



Esri International Developer Summit
Palm Springs, CA

Developing Custom JavaScript Applications with ArcGIS Online and Portal

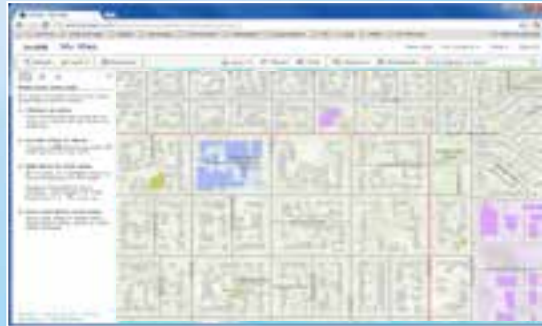
Kelly Hutchins Julie Powell Matt Driscoll
@kellyhutchins @JuliePowellGIS @driskull

ArcGIS Online Web maps

Can be used in many ways, in many applications, on many devices



ArcMap



Web map



Custom Applications



Mobile Apps

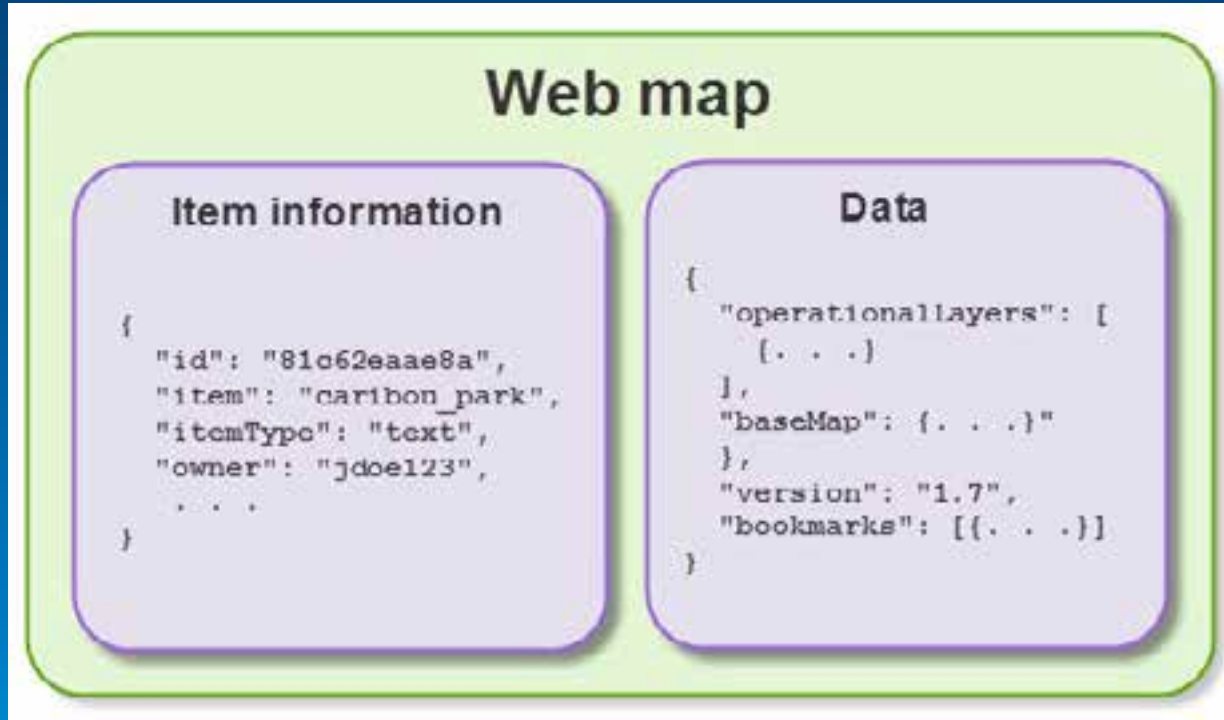


Configurable App Templates

Author a map

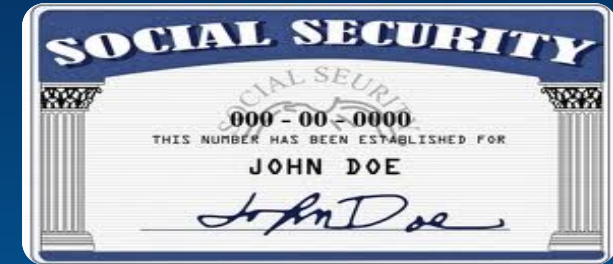


Web Map Specification



Map ID

A unique identifier, like a social security number



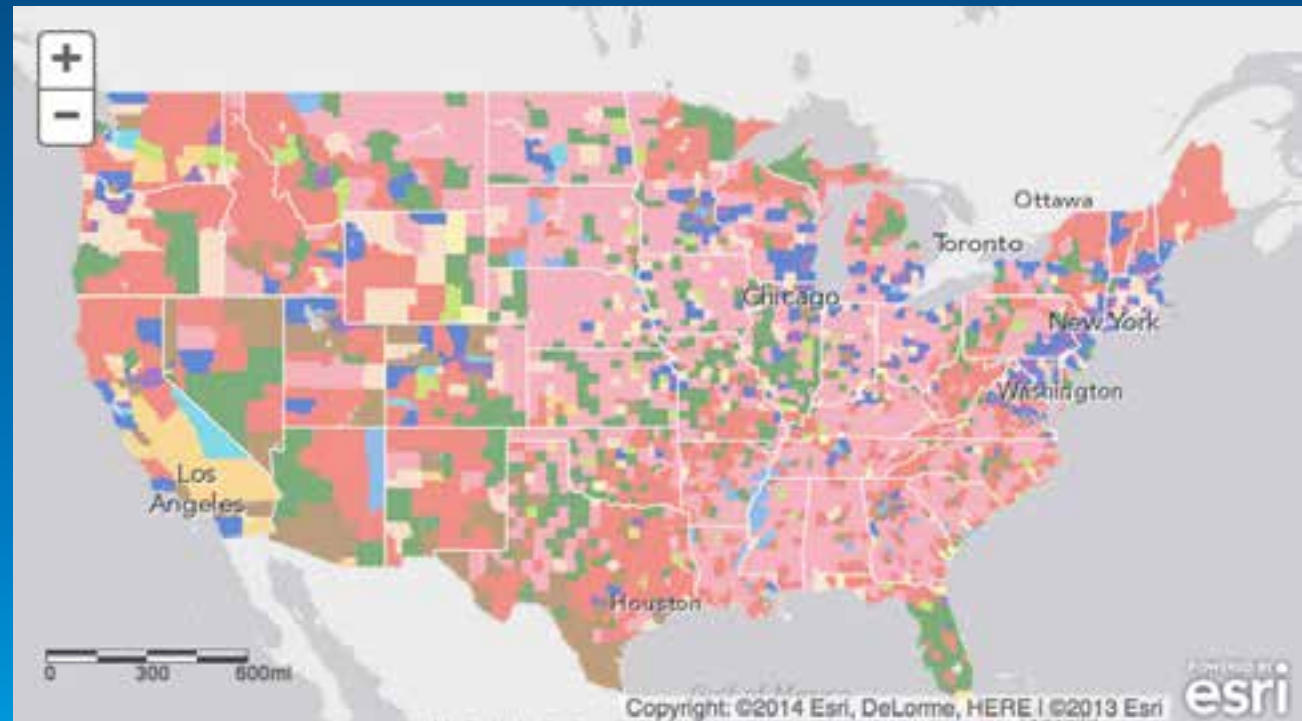
www.arcgis.com/home/webmap/viewer.html?webmap=ef5920f160bd4239bdeb1348de3a3156

