



# Web AppBuilder for ArcGIS: A Deep Dive in Enterprise Deployments

Nick Brueggemann

David Martinez

2019 ESRI DEVELOPER SUMMIT  
Palm Springs, CA

# Agenda

- Patterns
  - Installation
  - Configuration
  - Application Development
  - Deployment
  - Post-Deployment
- 



# Enterprise Deployment

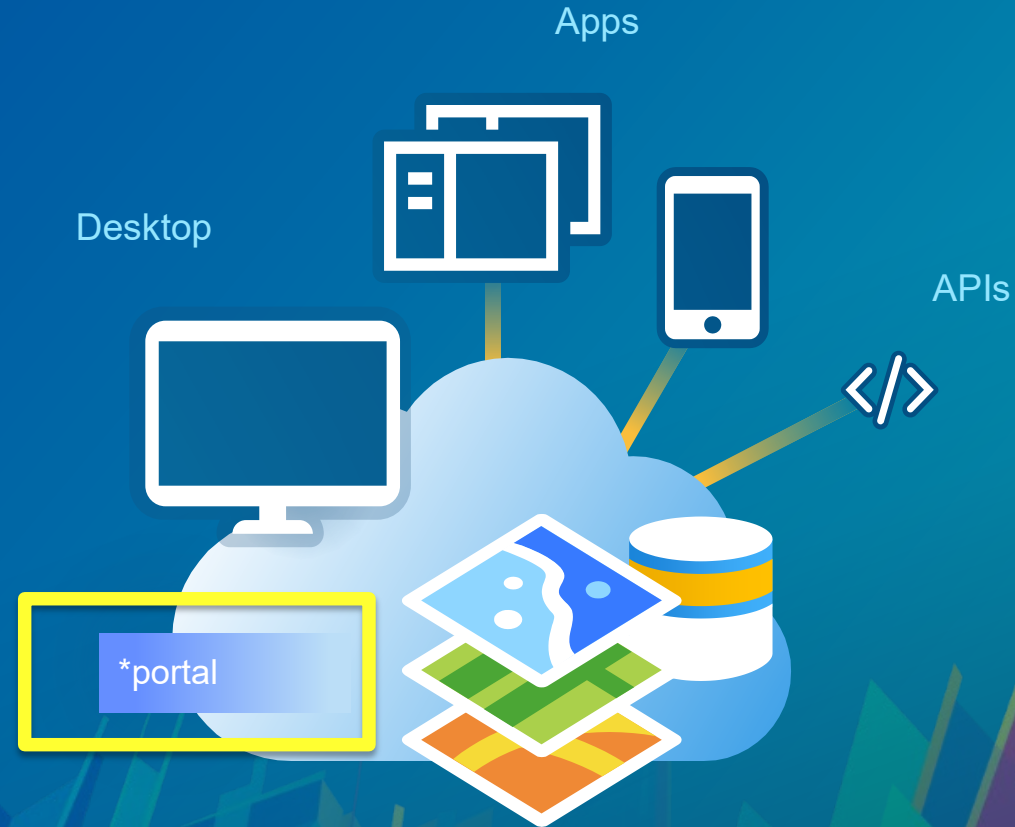
What is it and what does it mean?



# Implementation Patterns

## Web GIS

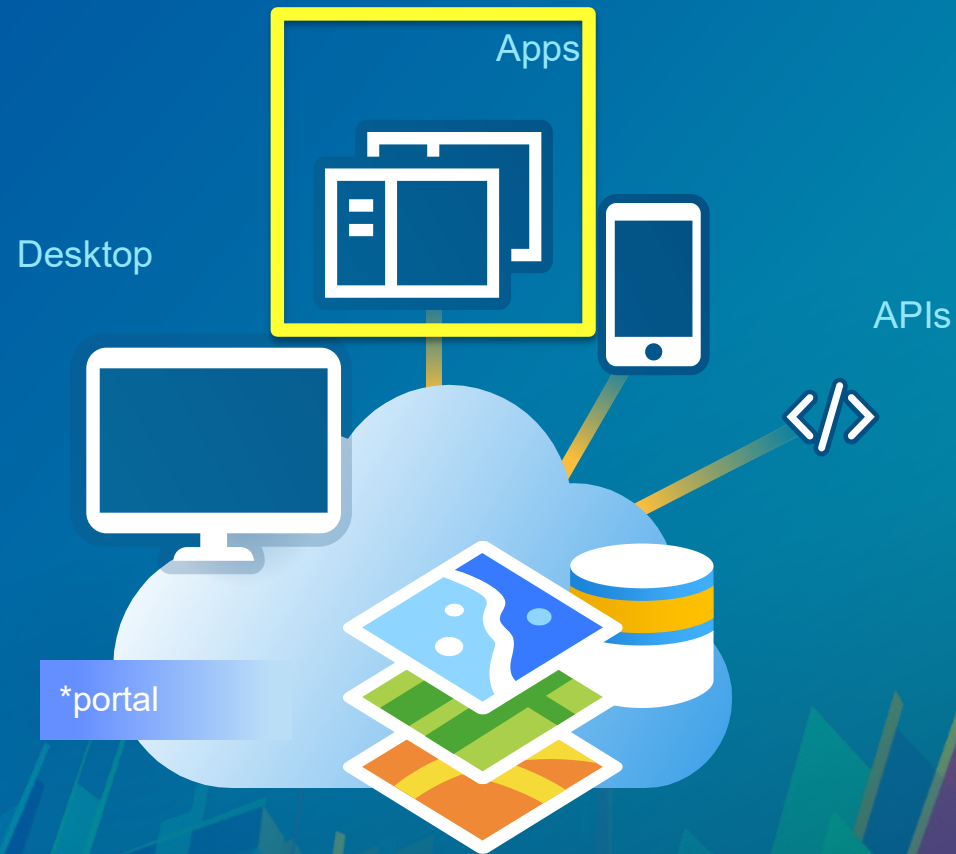
- ArcGIS Online
  - Esri Managed
  - Multi-tenant
  - Scale
  - Automatic Updates
- ArcGIS Enterprise
  - Your infrastructure
  - Customization
  - You control SLA
  - Integrated security
- Hybrid
  - Use technologies from both implementations



# Implementation Patterns

## Web AppBuilder

- Embedded Version
  - Hosted
  - Fully – integrated
  - Built In
- Developer Edition
  - Required for customization
    - Widgets
    - Themes
  - Direct Access
    - Doesn't require launching from a webmap
    - Widgets





# Installation

Getting Started...

