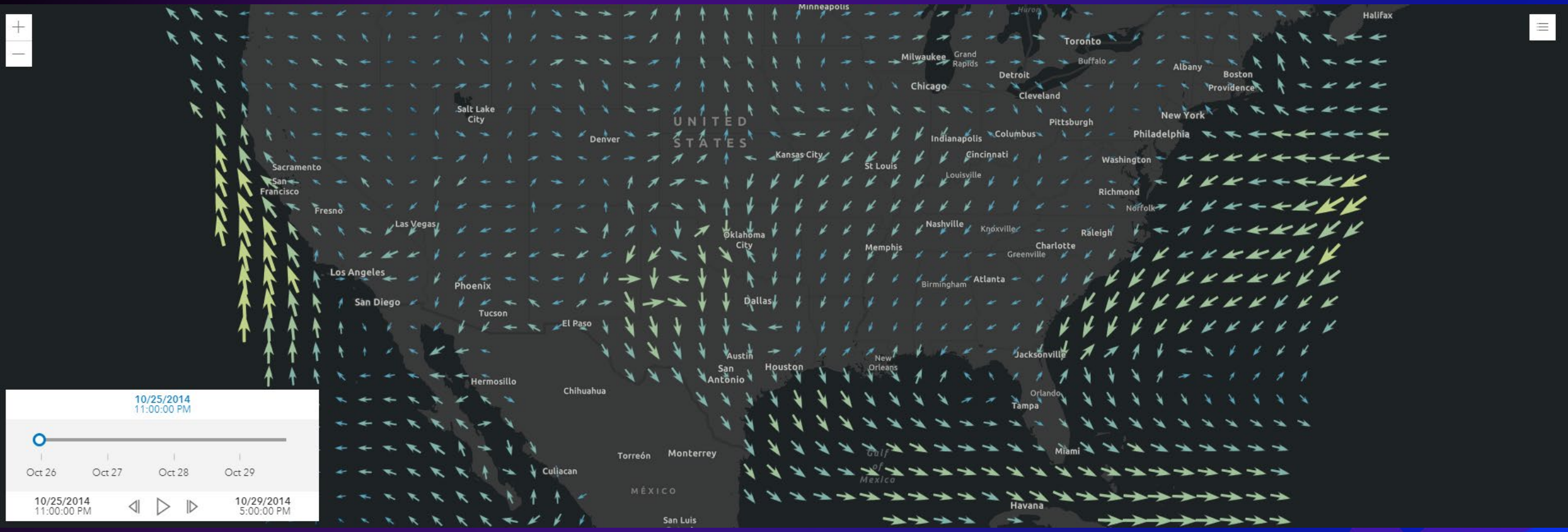




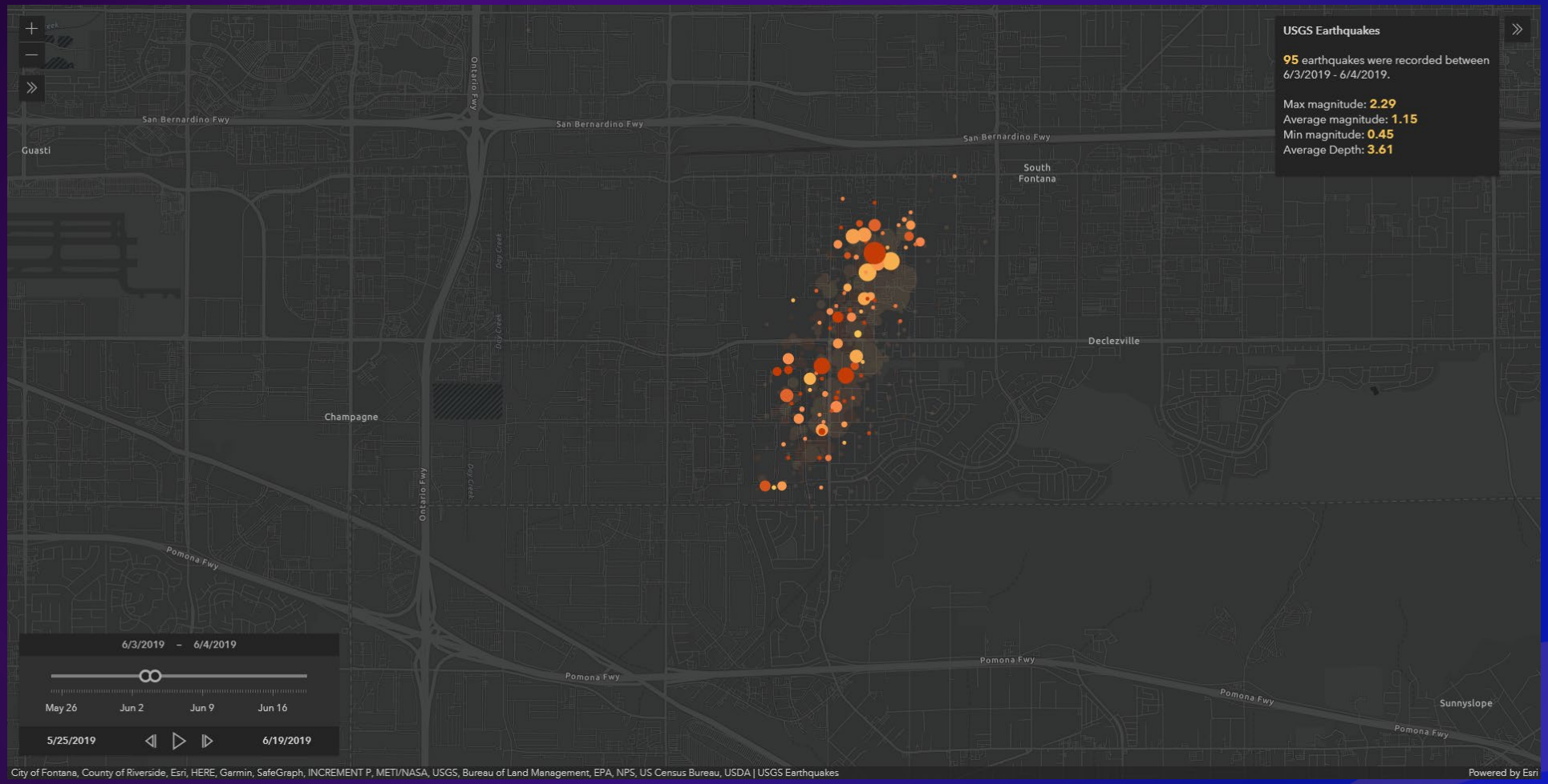
Data filter animation

- Filter data by attributes, geometry or time extent
  - Use layer.definitionExpression, layerView.effect or layerView.filter
    - Based on attributes or geometry
  - Use TimeSlider to animate temporal data
    - Requires service timeInfo or date field
- Renderer is fixed
- Filter controls whether the feature is visible or *emphasized*
  - When working with a MapImageLayer or ImageryLayer map is generated on the server
  - When working with a client side layer (FeatureLayer, GeoJSON, etc) all features are requested from the server and feature visibility is controlled on the client
    - Results in fastest display

