

Interactive Client-Side Mapping with the ArcGIS API for JavaScript

Kristian Ekenes [@kekenes] | Richie Carmichael [@kiwiRichie]

<https://git.io/fjr3N> (printer friendly, pdf)



Agenda

- Layers and Layer Views
- Client-side queries
- Time
- Filters & Effects
- Updating Renderers
- Widgets and charts

Layers and Layer Views



Layers and Layer Views

- Server-side
 - (*fetch or stream features on demand*)
 - FeatureLayer
 - SceneLayer
- Client-side
 - (*fetch all features at once*)
 - CSVLayer
 - GeoJSONLayer
 - FeatureLayer (from feature collection)

Client-side query

You can work with the data from all these layers client-side using the same query APIs!

- (`CSV|GeoJSON`)`Layer`
- (`CSV|GeoJSON|Feature|Scene`)`LayerView`
 - `queryFeatures()`
 - `queryFeatureCount()`
 - `queryObjectIds()`
 - `queryExtent()`

Age Pyramid, Homicides, 3D buildings

Client-side query

- Features aren't always as they seem!
 - Quantized/generalized geometries in FeatureLayer
 - Very small features may not be present even if technically within the view's extent

Query neighbors

Client-side layers

```
const url = "https://earthquake.usgs.gov/earthquakes/feed/v1.0/"
const geoJSONLayer = new GeoJSONLayer({
  url: url + "summary/all_month.geojson",
  copyright: "USGS Earthquakes"
});
const csvLayer = new CSVLayer({
  url: url + "summary/2.5_week.csv",
  copyright: "USGS Earthquakes"
});
```

```
const featureLayer = new FeatureLayer({
  source: [
    new Graphic({ attributes: { myOid: 1 }, geometry: { ... } })
    new Graphic({ attributes: { myOid: 2 }, geometry: { ... } })
    new Graphic({ attributes: { myOid: 3 }, geometry: { ... } })
  ],
  fields: [
    new Field({
      name: "myOid",
      type: "oid"
    })
  ]
});
```

Client-side vs. Server-side

```
// Query all data on the server.  
var query = new Query({  
  where: "magnitude >= 5"  
});
```

```
// Query all data on the server.  
featureLayer.queryFeatureCount(query).then(function(count) {  
  console.log(` ${count} quakes found.`); // 54820 quakes found.  
});
```

```
// Query only data downloaded to the browser.  
featureLayerView.queryFeatureCount(query).then(function(count) {  
  console.log(` ${count} quakes found.`); // 9235 quakes found.  
});
```

About Time!

- Time Support @ 4.12
 - `Query.timeExtent`
 - `FeatureLayer`, `ImageryLayer`, `MapImageLayer`, `CSVLayer` and `GeoJSONLayer`
 - `MapView.timeExtent` & `SceneView.timeExtent`
 - `TimeSlider` widget
- Time Support @ 3.29
 - Time offsets, temporal renderer

Filters and Effects

- Filter - *Client-side spatial/aspatial/temporal filtering.*

```
// Only show earthquakes that occurred between 2000 and 2006.  
featureLayerView.filter = new FeatureFilter({  
    timeExtent: new TimeExtent({  
        start: new Date(2000, 0, 1),  
        end: new Date(2007, 0, 1)  
    })  
});
```

- Effects - *Visual effects applied to included/excluded features.*

```
// Show quakes less than 7 magnitude as faint shadows.  
featureLayerView.effect = new FeatureEffect({  
    filter: {  
        where: "magnitude >= 7"  
    }  
    excludedEffect: "grayscale(100%) opacity(0.5)"  
});
```