Create Map



Web Maps

- `createMap`
- `getItem`
- `getLegendLayers`



ArcGIS Portal

- REST-based API to access and work with
 - Users
 - Groups
 - Items



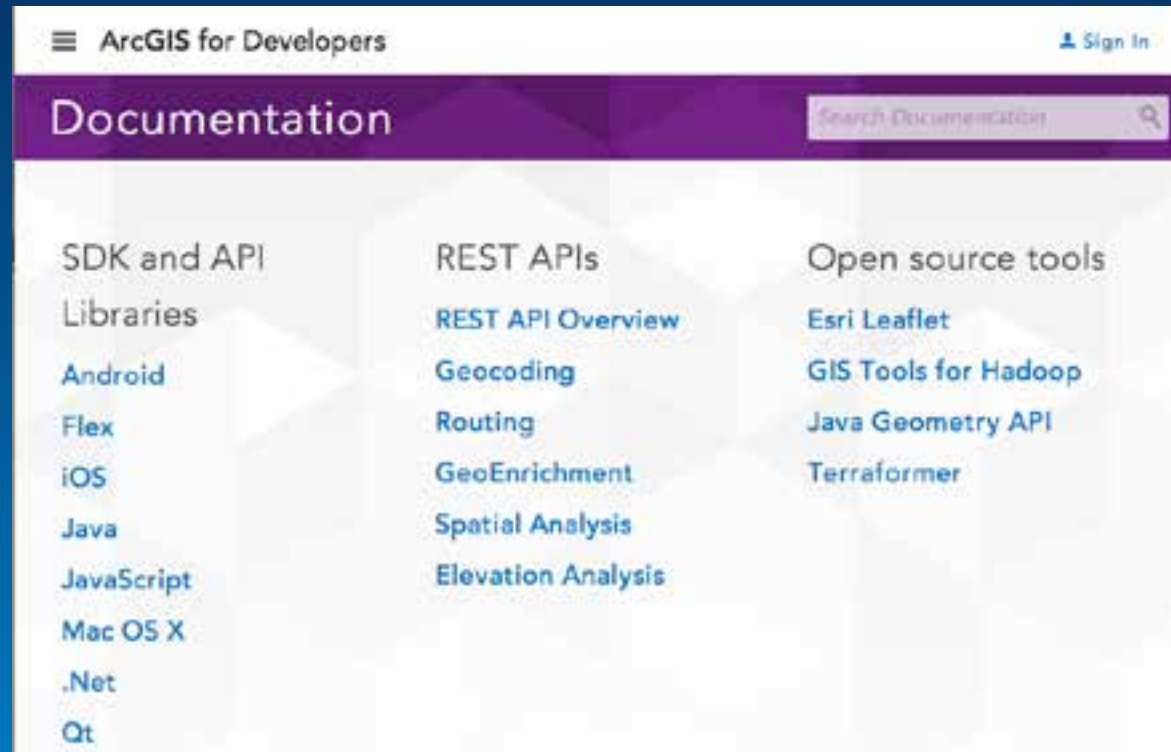
The screenshot displays the ArcGIS Resources website. At the top, the title "ArcGIS Resources" is centered, with navigation links for "Home", "Communities", "Help", and "Blog". Below this, the "ArcGIS REST API" section is highlighted in green. A sidebar on the left lists various resources, including "The ArcGIS REST API", "Services from Esri", "Your own services", and "Managing your organization". The main content area features a section titled "Working with users, groups, and items" under the heading "Managing your organization". This section includes a list of sub-topics: "Concepts", "Generate Token", "Root", "Search", "Community", "Content", "Portals", "Marketplace", and "OAuth 2.0". To the right of the text, there are four map thumbnails with captions: "USA Social Vulnerability Index", "USA Population Density", "USA Projected Population Change", and "Gulf Coast Oil Spill Plume". The "Gulf Coast Oil Spill Plume" map includes a legend with categories: "Oil Spill Plume Trajectory", "Uncertainty", "Light", "Medium", "Heavy", and "Beached Oil".

Using the REST API

```
request({
  url: portalUrl + '/sharing/rest/content/features/generate',
  content: myContent,
  form: dom.byId('uploadForm'),
  handleAs: 'json',
  load: lang.hitch(this, function (response) {
    if (response.error) {
      errorHandler(response.error);
      return;
    }
    var layerName = response.featureCollection.layers[0].layerDefinition.name;
    dom.byId('upload-status').innerHTML = '<b>Loaded: </b>' + layerName;
    addShapefileToMap(response.featureCollection);
  }),
  error: lang.hitch(this, errorHandler)
});
}
```