# Installation...

Where do we get the JSAPI?





# Installation...

## Nodejs



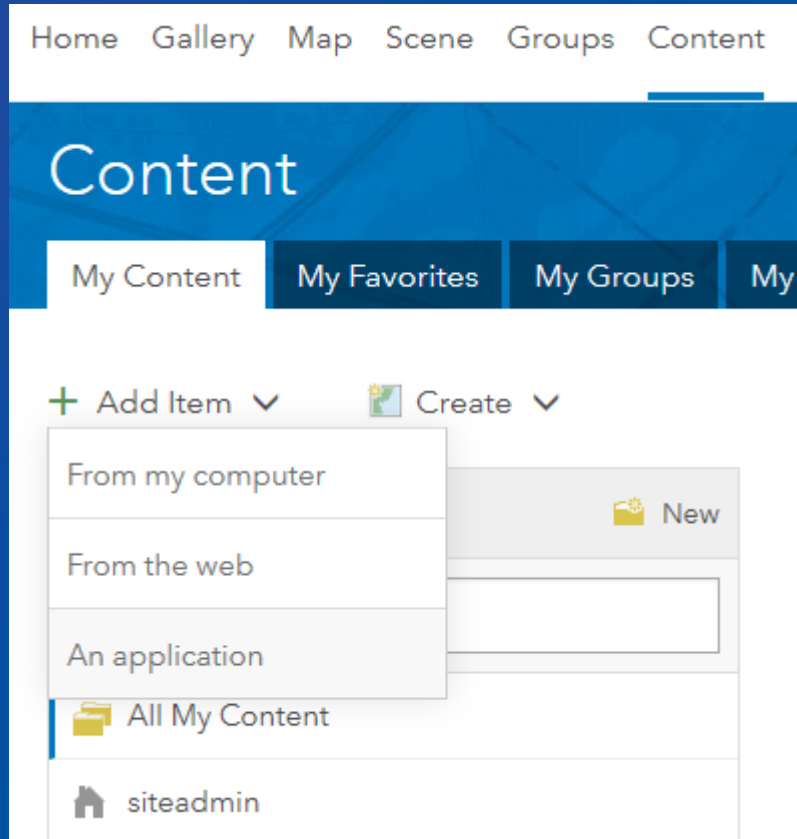
- Two options
  - Windows Service
  - Run as user
- As User
  - Startup.bat
  - Command Line
    - Add node.js to PATH
    - Navigate to <wab install>/server
  - node server.js
- Windows Service
  - Navigate to <wab install>/server
  - Install npm
  - npm run-script install-windows-service

```
c:\>cd WebAppBuilderForArcGIS2.11
c:\WebAppBuilderForArcGIS2.11>cd server
c:\WebAppBuilderForArcGIS2.11\server>node server.js
Server listening tcp connection on port 3344 in development mode
[2019-02-25T13:45:50.914] [INFO] repo - Read repository items: c:\WebAppBuilderForArcGIS2.11\client\stemapp\widgets
Server listening http connection on port 3345 in development mode
Server listening https connection on port 3346 in development mode
[2019-02-25T13:46:00.905] [INFO] repo - Read repository items: c:\WebAppBuilderForArcGIS2.11\client\stemapp\themes
[2019-02-25T13:46:02.675] [INFO] repo - Read repository items: c:\WebAppBuilderForArcGIS2.11\client\stemapp3d\widgets
[2019-02-25T13:46:05.105] [INFO] repo - Read repository items: c:\WebAppBuilderForArcGIS2.11\client\stemapp3d\themes
.....Repository items refreshed.....
c:\WebAppBuilderForArcGIS2.11\server>node server.js
Server listening tcp connection on port 3344 in development mode
```



# Installation...

Adding the app to portal



## Add an application ?

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

Type:

- ☐ Web Mapping
- ☐ Mobile
- ☐ Desktop
- ☒ Application
- ☐ Application Extension (Operations Dashboard)
- ☐ Application Extension (AppBuilder)

Title:

Web AppBuilder

Tags:

WAB × web appbuilder × appid ×

Add tag(s)

Add Item Cancel

# Installation...

Registering the app with portal (not necessary on web-tier-authenticated portals)

## Web AppBuilder

Overview Settings

General Settings Application Settings

### General Settings

#### Delete Protection

☐ Prevent this item from being accidentally deleted.

Delete Item

#### Extent

No extent has been set for this item.

Set Extent

Save

Cancel

### Application Settings

#### Data Source

URL

#### App Registration

Update

Unregister App

Registered Info

## Registered Info



App ID:

UtxmlforE35tjX6h

App Secret:

[Show Secret](#)

App Type:

Multiple

Redirect URI:

https://<server>[:port]

Add

Delete

urn:ietf:wg:oauth:2.0:oob

Update

Reset Secret

Cancel

# Installation...

## OAuth

### Updated Item Page

Application Settings

Data Source

URL

App Registration

App ID:  
UtxmlforE35tjX6h


App Secret:  
[Show Secret](#)

App Type:  
Multiple


Redirect URI's:  
urn:ietf:wg:oauth:2.0:oob

### Web AppBuilder Authorization

Specify the URL to your organization or Portal for ArcGIS



Provide the App ID for Web AppBuilder registered in the portal you specify above ( ? )



[Continue](#)

[Update](#)[Unregister App](#)

Signininfo.json

```
[{  
  "portalUrl": "http://[webserver]/[webadaptor]",  
  "appId": "UtxmlforE35tjX6h",  
  "supportsOAuth": true,  
  "isWebTier": false  
}]
```





# Configuration

Updating the defaults

# Configuration...

Sharing the app to your portal

Home page


## Share Item

Home Gallery Map Scene Groups Content Organization

### Web AppBuilder (Dev Edition) [Edit](#)

Overview Settings

[Edit Thumbnail](#)



Add a brief summary about the item.

by siteadmin

[Web Mapping Application](#)

★ Add to Favorites

#### Description

Add an in-depth description of the item.

#### Access and Use Constraints

Add any special restrictions, disclaimers, terms and conditions, or license information.