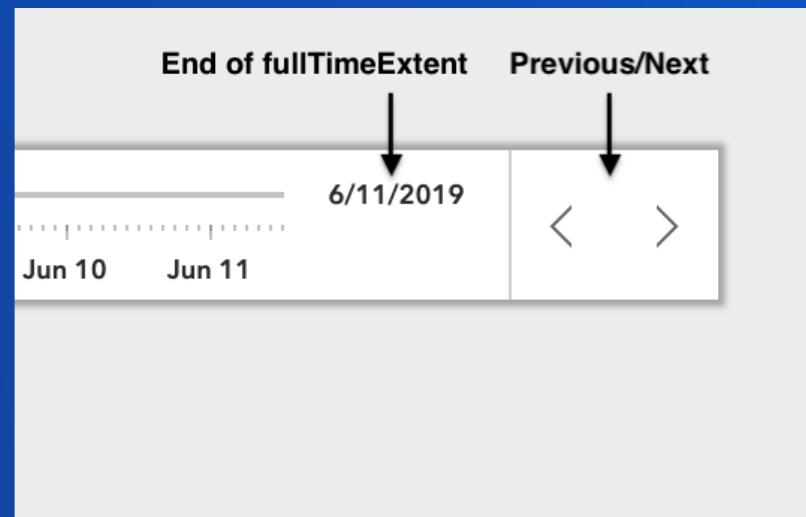
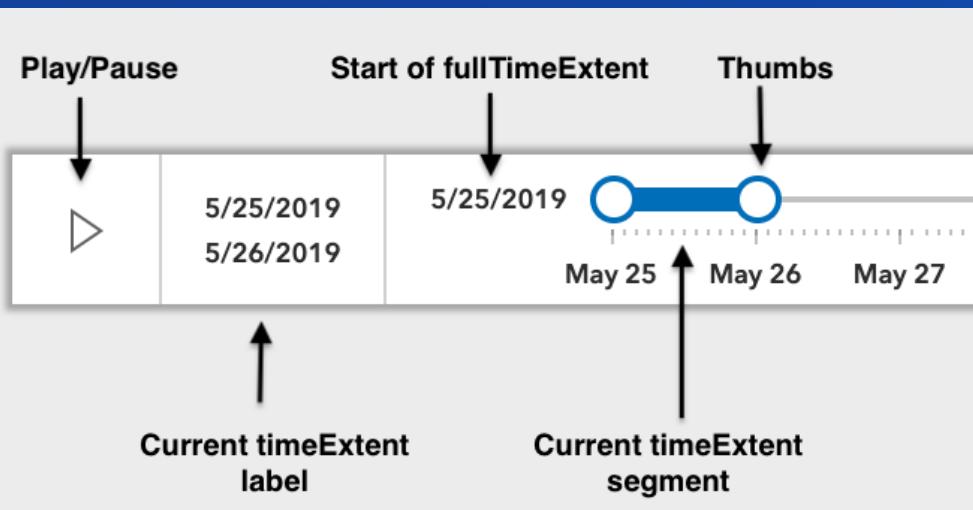