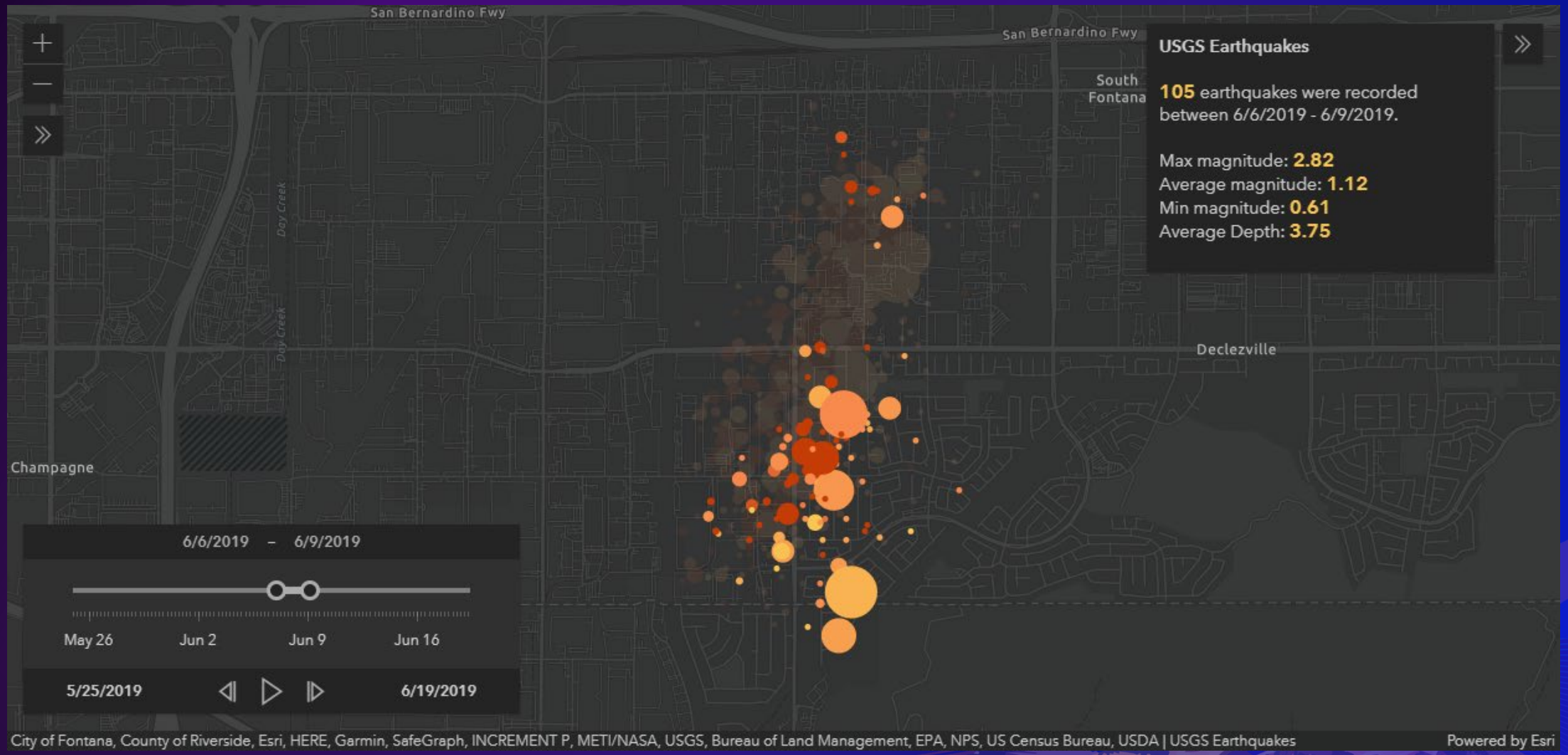
Animating features on the server – One image request at a time



Animating features on the client – faster performance by leveraging browser capabilities

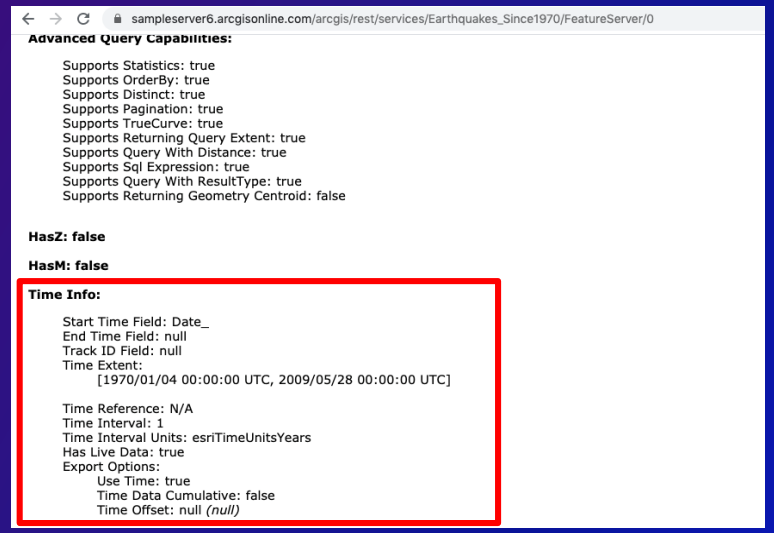


More than just controlling visibility. Can use Effects to emphasize or de-emphasize features that are in or outside of time slider window

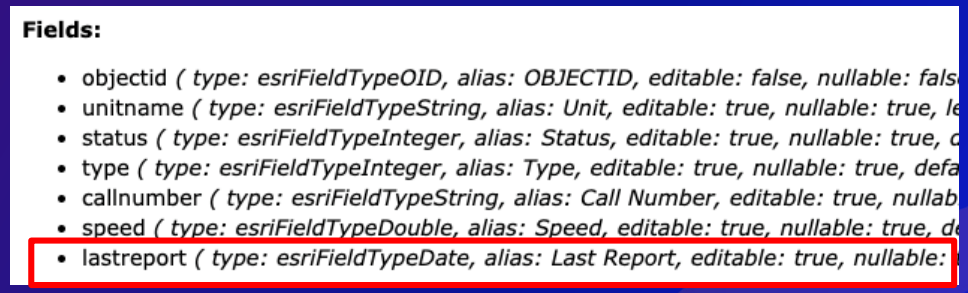


Animating temporal data with TimeSlider

- Makes temporal animation simple
- Animate time enabled services or layer
  - Service (map service, feature service, image service) must have timeInfo
  - Client-side layers with timeInfo defined
  - Set TimeSlider's view property



- Animate data using date field
  - Layer must have a date field
  - Watch TimeSlider.timeExtent property
    - Filter data by date using layer's definitionExpression, layerView.effect or layerView.filter



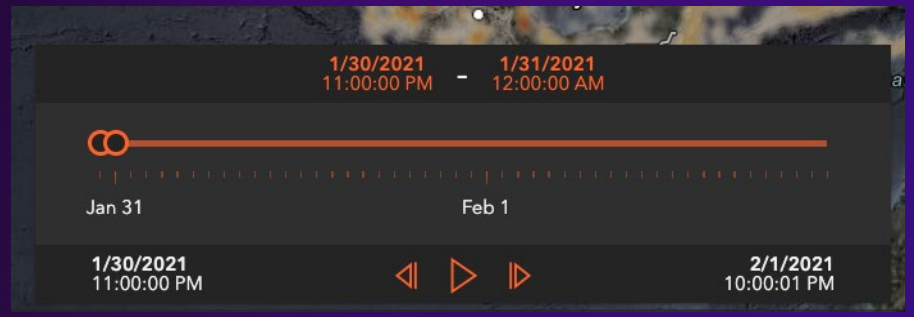
Animating time enabled services

- Map, feature and image services can be published with time info

Time Info:

Start Time Field: todate
End Time Field: null
Track ID Field: null
Time Extent
[1/31/2021 7:00:00 AM UTC, 2/2/2021 6:00:01 AM UTC]

Time Reference: UTC
Time Interval: 1
Time Interval Units: esriTimeUnitsHours
Has Live Data: false
Export Options:
Use Time: false
Time Data Cumulative: false
Time Offset: 0
Time Offset Units: esriTimeUnitsCenturies