#### Comments (0)

#### Share

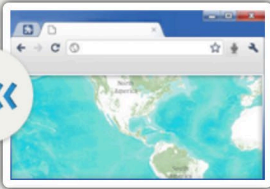
Share the item(s) with:

- ☒ Everyone
- ☒ Portal for ArcGIS
- ☒ These groups:
  - ☐ Dev Summit
  - ☒ Featured Maps and Apps

These settings will replace the current settings.

[OK](#) [Cancel](#)

#### Featured Maps and Apps



Web AppBuilder

### Tags

[Edit](#)

web, app, builder

### Credits (Attribution)

[Edit](#)

Acknowledge this item's source.

### URL

[View](#)

<https://clt-mtorrey10l.esri.com:3344/web>



# Configuration...

## Disabling embedded Web AppBuilder

**Share**

**Choose who can view this map.**  
Your map is currently shared with these people.

☒ Everyone  
☒ Portal for ArcGIS  
☐ Members of these groups:

☐ Featured Maps and Apps  
☐ Web App Templates

**Link to this map**  
[http://\[redacted\].esri.com/portal/home/webmap/viewer.html?webmap=09f80339f939442f9df43d5ef4f06ce7&extent=-126.6804,21.005,-81.3288,56.196](http://[redacted].esri.com/portal/home/webmap/viewer.html?webmap=09f80339f939442f9df43d5ef4f06ce7&extent=-126.6804,21.005,-81.3288,56.196)

☒ Share current map extent

**Embed this map**

EMBED IN WEBSITE CREATE A WEB APP

**Create a New Web App**

Configurable Apps Web AppBuilder

To create a new app with Web AppBuilder, enter a title, tags and summary.

Title: Demo Map: Cropland vs Household Size

Tags: demo x  
Add tag(s)

Summary: Enter a summary (Optional)

Save in folder: asizer\_ijcdev

☒ Share this app in the same way as the map (Everyone)

webmap > Save > Share

Create a Web App > Web AppBuilder tab



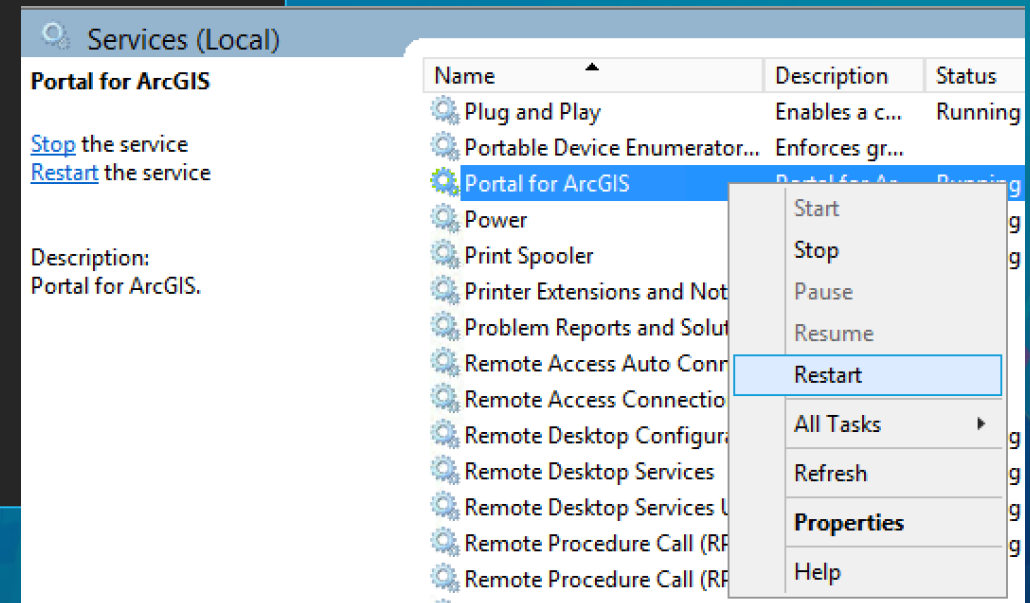
# Configuration...

## Disabling embedded Web AppBuilder...

C:\Program Files\ArcGIS\Portal\customizations\  
10.6.0\webapps\arcgis#home\js\arcgisonline\config.json

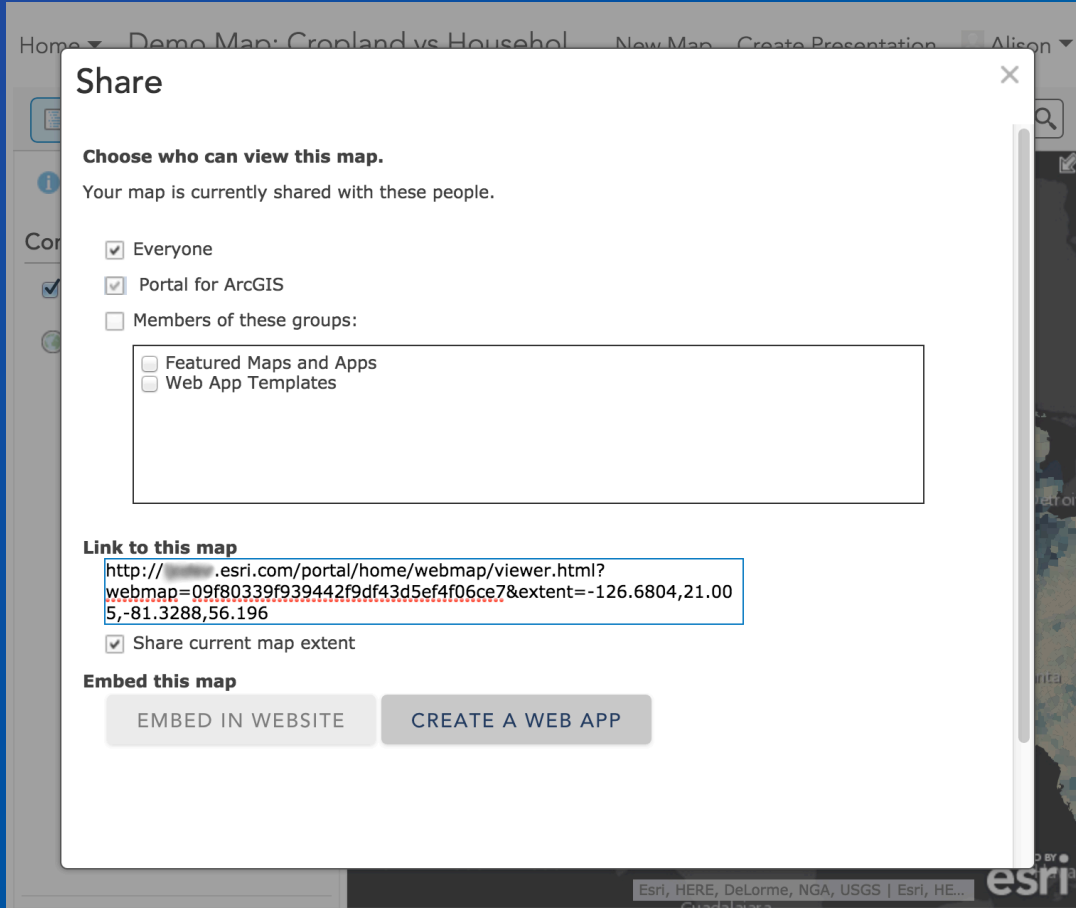
```
...  
searchArcGISOnlineEnabled: true,  
publishTilesFromFeaturesEnabled: true,  
federatedServerConfigEnabled: true,  
hostedServerConfigEnabled: true,  
findFeaturesWebMapEnabled: true,  
webAppBuilderEnabled: true,  
sceneViewerEnabled: true,  
restrictOrganizationPageToAdmin: false, ...
```