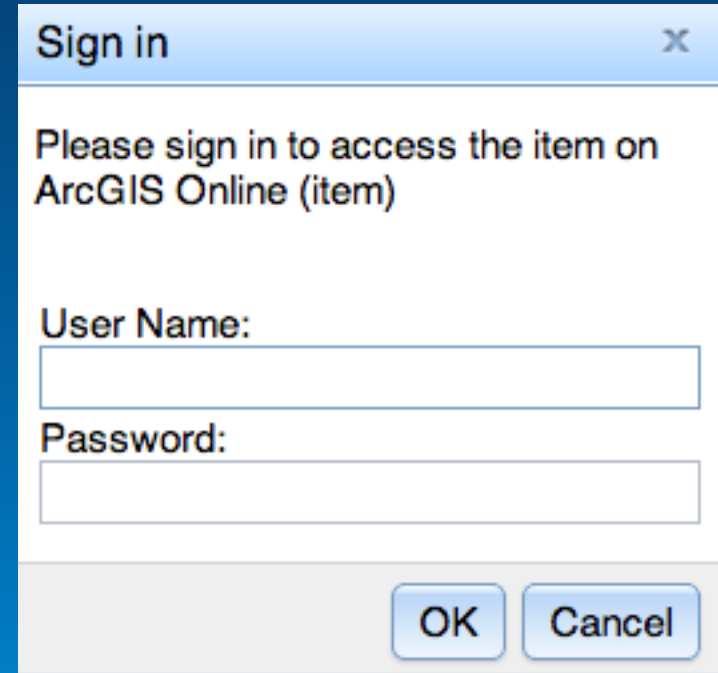
Client APIs

- Web APIs
- Runtime SDKs
 - iOS
 - Android



Authentication

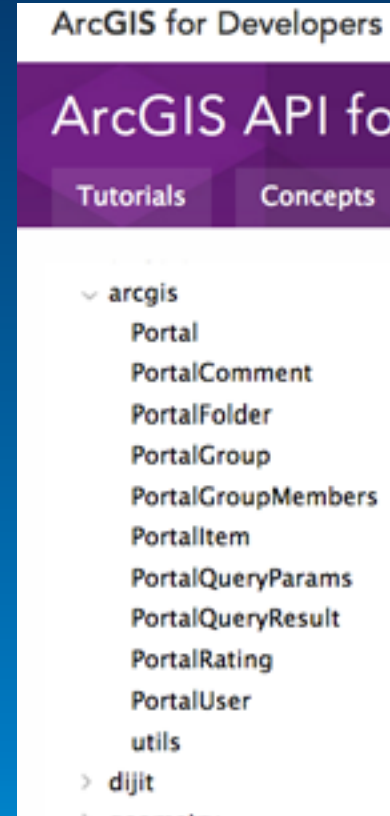
- Anonymous
- Authenticated



A screenshot of a 'Sign in' dialog box. The title bar contains the text 'Sign in' and a close button 'x'. The main text reads 'Please sign in to access the item on ArcGIS Online (item)'. Below this, there are two input fields: 'User Name:' followed by a text box, and 'Password:' followed by a text box. At the bottom right, there are two buttons: 'OK' and 'Cancel'.

ArcGIS API for JavaScript

- **Portal Class**
 - **Groups**
 - **Items**
 - **Users**



Templates: Download and Tweak

Julie Powell



Storytelling Text and
end



Storytelling Basic



Storytelling Swipe



Legend



r



Map Tour

Building Templates with ArcGIS WebApp Builder

Julie Powell



Make Ready to Use Apps –OR– App Templates

Builder



Ready to use app



App template



ArcGIS Portal/Online

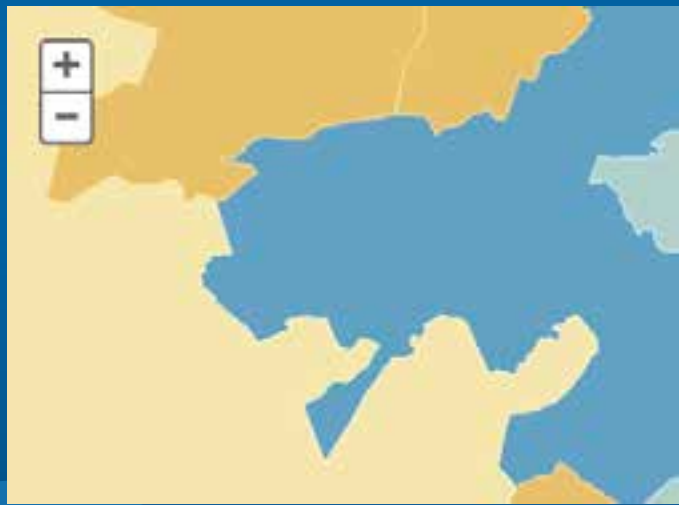


Templates
(Configurable Apps)



Applications

Application Boilerplate JS



```
<script type="text/javascript" src="//js.arcgis.com/3.11"></script>
<script type="text/javascript">
    require([
        "dojo/on",
        "application/main",
        "application/template",
        "dojo/domReady!"
    ], function(
        on,
        Main,
        Template
    ){
        //create the template. This will take care of the DOM
        //If you need localization set the supportsLocalization
        //from the javascript/nls/resource files
        //Note that we've included a placeholder
        //to show how to setup the strings file.

        var supportsLocalization = true; //default

        var myTemplate = new Template(supportsLocalization);

        on(myTemplate, "ready", function(config) {
            //The config object contains the following:
            //i18n, appid, webmap and any custom
            //In this example we have one called Main
            Main(config);
        });
    });
</script>
</body>
</html>
```

What is it?

- **Starter Application**
- **Template & app logic**
- **Working with ArcGIS Online**
 - **Webmaps**
 - **Services**
 - **i18n**
 - **Authentication**
 - **Configurable**

JS

HTML



CSS



Customizing the Boilerplate

Layers

Widgets

Panels

Legend

Anything!