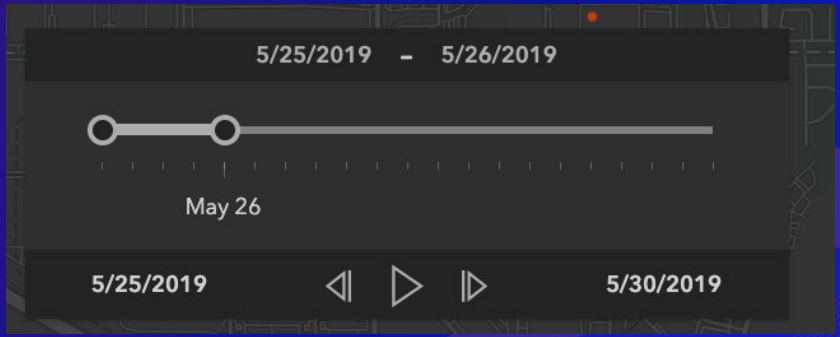

```
const smokeLayer = new FeatureLayer({
  url: "https://services9.arcgis.com/RHVPKKiFTONKtxq3/arcgis/rest/services
    /NDGD_SmokeForecast_v1/FeatureServer/0"
});

// set up time slider
view.whenLayerView(smokeLayer).then(function (lv) {
  const { fullTimeExtent, interval } = smokeLayer.timeInfo;

  const timeSlider = new TimeSlider({
    container: "timeSlider",
    mode: "time-window",
    view: view,
    timeVisible: true,
    fullTimeExtent,
    playRate: 100,
    loop: true,
    stops: {
      interval
    }
  });

  view.ui.add(timeSlider, "manual");
});
```

Animating time enabled client-side layers

- Set timeInfo property at the time of initialization for CSVLayer, GeoJSONLayer and client side FeatureLayer
- Layer must have a date field

```
const layer = new GeoJSONLayer({
  url:
    "https://bsvensson.github.io/various-tests/geojson
     /usgs-earthquakes-06182019.geojson",
  copyright: "USGS Earthquakes",
  title: "USGS Earthquakes",
  // set the CSVLayer's timeInfo based on the date field
  timeInfo: {
    startField: "time", // name of the date field
    interval: {
      // set time interval to one day
      unit: "days",
      value: 1
    }
  },
  renderer: {...},
  popupTemplate: {...}
});
```

```
const timeSlider = new TimeSlider({
  container: "timeSlider",
  playRate: 50,
  fullTimeExtent:{
    start: new Date(2019, 4, 25),
    end: new Date(2019, 4, 30)
  },
  stops: {
    interval: layer.timeInfo.interval
  },
  view: view
});
view.ui.add(timeSlider, "bottom-left");
```



Animating data using date field

- Initialize TimeSlider without view
  - Set TimeSlider properties as needed
  - Watch TimeSlider.timeExtent property and filter data based on the timeExtent

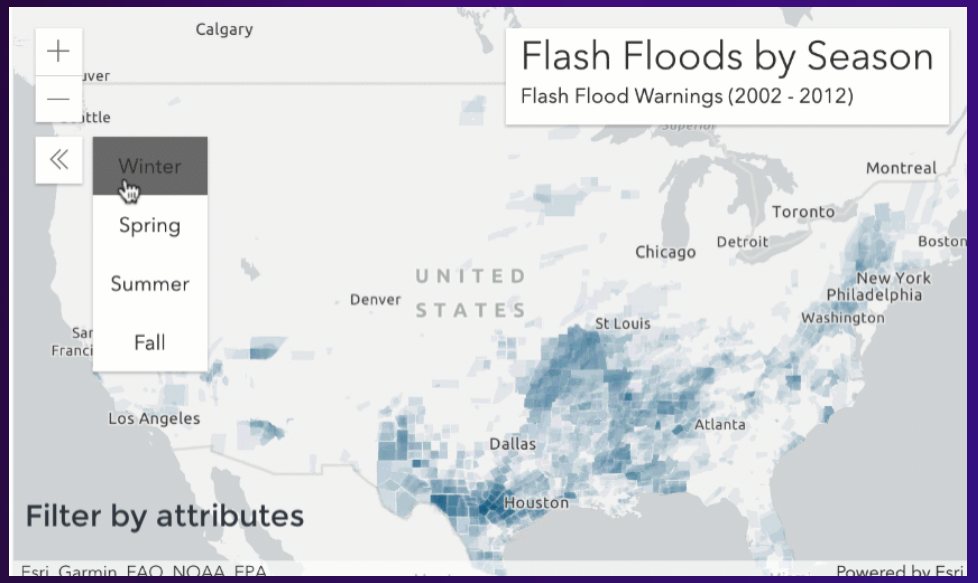
```
const featureLayer = new FeatureLayer({
  url: "https://services3.arcgis.com/T4QMspbfLg3qTGWY
    /arcgis/rest/services/Historic_GeoMAC_Perimeters_2019
    /FeatureServer/0",
  definitionExpression: "incidentname = 'KINCADE'",
  outFields: ["*"],
  renderer: {...},
  effect: "brightness(600%)",
  blendMode: "overlay",
});
```

```
var timeSlider = new TimeSlider({
  container: "timeSlider",
  mode: "instant",
  timeVisible: true,
  playRate: 200
});
view.ui.add(timeSlider, "bottom-left");
```

```
view.whenLayerView(featureLayer).then(function(lv){
  var layerView = lv;

  var query = featureLayer.createQuery();
  query.orderByFields = ["perimeterdatetime"];
  featureLayer.queryFeatures(query).then(function(results){
    results.features.forEach(function(feature){
      var dt = new Date(feature.attributes.perimeterdatetime);
      dates.push(dt);
    });
    timeSlider.stops.dates = dates;
    timeSlider.fullTimeExtent = {
      start: dates[0],
      end: dates[dates.length-1]
    };

    var start = dates[0].getTime();
    var where = `perimeterdatetime = ${start}`;
    layerView.filter = {
      where: where
    };
  });

  timeSlider.watch("timeExtent", function(){
    var start = timeSlider.timeExtent.start.getTime();
    var where = `perimeterdatetime = ${start}`
    layerView.filter = {
      where: where
    };
  });
});
```

Filter data by attributes

- Set layer's definitionExpression
  - (Feature|Imagery|Stream|MapImageLayer's Sublayer) - server-side operation
  - (CSV|GeoJSON) - client-side operation
- (Feature|OGCFeature|CSV|GeoJSON|Stream)LayerView
  - Set layerview's filter.where parameter
  - Set layerview's effect.filter.where parameter
  - Purely client-side operation



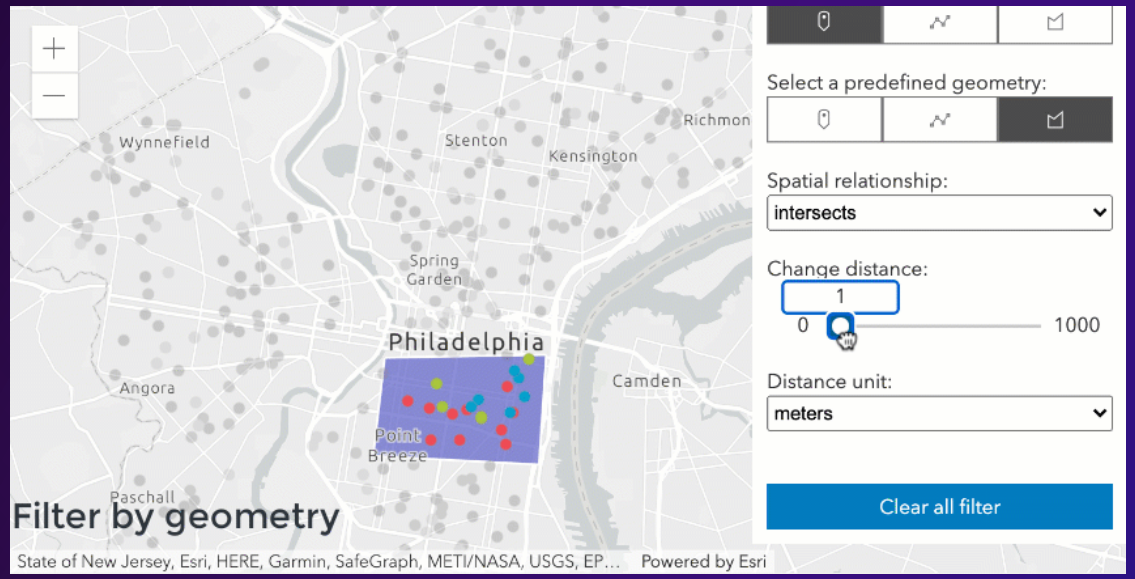

```
// flash flood warnings layer
const layer = new FeatureLayer({
  portalItem: {
    id: "f9e348953b3848ec8b69964d5bceae02"
  },
  outFields: ["SEASON"]
});
```

```
view.whenLayerView(layer).then(function (layerView) {
  floodLayerView = layerView;
});
```

```
function filterBySeason(event) {
  const selectedSeason = event.target.getAttribute("data-season");
  floodLayerView.filter = {
    where: "Season = '" + selectedSeason + "'"
  };
}
```

