Mapping API’s: Leaflet - Dash, Line Cap and Line Shape Path Options

Bailey A. Hanson
Iowa State University, bahanson@iastate.edu

Christopher J. Seeger
Iowa State University, cjseeger@iastate.edu

Follow this and additional works at: http://lib.dr.iastate.edu/extension_pubs

Part of the Architectural Technology Commons, Landscape Architecture Commons, Other Architecture Commons, and the Urban, Community and Regional Planning Commons

Recommended Citation
Hanson, Bailey A. and Seeger, Christopher J., "Mapping API’s: Leaflet - Dash, Line Cap and Line Shape Path Options" (2016). Extension and Outreach Publications. 188.
http://lib.dr.iastate.edu/extension_pubs/188

Iowa State University Extension and Outreach publications in the Iowa State University Digital Repository are made available for historical purposes only. Users are hereby notified that the content may be inaccurate, out of date, incomplete and/or may not meet the needs and requirements of the user. Users should make their own assessment of the information and whether it is suitable for their intended purpose. For current publications and information from Iowa State University Extension and Outreach, please visit http://www.extension.iastate.edu.
Mapping API’s: Leaflet - Dash, Line Cap and Line Shape Path Options

Welcome to the Essential ArcGIS Task Sheet Series. This series supplements the Iowa State University Geospatial Technology Training Program short course series. The task sheets are designed to provide quick, easy instructions for performing mapping tasks.

This task sheet builds off of Mapping API’s: Getting Started PM2082-14r and Mapping API’s: Leaflet - Creating Polylines and Polygons PM2082-15j. The code for those task sheets and this task sheet (pathOptionsLeaflet.html) can be found on the ISU Geospatial Technology Program GitHub page at https://github.com/ISUEOGTP/GISTaskSheets. This task sheet demonstrates how to create polygons and polylines and utilize three path options: dash, line cap, and line shape to modify how the stroke line appears.

1. Introduction and Setup
   a. First, you will need to start with a basic Leaflet map setup. Reference the task sheet: Mapping API’s: Leaflet - Getting Started PM2082-14r to learn how to get this set up, or get the starter code from our GitHub page at https://github.com/ISUEOGTP/GISTaskSheets/tree/master/Leaflet-Tutorials/helloLeaflet.htm.

   ```javascript
   var map = L.map('map').setView([43, -93], 5); 
   ```

   b. Change the zoom level from 10 to 5. **Hint: found at the end of line 24 in the starter code.**

2. Add Polyline and Polygon
   a. Following the map constructor, create a polyline and add it to the map. Be sure to include the options for the color, weight and opacity.

   ```javascript
   var polyline = L.polyline([  
      [47.2, -91.5],  
      [44.4, -95.2],  
      [42.3, -90.5],  
      [40.6, -95.5],  
      [37.2, -93.7],  
      [40.6, -85],  
   ], {  
      color: 'red',  
      weight: 5,  
      opacity: 0.7,  
   }).addTo(map
   ```

   b. Change `L.polyline` to `L.polygon` from step 2a to create a filled polygon instead of a polyline.
3. Add Dash Line Pattern

a. To add a dash line pattern, add `dashArray: '10, 20'` after the opacity option. The dashArray parameter takes a set of parameters to define the line and space lengths. The measurements are in pixels and are repeated along the entire line path. Note: The weight of the line will impact the appearance of the dash space by covering the open space and making it appear shorter.

    ```javascript
    var polyline = L.polyline([47.2, -91.5],
        [37.2, -93.7],
        [40.6, -85],
        {color: 'red',
         weight: 5
         opacity: 0.7
         dashArray: '4,12,20,12'
         lineJoin: 'miter',
         lineCap: 'square'
      }).addTo(map);
    ```

b. If you wanted a dot space dash pattern you could input parameters of `dashArray: '1,4,10,4'`. Note: on Retina display devices the actual pixel length will be longer than what is designated. Dash patterns Do not work on canvas-powered layers (e.g. Android 2).

c. The lineJoin option is used to specify the shape to be used at the corners of paths. Four values can be used by this option: miter, round, bevel, or inherit. Add the `lineJoin` option below dashArray. Note: make sure and include a comma following the dashArray option.

    ```javascript
    lineJoin: 'miter',
    ```

d. The lineCap option specifies the shape to use at the end of a line or line segment. A lineCap can have one of four values: butt, round, square, or inherit. Add the `lineCap` option below lineJoin. Note: make sure and include a comma following the lineJoin option.

    ```javascript
    lineCap: 'square'
    ```

e. Find more dash line patterns in the image to the right. Plug in the numbers to the dashArray option in Leaflet to visualize the pattern on the map.

    ```javascript
    // More dash line patterns
    ```

Contact:

Bailey Hanson bahanson@iastate.edu, 515-520-1436 or Professor Christopher J. Seeger, ASLA, GISP cseeger@iastate.edu, 515-509-0651 for more information about the Geospatial Technology Program. This task sheet and more are available at [www.extension.iastate.edu/communities/gis](http://www.extension.iastate.edu/communities/gis)