What Do I Get?

Defaults

- Theme, proxy, custom settings for your app

Localization (Optional)

Web Map

Organization (Optional)

- Geocoders, print tasks, geometry services

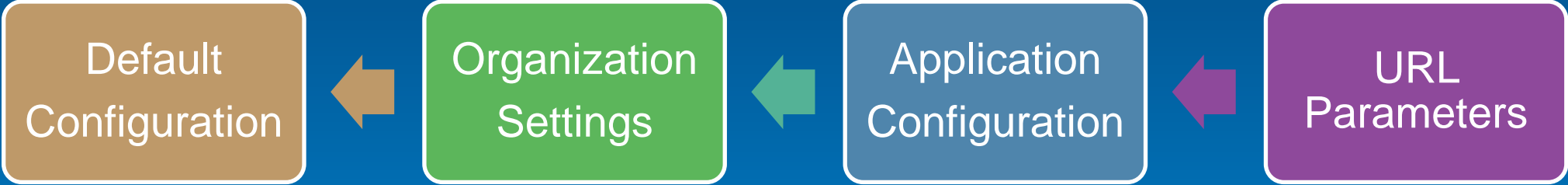
Configured Application (Optional)

- ?appid=<id>

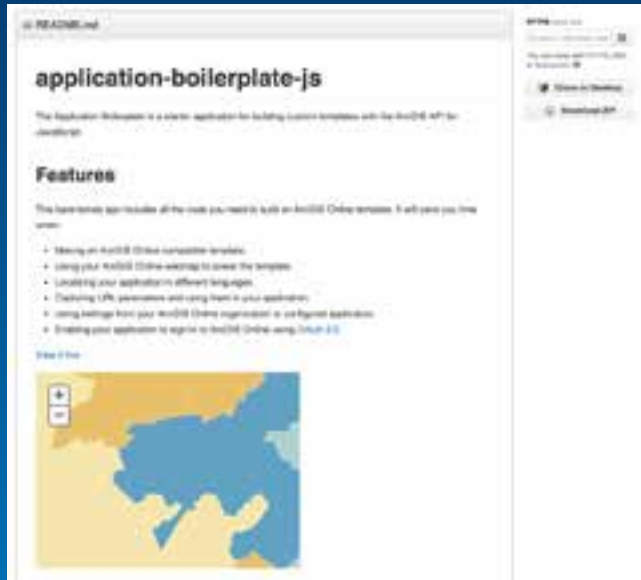
URL Parameters (Optional)

- ?webmap=<webmap>

Mixing-in Layers of Data



Show me the way



GitHub

esriurl.com/boilerplate

Boilerplate Structure

esriurl.com/boilerplate

Configurable Demo App!

Locate Button

Home Button

Customizable Title



Do You Live in Canine Country?

POWERED BY
esri

Prep Work

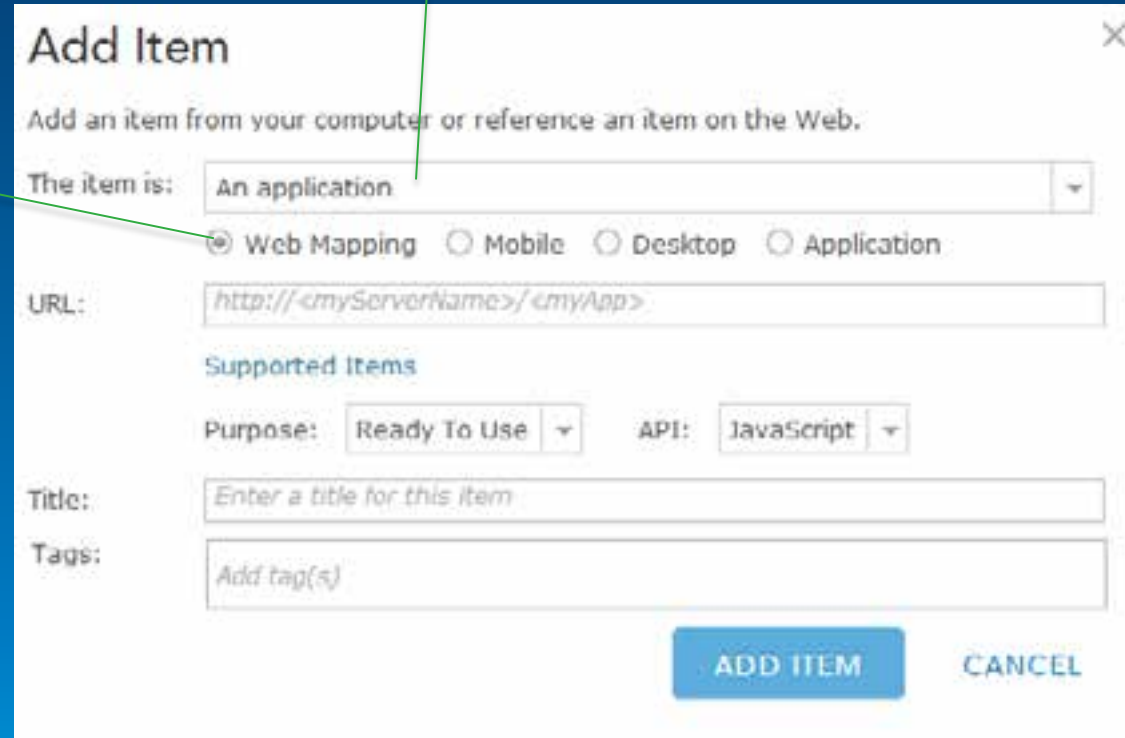
1. Create a configurable web application item
2. Create custom template group
3. Configure map viewer in org for custom templates
4. Share item in templates group