Demo

- Time, Filters and Effects - The Trifecta



TimeSlider Components

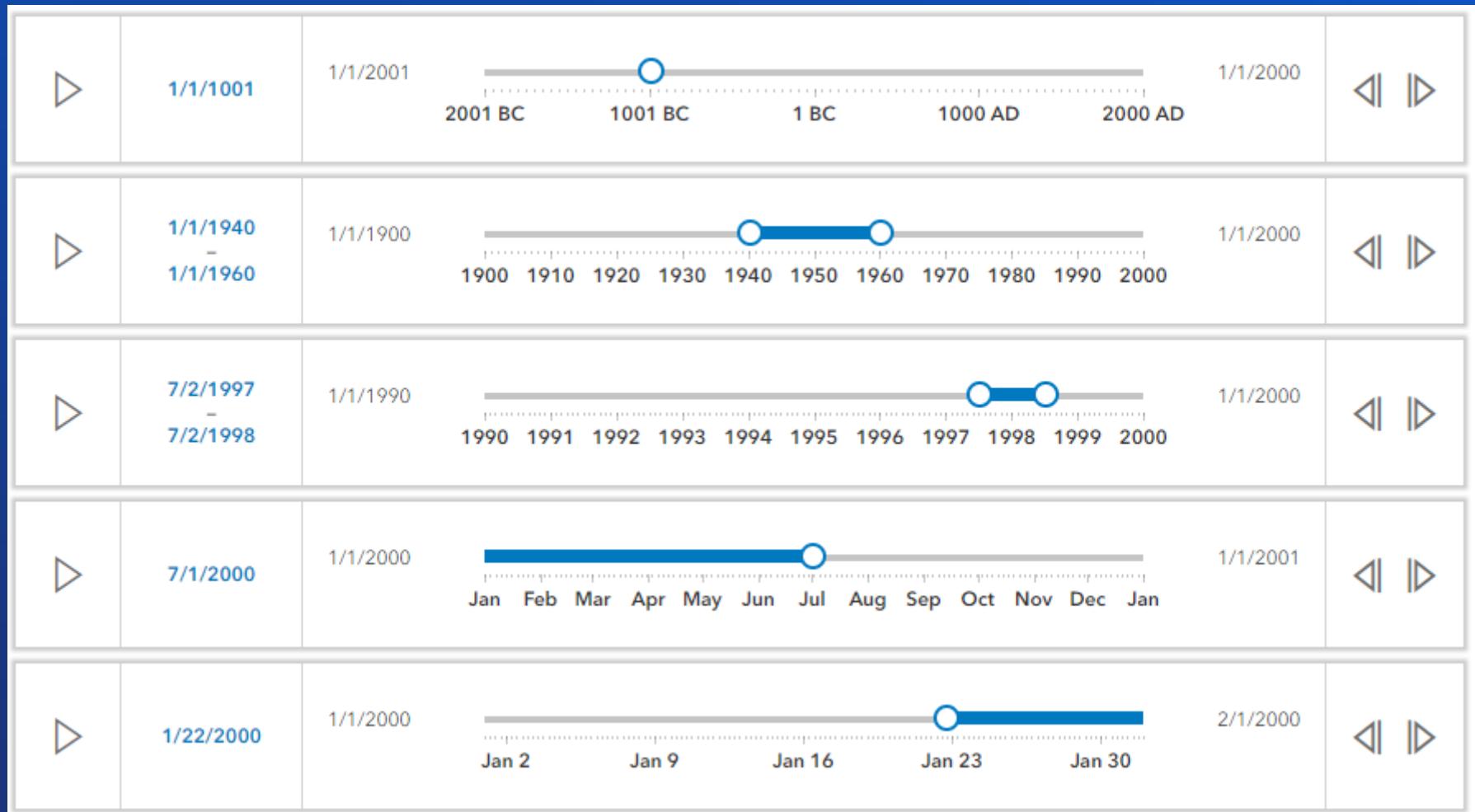


TimeSlider Properties

```
var timeSlider = new TimeSlider({
  container: "timeSliderDiv",           // HTML node
  fullTimeExtent: { start, end },      // Temporal width
  // optional
  mode: "time-window",                // Handle configuration
  playRate: 1000,                     // Play speed
  stops: {                           // Handle stops/steps
    interval: { value: 1, unit: "years" }
  },
  loop: true,                         // Allow looping
  timeVisible: false,                 // Show times
  values: [ first, second ],          // Handle positions
  view: view                          // The MapView/SceneView
});
```

TimeSlider Modes

instant | time-window | cumulative-from-start | cumulative-from-end



TimeSlider Stops

```
const timeSlider = new TimeSlider({  
  stops: {  
    count: 10 // Create ten evenly spaced stops.  
  }  
}) ;
```

```
const timeSlider = new TimeSlider({  
  stops: {  
    interval: { // Create stops spaced one year apart.  
      value: 1,  
      unit: "years"  
    }  
  }  
}) ;
```

```
const timeSlider = new TimeSlider({  
  stops: {  
    dates: [ // Explicitly create stops at these dates.  
      new Date(2000, 0, 1), new Date(2001, 3, 8),  
      new Date(2003, 12, 8), new Date(2004, 2, 19)  
    ]  
  }  
}) ;
```

Filter

```
// Only show earthquakes with a magnitude of 7 or greater.  
featureLayerView.filter = new FeatureFilter({  
  where: "magnitude >= 7"  
}) ;
```

```
// Only show buildings within 10 miles of the mouse cursor.  
mapView.on("pointer-move", function(e) {  
  buildingLayerView.filter = {  
    geometry: mapView.toMap({e.x, e.y}),  
    distance: 10,  
    units: "miles"  
  }  
}) ;
```

[API Reference](#) | [filter by attribute sample](#) | [filter by geometry sample](#)

Effects

```
// Show quakes less than 7 magnitude as faint shadows.  
featureLayerView.effect = new FeatureEffect({  
  filter: {  
    where: "magnitude >= 7"  
  }  
  excludedEffect: "grayscale(100%) opacity(0.5)"  
}) ;
```

```
// brightness(0.4);  
// contrast(200%);  
// grayscale(50%);  
// hue-rotate(90deg);  
// invert(75%);  
// opacity(25%);  
// saturate(30%);  
// sepia(60%);
```

Wait, there's more.

- A century of Earthquakes
- May/June Quakes
- Hurricanes and Storms
- Imagery with Time
- MapServer with Time
- Quakes in 3d

Updating renderers

- Update a visual variable
 - color, size, opacity, rotation
- Update a data value or expression
 - Field name
 - Arcade expression

Population Density by race/ethnicity | Housing density by decade

Median household income | Temperature anomaly | Population change

Histogram

```
// esri/renderers/smartMapping/statistics/histogram
const result = await histogram({
  layer: layer,
  field: "temp",
  numBins: 100
}) ;

// esri/widgets/Histogram
const histogramWidget = Histogram.fromHistogramResult(result);
histogramWidget.container = "histogram";
```

Basic Histogram

HistogramRangeSlider (sample)

```
// esri/renderers/smartMapping/statistics/histogram
const histogramResult = await histogram({
  layer: layer,
  field: "temp",
  numBins: 100
});
```

```
// esri/widgets/HistogramRangeSlider
const durationSlider = new HistogramRangeSlider({
  bins: histogramResult.bins,
  min: histogramResult.min,
  max: histogramResult.max,
  values: [histogramResult.min, histogramResult.max],
  rangeType: "between",
  container: document.getElementById("slider-container")
});
```

```
durationSlider.on(["value-change", "values-change"], function(eve
  layerView.filter = {
    where: durationSlider.generateWhereClause("temp")
  };
}) ;
```



esri

THE
SCIENCE
OF
WHERE