## Restart Portal Service

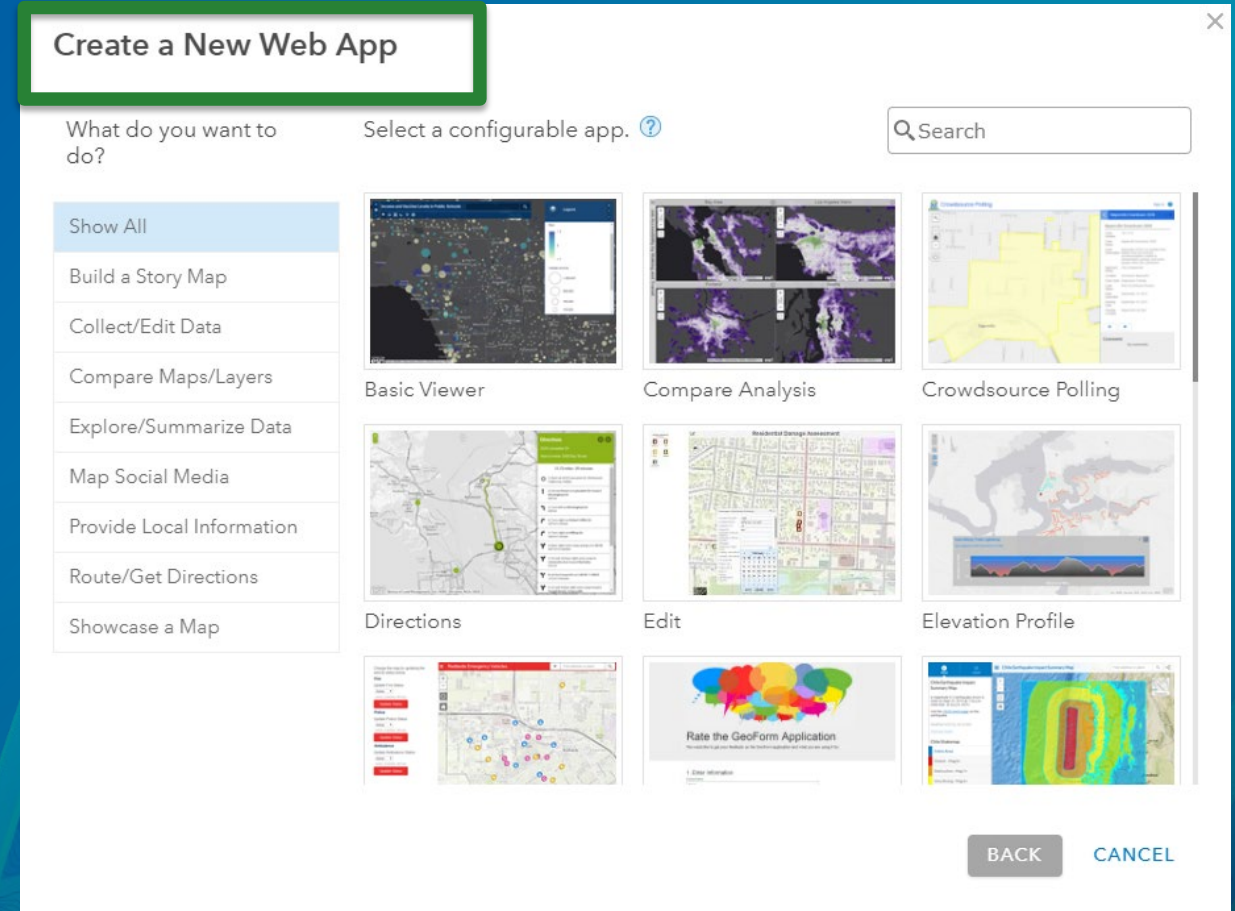


# Configuration...

## Disabling embedded Web AppBuilder...



webmap > Save > Share

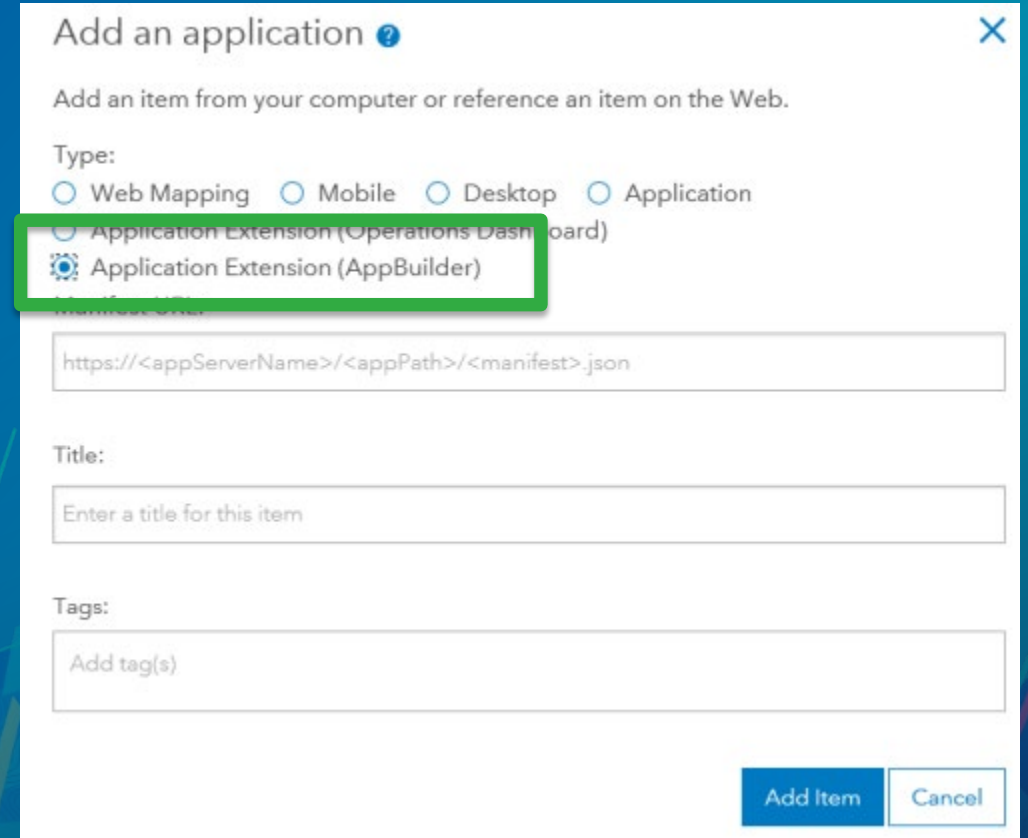
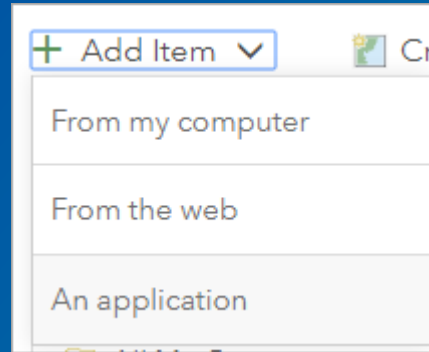


Create a Web App > Web AppBuilder tab

# Configuration

Add custom widgets to you portal

- Add virtual directory
  - Enable
  - anonymous access
  - HTTPS
  - CORS
  - JSON Handler
- Register widget manifest
- Share to Users

A screenshot of a dialog box titled 'Add an application'. The dialog box contains the following fields and options:

- Type:** A group of radio buttons with the following options: 'Web Mapping', 'Mobile', 'Desktop', 'Application', 'Application Extension (Operations Dashboard)', and 'Application Extension (AppBuilder)'. The 'Application Extension (AppBuilder)' option is selected and highlighted with a green box.
- Manifest URL:** A text input field containing the placeholder text 'https://<appServerName>/<appPath>/<manifest>.json'.
- Title:** A text input field containing the placeholder text 'Enter a title for this item'.
- Tags:** A text input field containing the placeholder text 'Add tag(s)'.
- Buttons:** At the bottom right, there are two buttons: 'Add Item' and 'Cancel'.



# Configuration...

## Use Your Certificates

- **WAB ships with a self-signed cert**
  - Functional, but not ideal for enterprise users, not allowed elsewhere
- **You can use your own certificate and replace the default cert**
  - Domain or commercially-signed “CA cert”