Adding the Application Item

Application

Web Mapping



The screenshot shows the 'Add Item' dialog box with the following fields and options:

- The item is:** A dropdown menu with 'An application' selected.
- Radio buttons:** 'Web Mapping' (selected), 'Mobile', 'Desktop', and 'Application'.
- URL:** A text input field containing the placeholder 'http://<myServerName>/<myApp>'.
- Supported Items:** A section with two dropdown menus: 'Purpose' set to 'Ready To Use' and 'API' set to 'JavaScript'.
- Title:** A text input field with the placeholder 'Enter a title for this item'.
- Tags:** A text input field with the placeholder 'Add tag(s)'.
- Buttons:** 'ADD ITEM' and 'CANCEL' at the bottom right.

Two green callout boxes are present: one labeled 'Application' pointing to the 'An application' dropdown, and another labeled 'Web Mapping' pointing to the selected 'Web Mapping' radio button.

Configurable Item

Application Link

Configurable

Add Item ✕

Add an item from your computer or reference an item on the Web.

The item is: ▼

Web Mapping Mobile Desktop Application

URL:

Supported Items

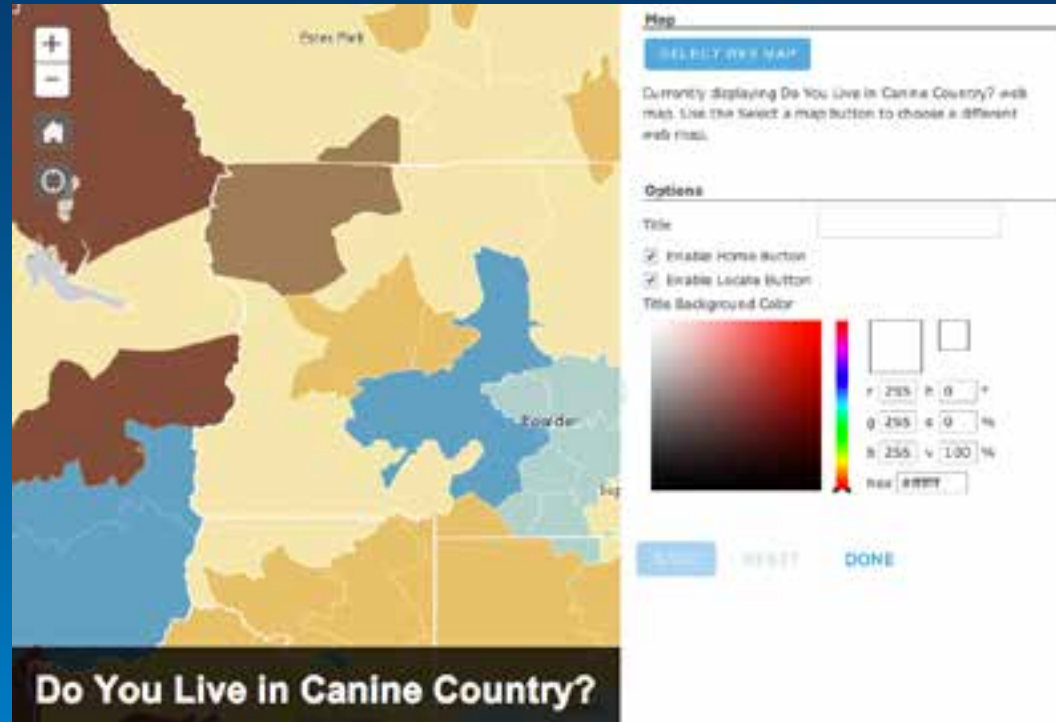
Purpose: ▼ API: ▼

Title:

Tags: ✕

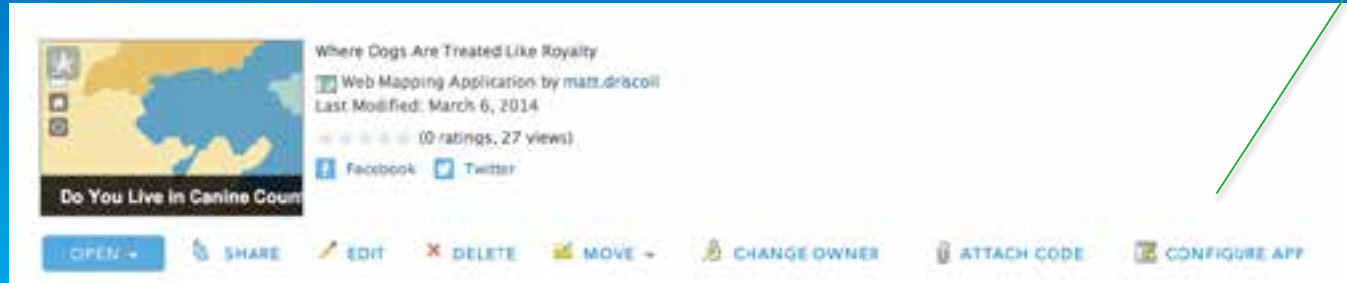
Add tag(s)

Lets make it configurable!



The screenshot shows a web mapping application interface. On the left is a map of the United Kingdom with a dark banner at the bottom that reads "Do You Live In Canine Country?". On the right is a configuration panel titled "Map" with a "SELECT YOUR MAP" button. Below this, it says "Currently displaying Do You Live In Canine Country? web map. Use the select a map button to choose a different web map." The "Options" section includes a "Title" input field, two checked checkboxes for "Enable Home Button" and "Enable Locate Button", and a "Title Background Color" selector with a color gradient and RGB/HEX input fields. At the bottom of the panel are "CANCEL" and "DONE" buttons.

Configure Button



The screenshot shows a listing page for a web mapping application. It features a small map thumbnail, the title "Where Dogs Are Treated Like Royalty", and the subtitle "Web Mapping Application by matt.driscoll". It also shows "Last Modified: March 6, 2014", "(0 ratings, 27 views)", and social media links for Facebook and Twitter. At the bottom, there is a row of action buttons: "OPEN", "SHARE", "EDIT", "DELETE", "MOVE", "CHANGE OWNER", "ATTACH CODE", and "CONFIGURE APP". A green line points from the "CONFIGURE APP" button in this screenshot to the "Configure Button" text in the adjacent green box.

Edit Application Item



Configurable Demo App

Web Mapping Application by matt.driscoll

Last Modified: March 6, 2014

☆☆☆☆ (0 ratings, 36 views)

Facebook Twitter

OPEN ▾

SHARE

EDIT

DELETE

MOVE ▾

CHANGE OWNER

ATTACH CODE

Configuration Parameters

Properties

URL

Purpose

API

Tags

Credits

Delete Protection Prevent this item from being accidentally deleted.