Filter data by geometry

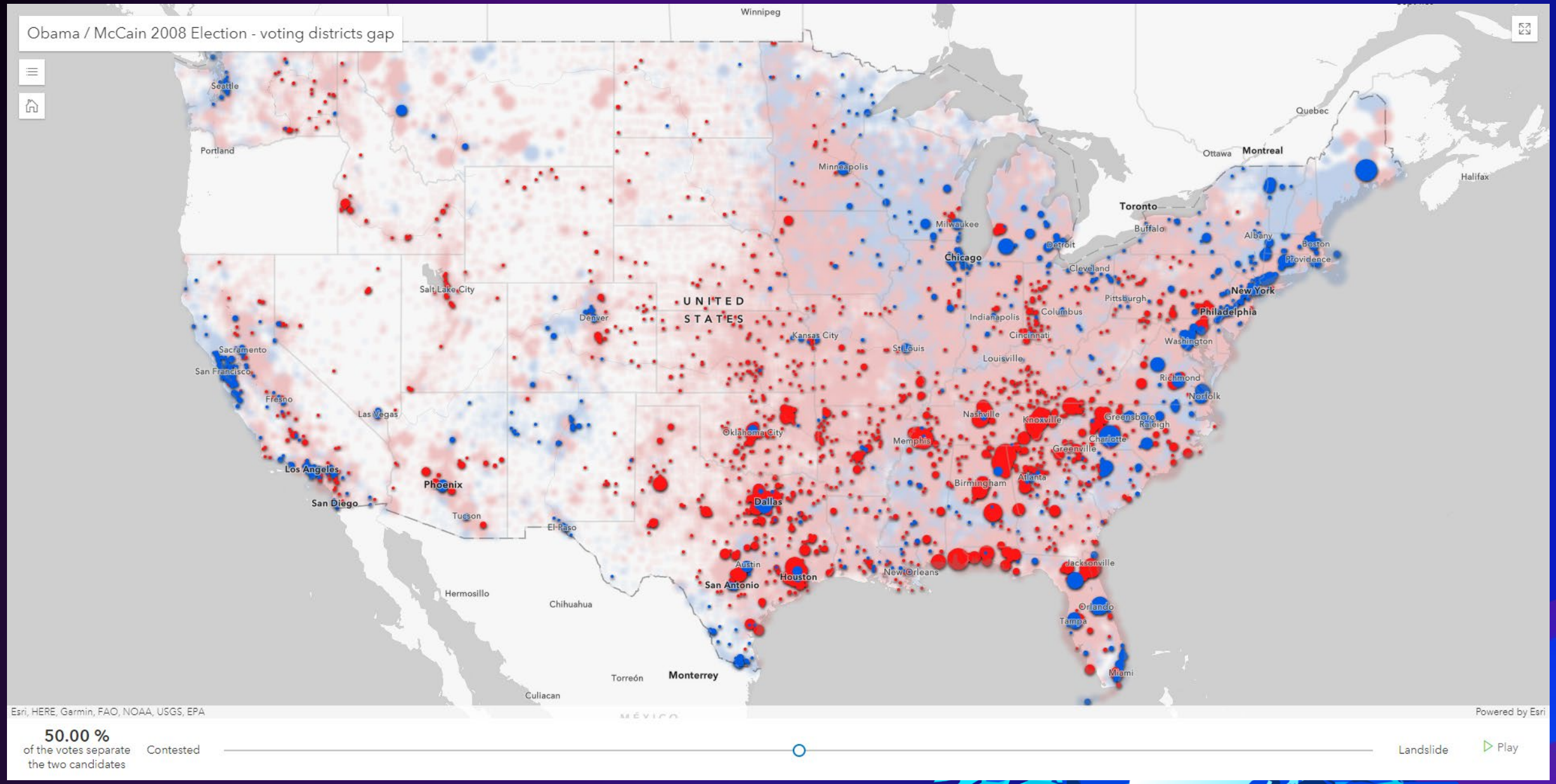
- (Feature|OGCFeature|CSV|GeoJSON|Stream)LayerView
  - Set LayerView's filter.geometry, filter.distance, filter.unit or
  - Set LayerView's effect.filter.geomery, effect.filter.distance, effect.filter.unit

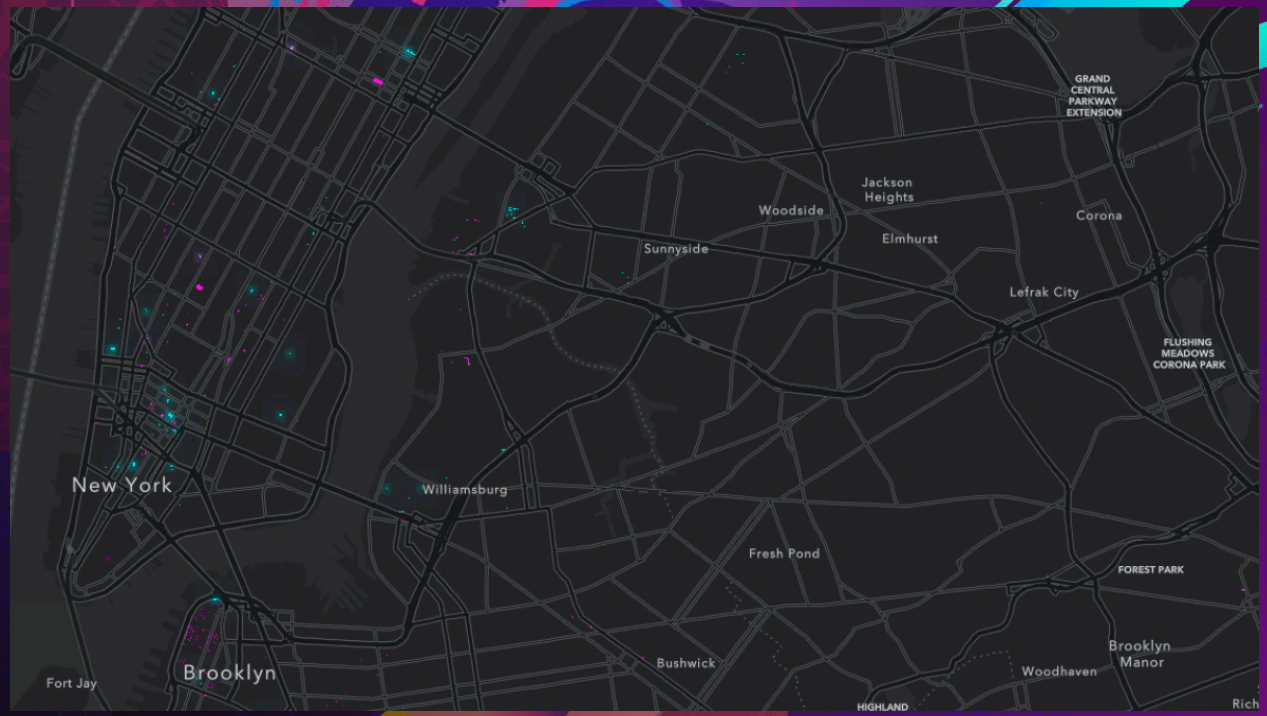



```
view.whenLayerView(layer).then(function (layerView) {
  featureLayerView = layerView;
});
```