  - You’ll need server certificate and key pair
- **Create .pem files**
  - Convert .pfx certificate into a certificate/key pair
  - Replace cacert.pem and cakey.pem in \server
  - Restart WAB

```
08/27/2018 03:39 PM      1,344 cacert.pem
08/27/2018 03:39 PM      1,702 cakey.pem
08/27/2018 03:39 PM        294 config.json
02/26/2019 10:15 AM    <DIR>      db
08/27/2018 03:39 PM      1,668 db-engine.js
01/09/2019 06:00 PM        806 dbutils.js
08/27/2018 03:39 PM      92,148 license.txt
```



# Enterprise Development

# Enterprise Development

What do we mean?

Web App Builder Applications that are of moderate to large complexity and have the following similar characteristics:

- More than a single developer contributing
- Running into design challenges with code in WAB widgets.
- Deployed to multiple environments (dev, test, prod)
- Has a mix of custom widgets, modified out of the box widgets, and custom application logic
- Automated unit testing is necessary to test application
- CI/CD pipelines are in place to automate execution of tests and automate deployment



# Enterprise Development

What do we mean?

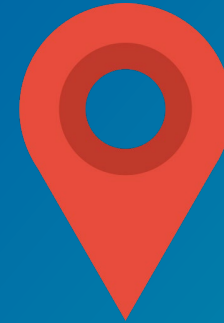


No coding  
just config



Dev Edition

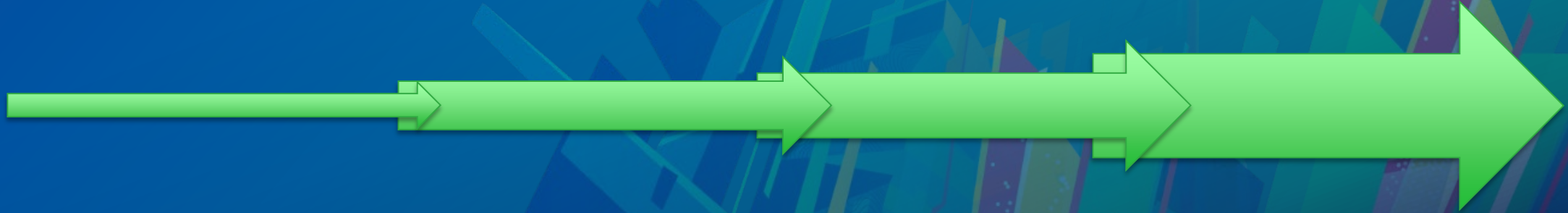
Custom  
Widgets



You Are  
Here



Custom  
App



# Enterprise Development

How do we end up here?

## Stakeholder Driven

We chose Web App Builder because all we initially needed was an application with widgets that WAB already had. This allowed us to get an app out the door fast.

