

[Back](#)

# nickcam/FlareClusterLayer

By  
github.com  
3 mins  
[View Original](#)

# nickcam/FlareClusterLayer

[github.com](#) | July 15, 2016

## FlareClusterLayer

A custom graphics layer that inherits from ArcGIS js graphics layer. Clustering is nothing special but I couldn't find an arcgis js api layer that clustered with flares which is why I created this.

The clustering is performed using a grid system based on the current extent, the pixel dimensions of the map and a configurable cluster ratio that can be set to suit a data set.

Support for both for 3.x and 4.x arcgis js apis.

Note: The latest version won't work with arcgis v4.x lower than v4.6. There are branches in this repo where you can get to the code that works for older versions.

## Features

- Flares will be created for individual points when the cluster contains  $\leq$  a configured amount of points. Selecting these clusters will open an info window for the object.
- Flares can also be created for counts of sub types. For example, if all of your data objects contain a property called 'Type', you can create flares for large clusters that contain the count of each unique value of the property 'Type'.
- The boundary of the points in a cluster can also be displayed as a polygon behind the flare with a separate renderer to style however you like. They can be displayed on hover, tap, all the time or not at all.
- Summary flares with the text '...' and a tooltip containing all data will be created if there's too many to fit in the configured total amount of flares to display.
- Configure the symbology to be whatever you want by using a renderer for the clusters and single symbols and an optional second renderer for the cluster boundaries.
- Supports MapView (2d) and SceneView (3d) for api v4.x.

All of the options are explained in the constructor of the layer/s so just check out the code for a full explanation. Demos are here:

v3.x - [http://flareclusterlayer.azurewebsites.net/index\\_v3.html](http://flareclusterlayer.azurewebsites.net/index_v3.html)

v4.x - [http://flareclusterlayer.azurewebsites.net/index\\_v4.html](http://flareclusterlayer.azurewebsites.net/index_v4.html)

## api v4.x notes

I used typescript to build the v4 version (because typescript rocks), so you could either use the typescript version and compile it to js in your own project or just use the compiled version in the fcl folder, same way the demo does.

Moved all of the animations out of the code and added css classes to elements instead. Animations can be performed using css instead of in code. This allows for much more flexibility. Example css that replicates/extends the v3.x animations are in the index\_v4.html example.

If you want to run the repo locally, just npm install and run npm start

There's no @types package for dojo v11.x yet, but there is an npm package 'dojo-typings'. Even the dojo-typings package doesn't go as high as 1.12.x which arcgis now uses. Had to just include a reference to the dojo types in a custom index.d.ts file, that also includes some custom typings as well.

If you plan to include the .ts file in your project you may need to change the import statements at the top of the file depending on the module loader and typescript compilation options you're using. For example, to use it in an angular 2 project you would change the imports to be -

```
import GraphicsLayer from "esri/layers/GraphicsLayer";
```

That is assuming you're using [esri-system-js](#) to load the arcgis api using systemjs.

You would probably also need to remove the reference to the index.d.ts from typings at the top of the file, and just include the required typings file however your project already manages this.

Cross browser notes on the example CSS animations:

- IE/Edge: These POS's don't support transforms on svg elements using css, so the css transform animations won't work.