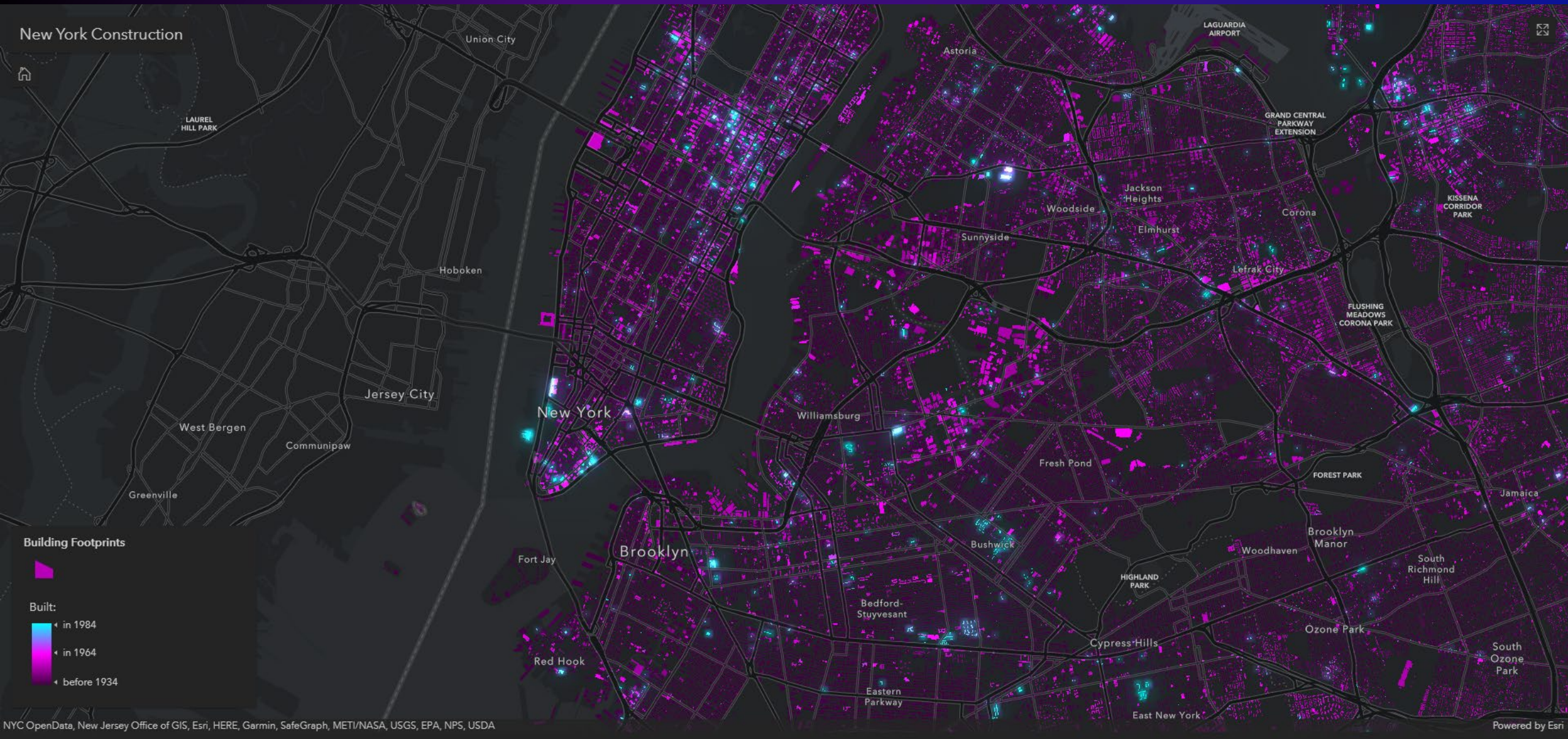
```
// set the geometry filter on the visible FeatureLayerView
function updateFilter() {
  featureFilter = {
    // autocasts to FeatureFilter
    geometry: filterGeometry,
    spatialRelationship: "intersects",
    distance: distance,
    units: unit
  };
  // set effect on excluded features
  // make them gray and transparent
  if (featureLayerView) {
    featureLayerView.effect = {
      filter: featureFilter,
      excludedEffect: "grayscale(100%) opacity(30%)"
    };
  }
}
```

Use LayerView Effects to hide features





Visual Variable Animation



1984 1880 2017 Play

Visual Variable Animation

- Fixed geometries/locations
  - Administrative boundaries (Countries/states/counties)
- **Update renderer** on each animation frame or slider change
  - Fixed data variable (usually a date, but could be another variable)
  - Change visual variable stops (color values, size, etc)
- Single field/column of interest in table

Data structure: Date field (or number representing year)

Footprints (Features: 1082433, Selected: 0)

NAME	BIN	BBL	CNSTRCT_YR	LSTMODDATE	LSTSTATYPE	DOITT_ID	HEIGHTROOF	FTRCODE
				PM				
	4,465,036	4163500300	1,938	2/13/2009, 4:00 PM	Constructed	539,338	18.82	2,100
	4,529,251	4159720050	1,920	2/13/2009, 4:00 PM	Constructed	980,512	11.86	5,110
	4,307,100	4163500002	1,929	9/18/2013, 5:00 PM	Constructed	137,895	22.31	2,100
	4,306,730	4163020016	1,920	2/13/2009, 4:00 PM	Constructed	730,180	35.62	2,100
	4,304,594	4162180003	1,930	2/13/2009, 4:00 PM	Constructed	408,322	37.11	2,100
	4,531,671	4156210015	2,006	6/8/2008, 5:00 PM	Constructed	998,124	34.62	2,100
	4,305,555	4162540064	1,935	2/13/2009, 4:00 PM	Constructed	599,241	22.40	2,100
	4,306,123	4162740064	1,930	2/13/2009, 4:00 PM	Constructed	546,445	30.61	2,100
	4,448,991	4161230094	1,930	2/13/2009, 4:00	Constructed	8,479	11.29	2,100