Extent

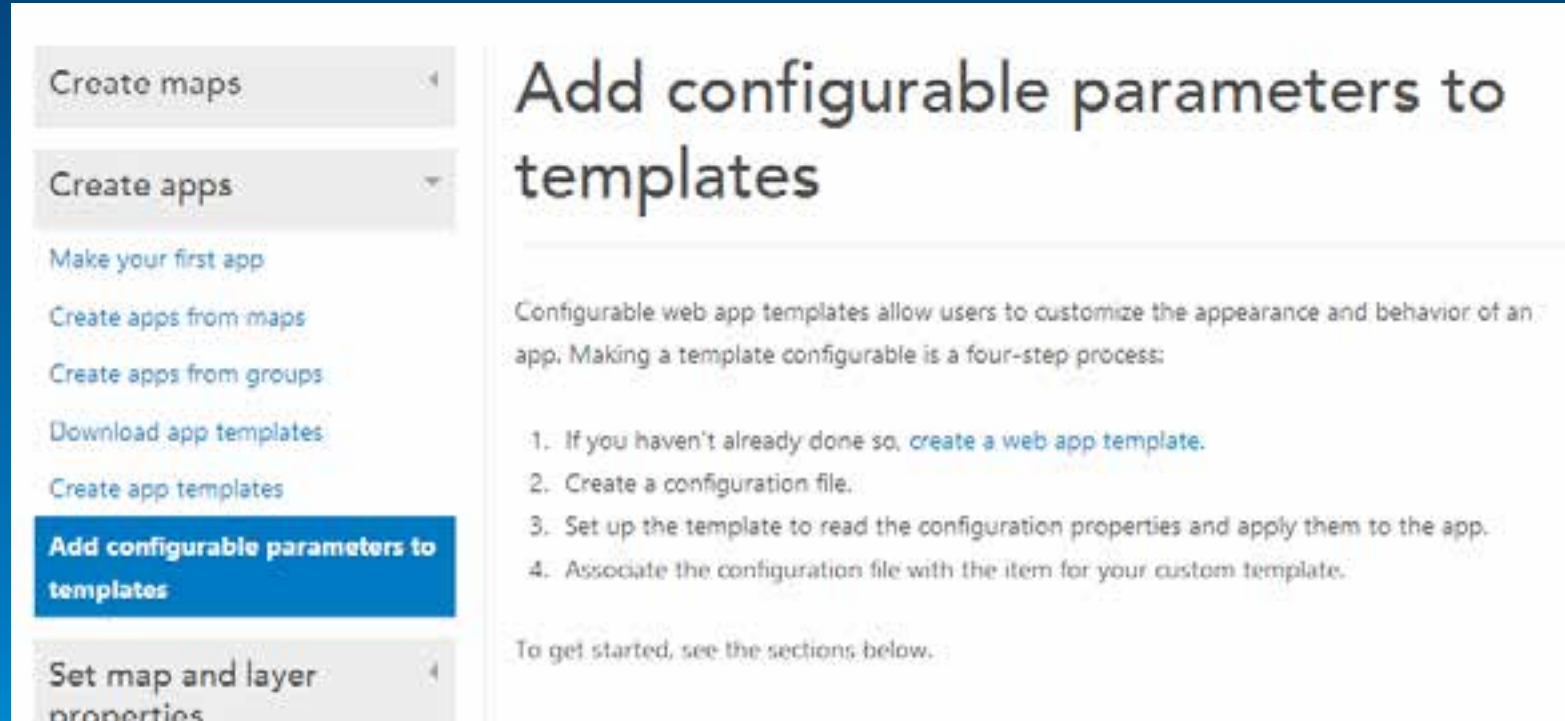
Configuration Parameters

{ JSON }

JSON!

```
{
  "configurationSettings": [{
    "category": "<b>Choose template theme</b>",
    "fields": [{
      "placeholder": "Defaults to map owner",
      "label": "Owner Text:",
      "fieldName": "owner",
      "type": "string",
      "tooltip": "Defaults to map owner"
    }, {
      "type": "string",
      "fieldName": "theme",
      "tooltip": "Color theme to use",
      "label": "Color Scheme:",
      "options": [{
        "label": "Chrome",
        "value": "chrome"
      }, {
        "label": "Seaside",
        "value": "seaside"
      }, {
        "label": "Pavement",
        "value": "pavement"
      }
    ]
  }
  ],
  "values": {}
}
```

Writing the Configurable JSON



The screenshot shows a navigation menu on the left and a main content area on the right. The navigation menu includes items like 'Create maps', 'Create apps', 'Make your first app', 'Create apps from maps', 'Create apps from groups', 'Download app templates', 'Create app templates', 'Add configurable parameters to templates' (highlighted in blue), and 'Set map and layer properties'. The main content area has a heading 'Add configurable parameters to templates' and a paragraph explaining that configurable web app templates allow users to customize appearance and behavior, followed by a four-step process.

Create maps

Create apps

Make your first app

Create apps from maps

Create apps from groups

Download app templates

Create app templates

Add configurable parameters to templates

Set map and layer properties

Add configurable parameters to templates

Configurable web app templates allow users to customize the appearance and behavior of an app. Making a template configurable is a four-step process:

1. If you haven't already done so, [create a web app template](#).
2. Create a configuration file.
3. Set up the template to read the configuration properties and apply them to the app.
4. Associate the configuration file with the item for your custom template.

To get started, see the sections below.

Show me the code!



Session Survey

Developing Custom JavaScript Apps with ArcGIS Online

<http://www.esri.com/events/devsummit/session-rater>



Understanding our world.