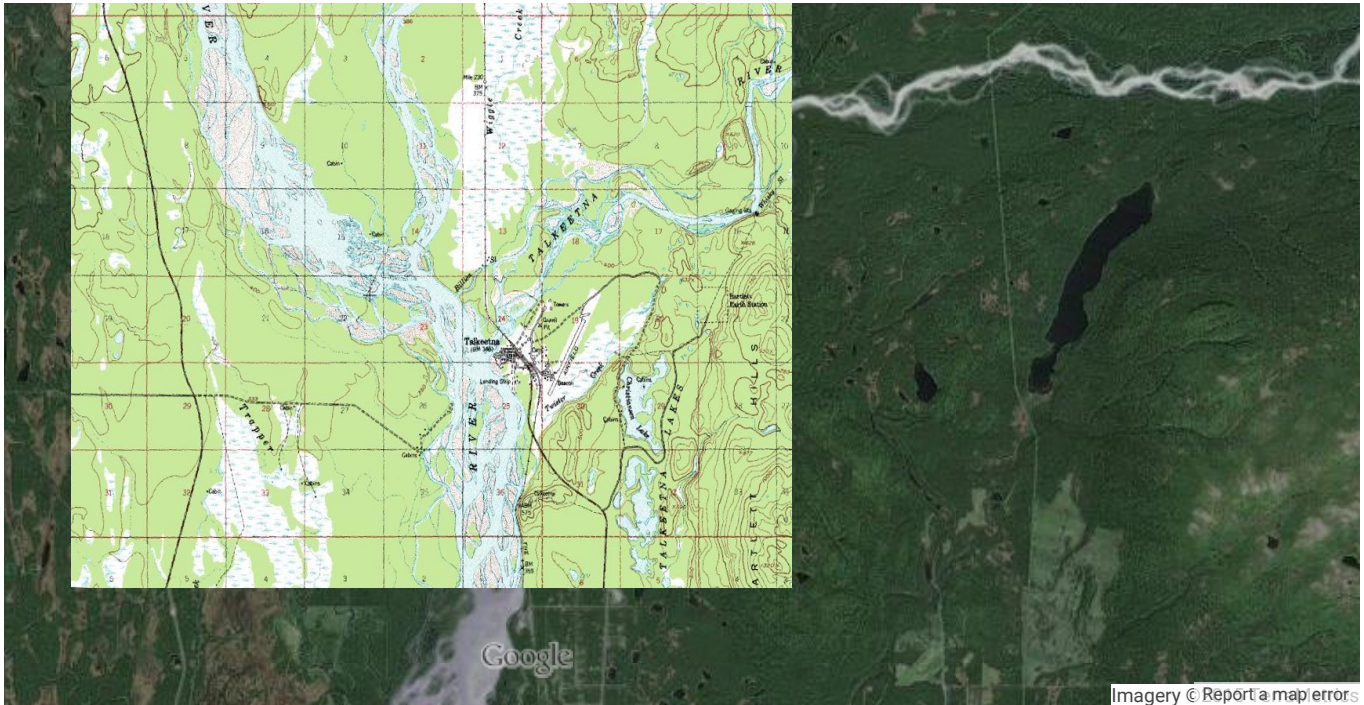


Adding a Custom Overlay



View this example [full screen](#).

JavaScript

JavaScript + HTML

```
// This example creates a custom overlay called USGSOverlay, containing
// a U.S. Geological Survey (USGS) image of the relevant area on the map.

// Set the custom overlay object's prototype to a new instance
// of OverlayView. In effect, this will subclass the overlay class.
// Note that we set the prototype to an instance, rather than the
// parent class itself, because we do not wish to modify the parent class.

var overlay;
USGSOverlay.prototype = new google.maps.OverlayView();

// Initialize the map and the custom overlay.

function initialize() {
  var mapOptions = {
    zoom: 11,
    center: new google.maps.LatLng(62.323907, -150.109291),
    mapTypeId: google.maps.MapTypeId.SATELLITE
  };
```

```

var map = new google.maps.Map(document.getElementById('map-canvas'), mapOptions);

var swBound = new google.maps.LatLng(62.281819, -150.287132);
var neBound = new google.maps.LatLng(62.400471, -150.005608);
var bounds = new google.maps.LatLngBounds(swBound, neBound);

// The photograph is courtesy of the U.S. Geological Survey.
var srcImage = 'https://developers.google.com/maps/documentation/javascript/';
srcImage += 'examples/full/images/talkeetna.png';

// The custom USGSOverlay object contains the USGS image,
// the bounds of the image, and a reference to the map.
overlay = new USGSOverlay(bounds, srcImage, map);
}

/** @constructor */
function USGSOverlay(bounds, image, map) {

    // Initialize all properties.
    this.bounds_ = bounds;
    this.image_ = image;
    this.map_ = map;

    // Define a property to hold the image's div. We'll
    // actually create this div upon receipt of the onAdd()
    // method so we'll leave it null for now.
    this.div_ = null;

    // Explicitly call setMap on this overlay.
    this.setMap(map);
}

/**
 * onAdd is called when the map's panes are ready and the overlay has been
 * added to the map.
 */
USGSOverlay.prototype.onAdd = function() {

    var div = document.createElement('div');
    div.style.borderStyle = 'none';
    div.style.borderWidth = '0px';
    div.style.position = 'absolute';

    // Create the img element and attach it to the div.
    var img = document.createElement('img');
    img.src = this.image_;
    img.style.width = '100%';
    img.style.height = '100%';
    img.style.position = 'absolute';
    div.appendChild(img);

    this.div_ = div;

    // Add the element to the "overlayLayer" pane.
    var panes = this.getPanes();
    panes.overlayLayer.appendChild(div);
};

```

```
USGSOverlay.prototype.draw = function() {  
  
    // We use the south-west and north-east  
    // coordinates of the overlay to peg it to the correct position and size.  
    // To do this, we need to retrieve the projection from the overlay.  
    var overlayProjection = this.getProjection();  
  
    // Retrieve the south-west and north-east coordinates of this overlay  
    // in LatLngs and convert them to pixel coordinates.  
    // We'll use these coordinates to resize the div.  
    var sw = overlayProjection.fromLatLngToDivPixel(this.bounds_.getSouthWest());  
    var ne = overlayProjection.fromLatLngToDivPixel(this.bounds_.getNorthEast());  
  
    // Resize the image's div to fit the indicated dimensions.  
    var div = this.div_;  
    div.style.left = sw.x + 'px';  
    div.style.top = ne.y + 'px';  
    div.style.width = (ne.x - sw.x) + 'px';  
    div.style.height = (sw.y - ne.y) + 'px';  
};  
  
// The onRemove() method will be called automatically from the API if  
// we ever set the overlay's map property to 'null'.  
USGSOverlay.prototype.onRemove = function() {  
    this.div_.parentNode.removeChild(this.div_);  
    this.div_ = null;  
};  
  
google.maps.event.addDomListener(window, 'load', initialize);
```

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#).

Last updated March 17, 2015.