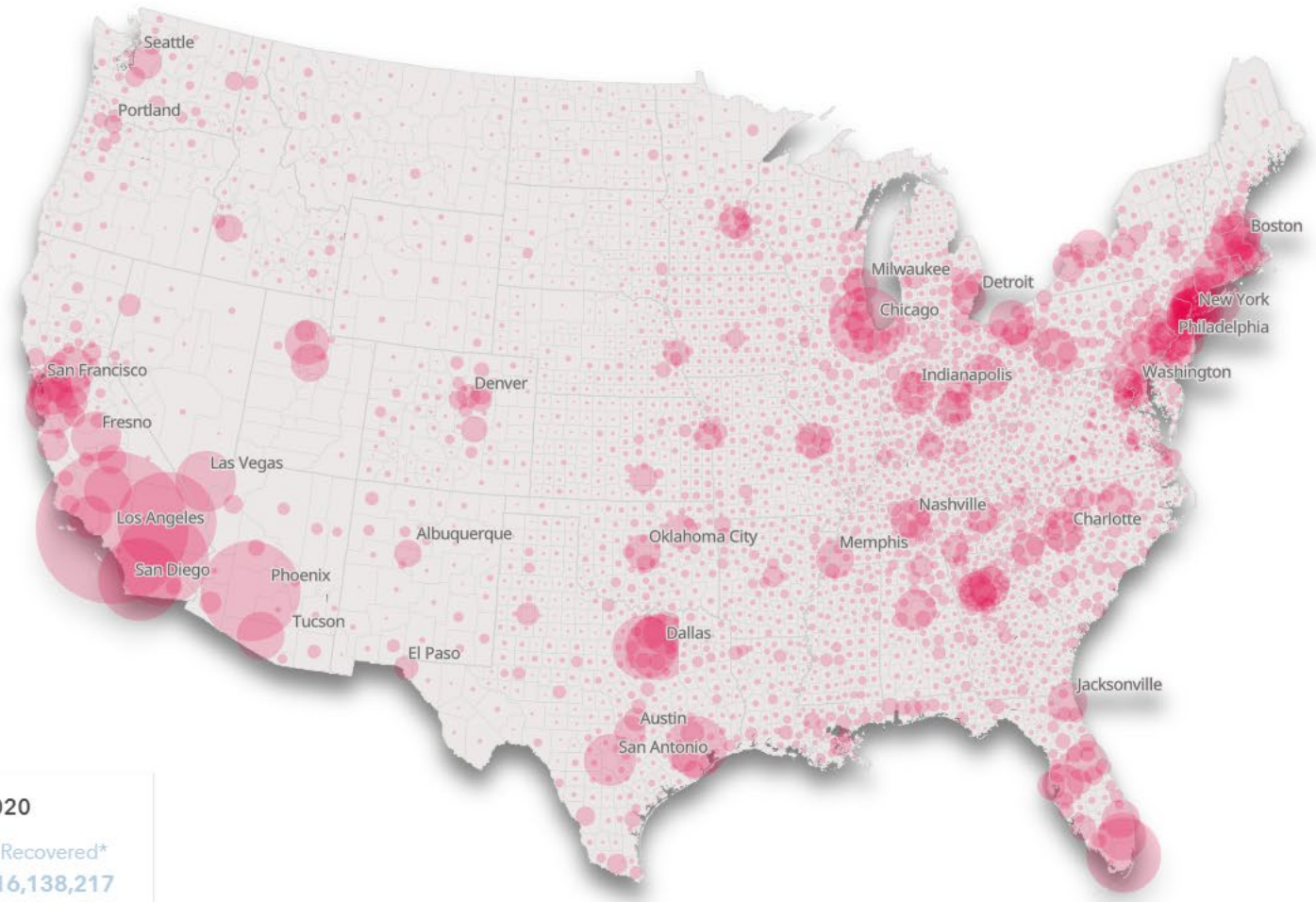
Same data attribute…new visual variable stops…

```javascript
function animate(startValue) {
  var animating = true;
  var value = startValue;

  var frame = function (timestamp) {
    if (!animating) {
      return;
    }

    value += 0.5;
    if (value > 2017) {
      value = 1880;
    }

    setYear(value);

    // Update at 30fps
    setTimeout(function () {
      requestAnimationFrame(frame);
    }, 1000 / 30);
  };

  frame();

  return {
    remove: function () {
      animating = false;
    }
  };
}
```

```javascript
function setYear(value) {
  sliderValue.innerHTML = Math.floor(value);
  slider.viewModel.setValue(0, value);
  layer.renderer = createRenderer(value);
}
```

```javascript
{
  type: "color",
  field: "CNSTRCT_YR",
  legendOptions: {
    title: "Built:"
  },
  stops: [
    {
      value: year,
      color: "#0ff",
      label: "in " + Math.floor(year)
    },
    {
      value: year - 10,
      color: "#f0f",
      label: "in " + (Math.floor(year) - 20)
    },
    {
      value: year - 50,
      color: "#404",
      label: "before " + (Math.floor(year) - 50)
    }
  ]
}
```

Data Variable Animation



Viral

Explore the spread of COVID-19 in the United States by county

Active cases

Explore date range

Estimated active* COVID-19 cases on 12/31/2020

- > 100,000
- 10,000
- 1,000
- 100
- 1
- < 0

*Active and recovered totals are estimates based on the CDC formula documented here.

December 31, 2020

Active*	Deaths	Recovered*
3,269,130	345,201	16,138,217
(996 per 100k)	(105 per 100k)	(4,917 per 100k)

*Active and recovered totals are estimates based on the CDC formula documented here. See source code for more information.

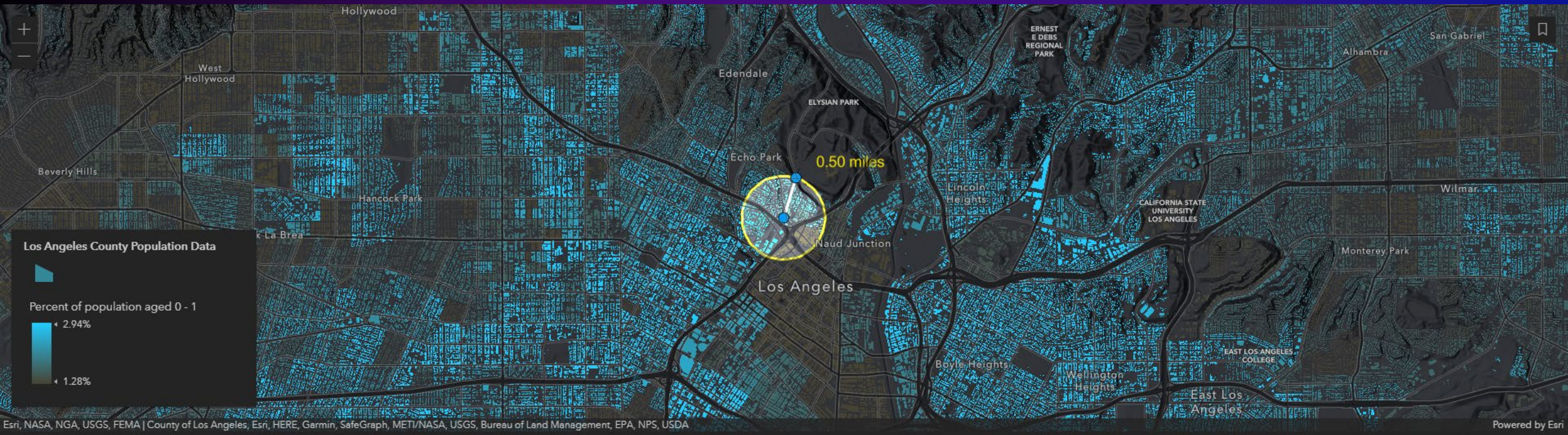
Labor Day | Thanksgiving | Christmas | Sturgis | One month ago | Two weeks ago

12/31/2020 | 1/22/2020

February March April May June July August September October November December

12/31/2020

Visualization: Centers for Civic Impact. Automation Support: Esri Living Atlas team, JHU APL, and JHU Sheridan Libraries. Data sources: Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University; the Red Cross; the Census American Community Survey; the Bureau of La... Powered by Esri



Age | Income

Render by age group

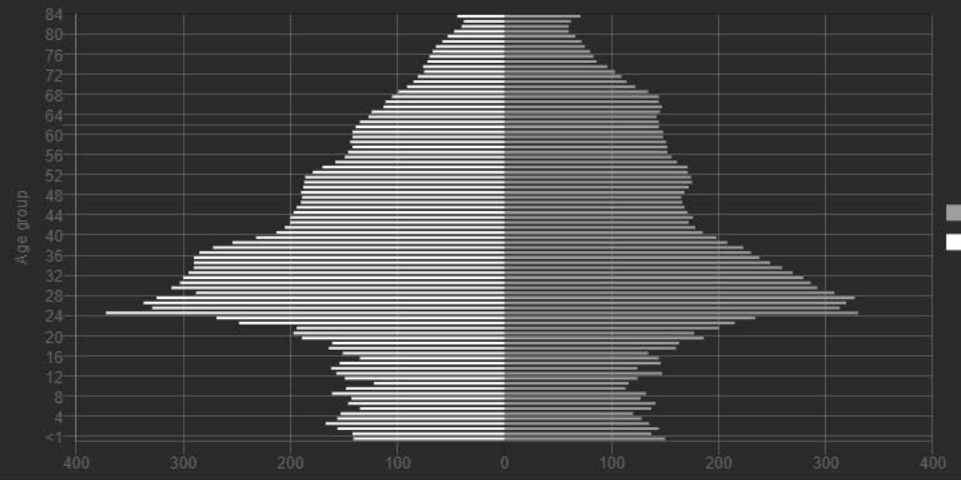
Choose from a predefined age range:

Custom	Infants (<1)	Children (1-11)	Teens (12-17)	College Age (18-22)	Millenials (24-39)	Generation X (40-55)	Baby Boomers (56-74)

Or use the slider to select a custom age range to be visualized:

0 01 84

Above and below color ramp



One Ocean

Explore temperature and salinity in the world's ocean

Salinity

Temperature

Salinity & temperature

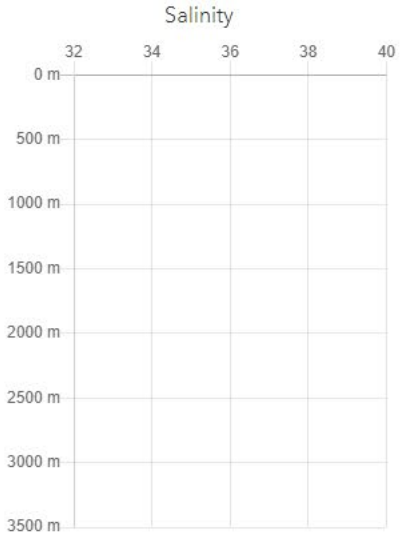
Show currents

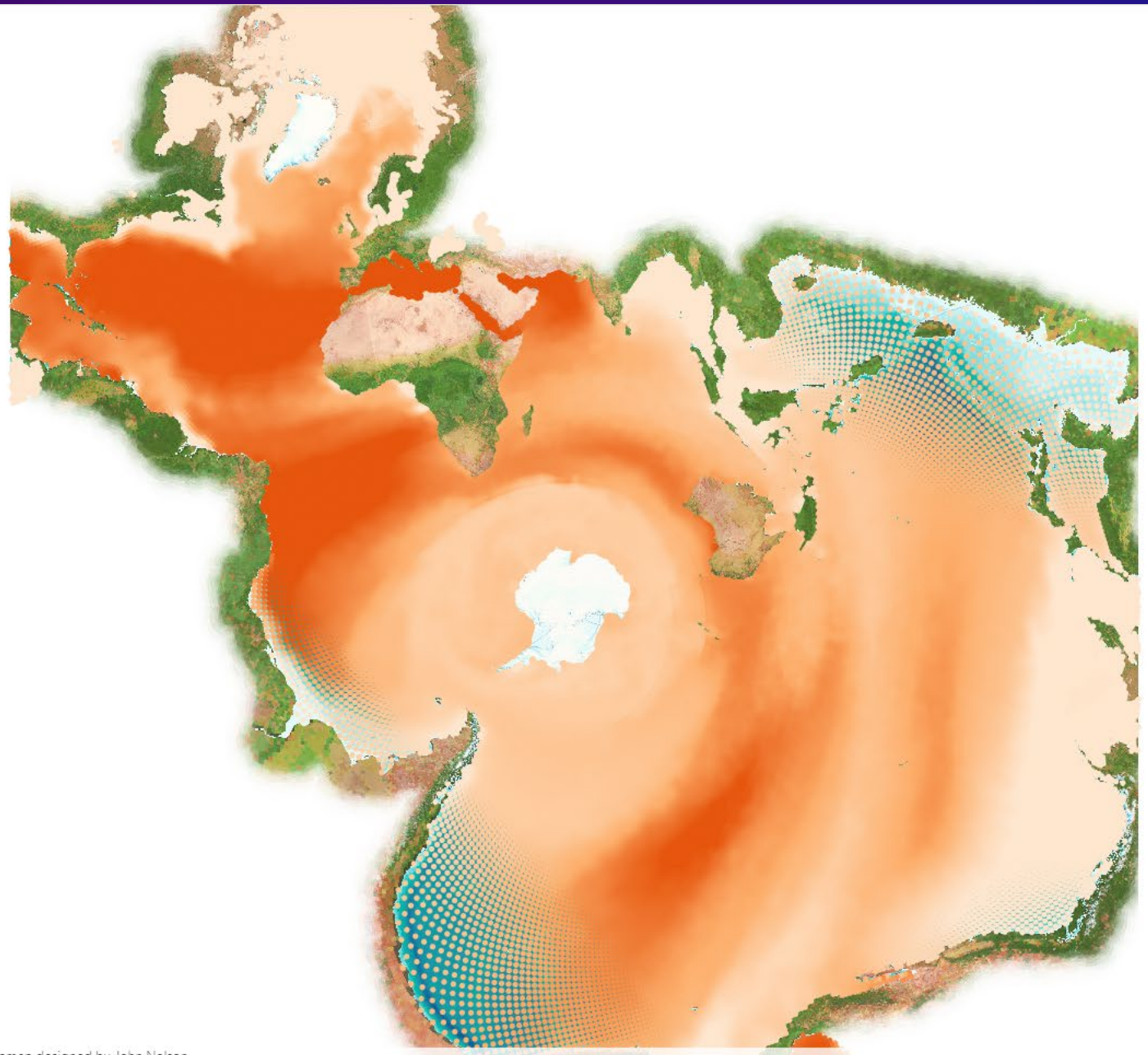
Depth

0

-600

Depth profile | Filter tools





App by Kristian Ekenes | Data provided by Esri | Basemap designed by John Nelson

Powered by Esri

Data variable animation

- Fixed geometries/locations
  - Administrative boundaries (Countries/states/counties)
- **Update renderer** on each animation frame or slider change
  - Changing data variable
    - Column name represents a date, but could be another dimension (like depth)
    - Data value could represent anything (population, votes, COVID positive tests, etc)
  - Fixed visual variable stops (color values, size, etc.) – IMPORTANT
- Requires one field per attribute per time interval (fat table)
  - Unless you have columns with separated values (Arcade workflow)

Data structure: one column per time interval

Yearly Temperature Anomaly by Time (Features: 2592, Selected: 1)

Year 2011	Year 2012	Year 2013	Year 2014	Year 2015	Year 2016	Year 2017	Year 2018	Year 2019	Year 2020
0.58	0.13	0.25	0.30	0.43	1.04	0.64	0.48	0.80	0.76
0.42	0.33	0.35	0.34	0.40	1.15	0.78	0.46	1.07	1.03
0.06	0.27	0.24	0.32	0.20	0.92	0.61	0.61	1.11	0.83
0.04	0.39						0.64	1.10	0.87
0.56	0.52						0.76	0.98	0.85
0.71	0.49						0.42	0.84	0.68
0.64	0.65						0.84	0.84	0.88
0.81	0.52						1.18	0.61	0.97
0.70	0.52						1.53	0.71	1.23
0.73	1.18						1.20	1.09	1.47
0.80	0.98						1.00	0.89	1.20
0.88	0.76						1.15	0.97	1.14
1.53	0.85	0.33	1.85	1.61	1.03	1.28	2.14	1.69	2.09
0.74	0.45	0.06	1.75	1.04	1.06	0.97	1.51	1.42	1.75
0.33	0.22	0.17	1.57	0.89	1.01	0.88	1.13	1.00	1.55

```
function updateRenderer(value) {
  renderer = layer.renderer.clone();
  const sizeVariable = renderer.visualVariables[0];
  const colorVariable = renderer.visualVariables[1];

  sizeVariable.valueExpression = getSizeValueExpression(value);
  colorVariable.field = `F${value}`;

  renderer.visualVariables = [sizeVariable, colorVariable];
  layer.renderer = renderer;
}
```