After initial application launch stake holders asked for some widgets that were specific to our business case so we wrote a few widgets to meet this need.

Stakeholders then started asking for changes to be made to some of the widgets we didn't write. We pushed back because it "wasn't our code" but that was what they wanted so we made the changes.

Complexity of application increases to a point where we question if we should have gone with a completely custom application.

# Enterprise Development

How do we end up here?

## Developer Driven

We prefer to keep user interface logic and application logic separate. Therefore it doesn't seem right that all the custom logic we write ends up in widgets.

We've written a bunch of utility methods that are contained in individual widgets. We want that code to not reside in the widget itself

We have logic that multiple widgets share but do not want to duplicate that logic in each widget.

We need some application initialization logic to occur when the app starts up but we do not want to have to display a widget to get that logic to execute.

# Code Beyond Widgets

## Scenario 1

During the course of writing the code for our custom widget we've written a few utility methods that operate on arrays.

The first method will take an array and give us back a list of unique values. The returned array will have no duplicates.

```
getUniqueValues: function (array)
```

The second method will take 2 arrays and return back an array of items that exist in the first array passed in but not the second.

```
difference: function (array1, array2)
```



# Code Beyond Widgets

## Scenario 1

### MyCustomWidget

#### Widget.js

```
getUniqueValues(array)  
difference(array1, array2)
```

#### arrayUtilities.js

#### arrayUtilities.js

```
getUniqueValues(array)  
difference(array1, array2)
```

### MySecondCustomWidget

#### Widget.js

# Code Beyond Widgets

## Scenario 2

### Goal:

Our application needs to allow a user to do some processing on a portion of their data that falls within a spatial area of their choosing.

We want a custom filtering widget that allows a user to define a spatial filter by drawing a polygon on the map.

Initial app startup the spatial filter should be set to the extent of the map.

After spatial filter has been modified the application should start back up with the user defined filter polygon drawn to the map (saved state).

# Code Beyond Widgets

## Scenario 2

Initial thought is that we'll create a widget that creates a graphics layer that will hold the user drawn spatial filter. On startup on the widget we'll create this layer.

...but we want the application to startup with the filter already drawn to the map. We likely don't want to have to require that this filter widget open up in order to do this.

Well we at least need to be able to save the spatial filter to persistent storage so that when the app starts up again we can return the app to using the same filter. Maybe we can put that in the custom widget?

...but this sounds a whole lot more like application logic and we may not want to put that into the widget itself.

# Code Beyond Widgets

## Scenario 2

**MyCustomWidget**

**Widget.js**

**FilterManager**

**graphicsLayer**

**initialize()  
saveFilter()  
loadFilter()**



# Code Beyond Widgets

## Scenario 2

If it's not the custom widget that initializes the filter manager then what does?

2 options:

### Controller Widget

- A custom widget that is configured to never display. It has no UI but is setup to start when the application starts.
- This widget would call initialize on the FilterManager which would allow it to load the saved spatial filter and draw it to the graphics layer without needing our custom filter fidget to be opened.

### Hooks into WAB Startup

- Logic "injected" into the startup of WAB itself.
- Can be done by extending core WAB classes and invoking your application startup logic.
- Controller is easier but this approach allows for more control on the timing of startup logic execution.

# Code Beyond Widgets

Where does all of this code that we're not putting into widget go?

We place it into a folder inside WAB called libs. This is the location WAB itself places other libraries and we're treating our application logic as another library our application depends on.



# Code Beyond Widgets

## Drawbacks

- It will be much more difficult to share just one of your widgets with another project since your logic is not contained in just the widget itself.



# Source Code Management Paradigms

- Essential for team collaboration
- There are a few different ways to do this
- We have a recommended approach but one size doesn't always fit all.





# Source Code Management Paradigms

## No Source Control



## Source Control Everything

- arc-gis-web-appbuilder
  - client
    - stemapp
    - ...
  - docs
  - server

## Source Control a Single Application

- arc-gis-web-appbuilder
  - client
    - stemapp
    - ...
  - docs
  - server
  - apps
    - 2

## Source Control Only your Code

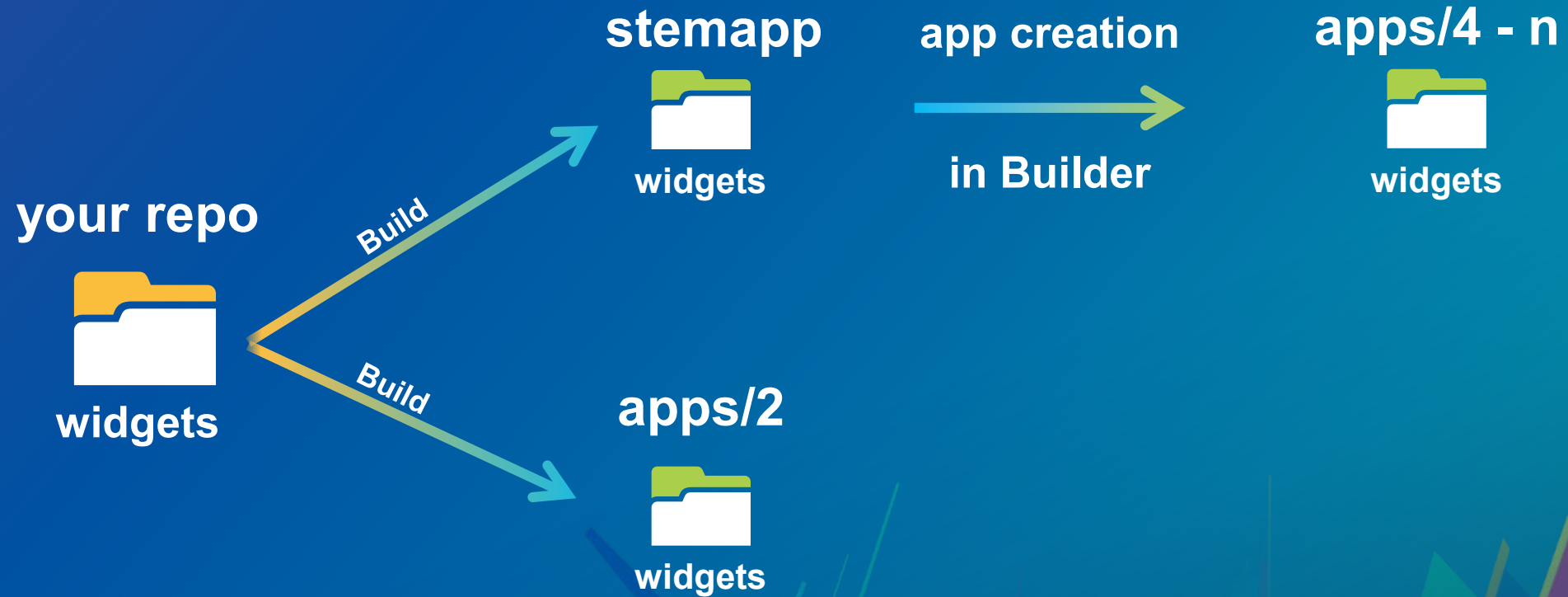
- my-project
  - build
  - src
    - custom
    - libs
  - widgets
  - wab \*

\* Added to gitignore



# App Build Process

Overview



**GRUNT**

# App Build Process

## Multi-Environment Configuration

- Typically configurations point to services your widget depend on
- There may need to change for deployments to different environments

### Typical Config

```
{  
  "service": {  
    "url": "https://sampleserver6.arcgisonline.com/arcgis/rest/serv..."  
  }  
}
```

### But We Need

```
{  
  "service": {  
    "url": "dev-service-url"  
  }  
}
```

```
{  
  "service": {  
    "url": "staging-service-url"  
  }  
}
```

```
{  
  "service": {  
    "url": "prod-service-url"  
  }  
}
```

# App Build Process

## Multi-Environment Configuration

- We use gulp-replace as a way to build these configurations for us
- Target one env at a time: gulp -env=dev (yargs for arguments)





# Unit Testing

## Overview

Has great benefits but often overlooked due to pressure to deliver features. Each sprint produces more. Eventually, manual testing can not keep pace.



Automating tests can help ensure regression testing is done.

*When was the last time we tested that feature we put in place back in sprint 1?*

Many test scenarios are just easier to test with unit tests. Examples:

- Failure cases for web requests
- User input

# Unit Testing

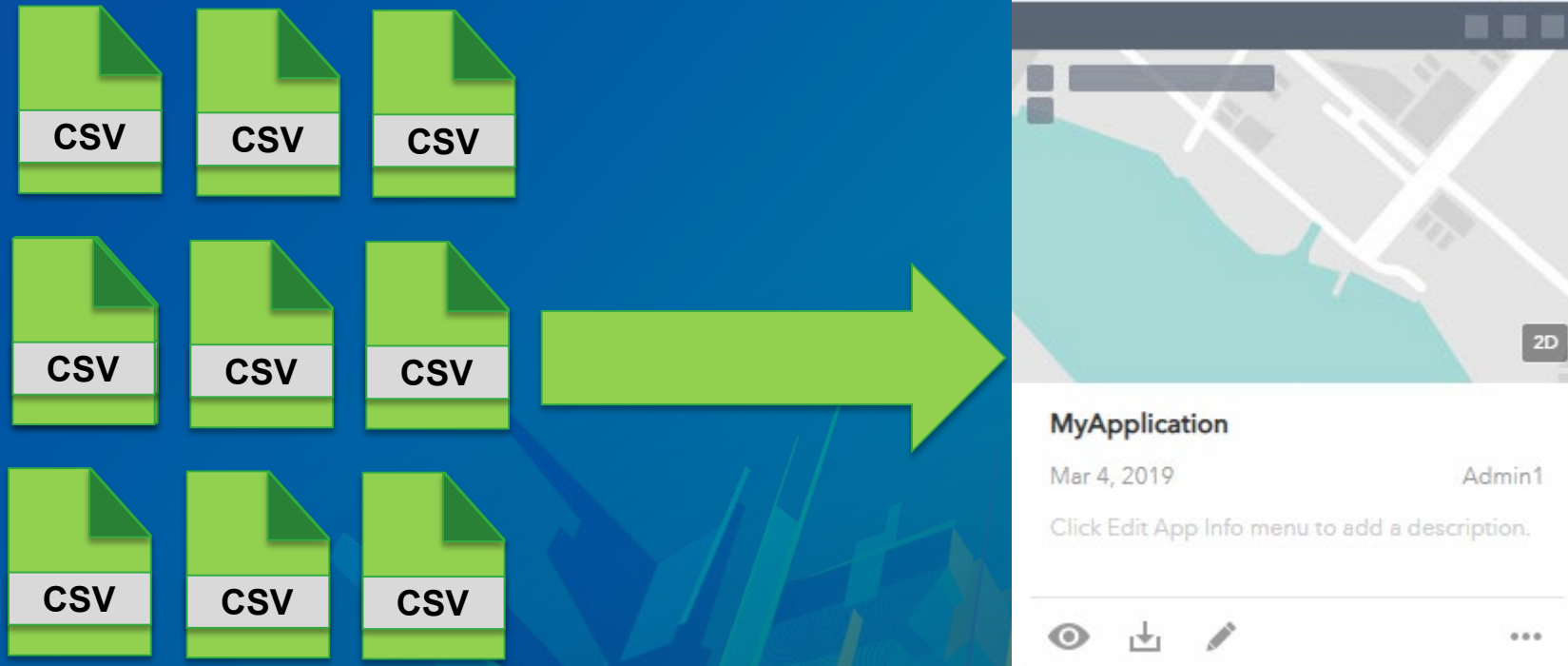
## Example

Allow a user to upload a csv file that contains a feature per row. Each feature contains a time value that is in either 1 columns or 2 columns. If the time is in 1 column the value will be the date and the time. If the time is in 2 columns one column will be the date and the other will be the time. We need to support values in both ISO8601 and RFC 2822 in addition to one custom format. This feature data needs to be added to a published feature service we have.

# Unit Testing

## Manual Approach to Testing

If we were testing this manually how would we do it?



# Unit Testing

Unit Test Approach to Testing

If we're testing this with unit tests how would we do it?