Data structure: one column per dimension per attribute

emu_thinned (Features: 84705, Selected: 0)

salinity_1	Direction _1	Velocity_1	temp _1	salinity_2	linity_3
31.6203994750976	42.5748100280761	0.02132532931864	-1.73431003093719	31.92269	.94371032
31.6254005432128	42.5748100280761	0.02132532931864	-1.73501002788543	31.92180	.94249916
31.6296100616455	43.6735916137695	0.02831844985485	-1.73491001129150	31.91990	.94038963
31.6310100555419	43.6735916137695	0.02831844985485	-1.73460996150970	31.91720	.93730926
31.6299896240234	41.8007202148437	0.02393255941569	-1.73390996456146	31.91501	.93429946
31.6333103179931	43.9740982055664	0.02814562991261	-1.73390996456146	31.91230	.93119049
31.6340904235839	41.9775314331054	0.02502400986850	-1.73351001739501	31.91039	.92880058
31.6324901580810	47.2189407348632	0.02865925058722	-1.73341000080108	31.90970	.92761039
31.6349086761474	46.6915512084960	0.02904842048883	-1.73470997810363	31.90888	.92639923
31.6364002227783	43.8481101989746	0.02668881975114	-1.73351001739501	31.90880	.92629051
31.6358089447021	45.4831199645996	0.02163220942020	-1.73250997066497	31.90711	.92459106
31.6357898712158	45.2141494750976	0.02295166067779	-1.73451006412506	31.90709	.92448997
31.6352100372314	47.0178489685058	0.02854073047637	-1.73240995407104	31.90820	.92568969
31.6340999603271	53.6866111755371	0.02942889928817	-1.73371005058288	31.90679	.92419052

```
emuLayer.renderer = {
  type: "simple",
  symbol: {…
  },
  visualVariables: [{
    type: "color",
    // temp_1, temp_2, temp_3, etc
    field: `temp_${getLevelFromDepth(depthSlider.values[0])}`,
    stops: […
    ]
  }, {
    type: "size",
    // Velocity_1, Velocity_2, Velocity_3, etc
    field: `Velocity_${getLevelFromDepth(depthSlider.values[0])}`,
    minDataValue: 0,
    maxDataValue: 0.5,
    // update size based on view scale
    minSize: {…
    },
    maxSize: {…
    }
  }, {
    type: "rotation",
    // Direction_1, Direction_2, Direction_3, etc
    field: `Direction_${getLevelFromDepth(depthSlider.values[0])}`,
    rotationType: "geographic"
  }]
};
```

Data preprocessing: table pivot

OBJECTID	latitude	longitude	temp	salinity	depth_level
1	89.75	-179.25	-1.54781	31.47959	1
2	89.75	-179.25	-1.65331	31.6204	2
3	89.75	-179.25	-1.73431	31.92269	3
4	89.75	-179.25	-1.73891	31.94371	4
5	70.75	-177.25	-1.53821	31.4886	1
6	70.75	-177.25	-1.65351	31.62961	2
7	70.75	-177.25	-1.73491	31.9199	3
8	70.75	-177.25	-1.7386	31.94039	4

- Reduces number of features (geometries)
- Increases number of attributes
- Reduces overall download size

OBJECTID	latitude	longitude	temp_1	salinity_1	temp_2	salinity_2	temp_3	salinity_3	temp_4	salinity_4
1	89.75	-179.25	-1.54781	31.47959	-1.65331	31.6204	-1.73431	31.92269	-1.73891	31.94371
2	70.75	-177.25	-1.53821	31.4886	-1.65351	31.62961	-1.73491	31.9199	-1.7386	31.94039

Data structure: multiple values per column

USdaily_count_poly (Features: 3142, Selected: 0)

DAYSTRING_08...	DAYSTRING_08...	DAYSTRING_08...	DAYSTRING_08...	DAYSTRING_08...	DAYSTRING_08/23/2020	DAYSTRING_08...	DAYSTRING_08...	DAYSTRING
1293|22	1304|22	1316|22	1318|22	1337|22	1343|22	1357|22	1365|22	1375|22
4029|30	4058|31	4100|32	4132|32	4148|32	4171|32	4247|32	4296|33	4330|34
692|7	694|7	710|7	712|7	715		28|7	736|7	741|7
507|5	512|6	515|6	521|6	521		26|6	530|5	532|5
1113						1224|6	1239|7	1251|7
535						541|13	542|13	544|13
809						824|36	827|36	829|36
2181					23	2360|24	2380|25	2409|26
949						982|38	1012|38	1015|38

4132|32

```
export function createTotalDeathsExpression (currentDateFieldName: string){
  return `
    var currentDayFieldName = "${currentDateFieldName}";
    var currentDayValue = $feature[currentDayFieldName];

    var parts = Split(currentDayValue, "|");

    var cases = Number(parts[0]);
    var deaths = Number(parts[1]);

    return deaths;
  `;
}
```

New data attribute…same visual variable stops…

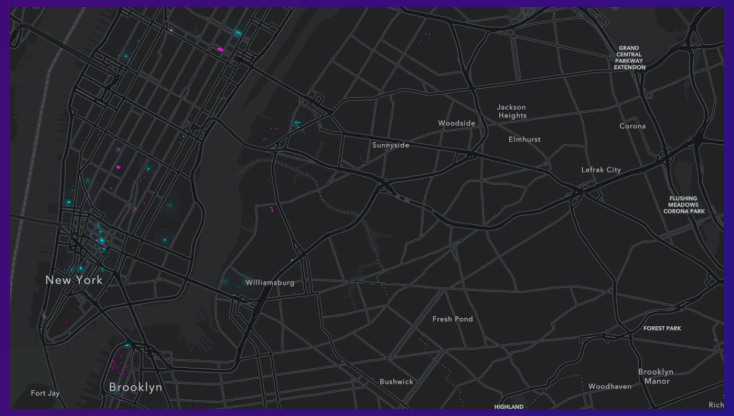
```
slider.on(["thumb-change", "thumb-drag"], function (event) {
  updateRenderer(event.value);
  updateHistogram(event.value);
  updateYearDisplay(event.value);
});
```

```
function updateRenderer(value) {
  renderer = layer.renderer.clone();
  const sizeVariable = renderer.visualVariables[0];
  const colorVariable = renderer.visualVariables[1];

  sizeVariable.valueExpression = getSizeValueExpression(value);
  colorVariable.field = `F${value}`;

  renderer.visualVariables = [sizeVariable, colorVariable];
  layer.renderer = renderer;
}
```

- Update renderer on each animation frame or slider change
- **Fixed visual variable stops (color values, size, etc.) – IMPORTANT**
- Changing data variable
  - Column name represents a date, but could be another dimension (like depth)
  - Data value could represent anything (population, votes, COVID positive tests, etc)

Conclusion

How do you know which technique to use?

	Data filter animation	Visual variable animation	Data variable animation
			
Animate locations? Changing geometry?	✓	✗	✗
A single event in time and location?	✓	✗	✗
One data value?	✓	✓	✗
Changing data values in same location?	✗	✗	✓

esri®

THE SCIENCE OF WHERE®

Links to demos

- https://ubatsukh.github.io/arcgis-js-api-demos/devsummit2021/effect-wildfires/
- https://ubatsukh.github.io/arcgis-js-api-demos/devsummit2021/fire-perimeter/
- Viral - https://ekenes.github.io/covid19viz/
- One Ocean - https://ekenes.github.io/one-ocean-performance-comparison/4.19/
- Dynamic age exploration - https://annelfitz.github.io/DevSummit-presentations/DS-2021/plenary/age-income-in-LA/
- Samples
  - Filtering data by attributes - https://developers.arcgis.com/javascript/latest/sample-code/featurefilter-attributes/
  - Filtering data by geometry - https://developers.arcgis.com/javascript/latest/sample-code/featureeffect-geometry/
  - Animate color visual variable - https://developers.arcgis.com/javascript/latest/sample-code/visualization-vv-color-animate/
  - Animate opacity visual variable - https://developers.arcgis.com/javascript/latest/sample-code/visualization-vv-opacity-animate/

Presentation Title

Presenter Names

esri THE SCIENCE OF WHERE®

2021 ESRI USER CONFERENCE

Section Header

Section Subhead

Demo Title

Presenter(s)

esri®

THE SCIENCE OF WHERE®