# More Sessions

## Javascript Applications for the Enterprise: Building for Production

Pasadena/Sierra/Ventura      **Wednesday, March 6 4:00 – 5:00 pm**

Randy Jones and George Bochenek

## Unit Testing with the Javascript API

Catilina/Madera      **Thursday, March 7 1:00 – 2:00 pm**

Randy Jones and George Bochenek

## Web AppBuilder for ArcGIS: Advanced Development Tools and Techniques

Catilina/Madera      **Thursday, March 7 5:30 – 6:30 pm**

Gavin Rehkemper and George Bochenek

# Deployment

The background features a vibrant blue gradient. On the right side, there are several diagonal stripes in shades of green, yellow, and pink. A faint, stylized map of Europe is visible in the lower right corner, overlaid on these stripes. In the top left corner, there are some abstract, overlapping geometric shapes in green, yellow, and pink.

# Proxies

## Overview

- Cross Domain Resources
- Oauth 2.0 app logins
- Referrer limiting
- Rate Limiting
- Token Based authentication
- logging



php



.NET



Java

**<https://github.com/esri/resource-proxy/>**

Deploy to the same web server as your Deployed App

# Proxies

Configure Web AppBuilder App to use proxy

## [downloaded-app]/config.json

```
...  
"httpProxy": {  
  "useProxy": true,  
  "alwaysUseProxy": false,  
  "url": "",  
  "rules": [{  
    "urlPrefix": "http://yourwebserver/yourwebadaptor/sharing/rest",  
    "proxyUrl": "http://yourproxyfolder/proxy.php"  
  }]  
}  
...
```



# Proxies

Accessing Secure Services

## [proxy-folder]/proxy.config

```
<serverUrls>
  <serverUrl
    url="http://services.arcgisonline.com"
    matchAll="true"
  />
  <serverUrl
    url="http://webserver/portal/sharing/rest"
    matchAll="true"
    clientid="xxxyyy12345"
    clientsecret="aaabbbccc987654321"
    oauth2Endpoint="https://webserver/portal/
                    sharing/rest/oauth2/token"
  />
</serverUrls>
```

# What about App Deployment

Did you know you can use Python API to clone items?



# Other WAB Presentations

Check These Out!

Web AppBuilder for ArcGIS: Advanced Development Tools and Techniques

Thursday 5:30 pm - 6:30 pm Catalina/Madera

Gavin Rehkemper and George Bochenek

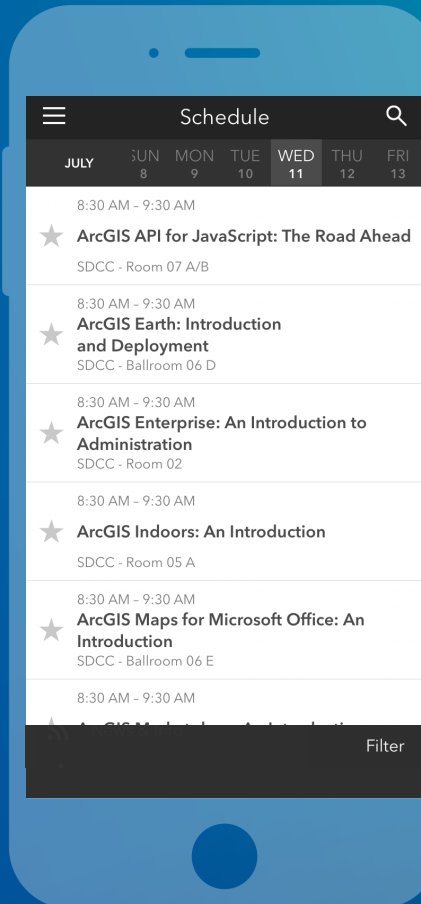


# Please Take Our Survey on the App

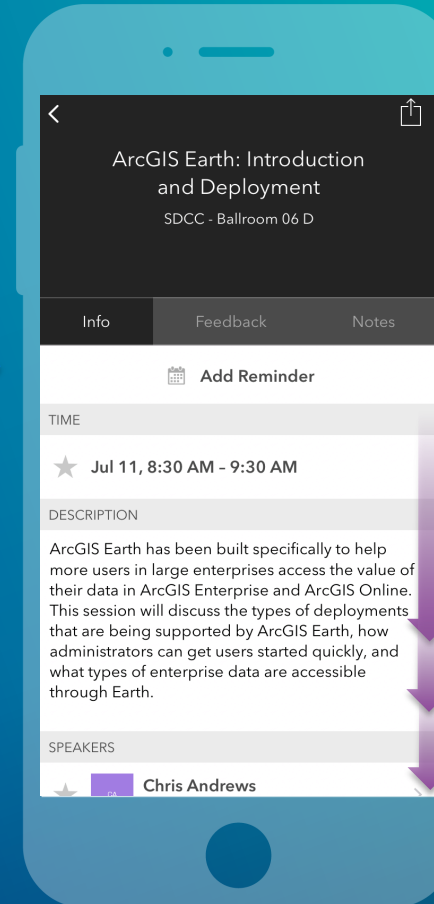
Download the Esri Events app and find your event



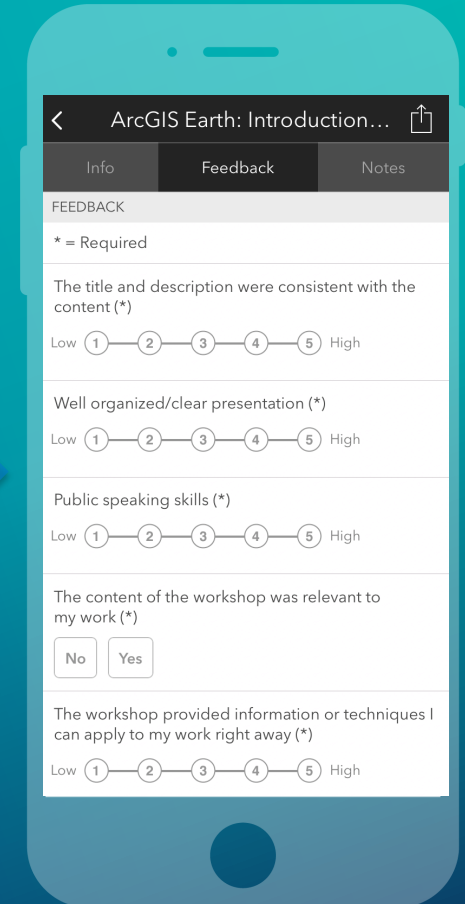
Select the session you attended



Scroll down to find the feedback section



Complete answers and select "Submit"







esri

THE  
SCIENCE  
OF  
WHERE