Developing Imagery Apps using the ArcGIS API for JavaScript and Web AppBuilder

Chayanika Khatua

Ziqian Ming

2019 ESRI DEVELOPER SUMMIT
Palm Springs, CA

Session goals

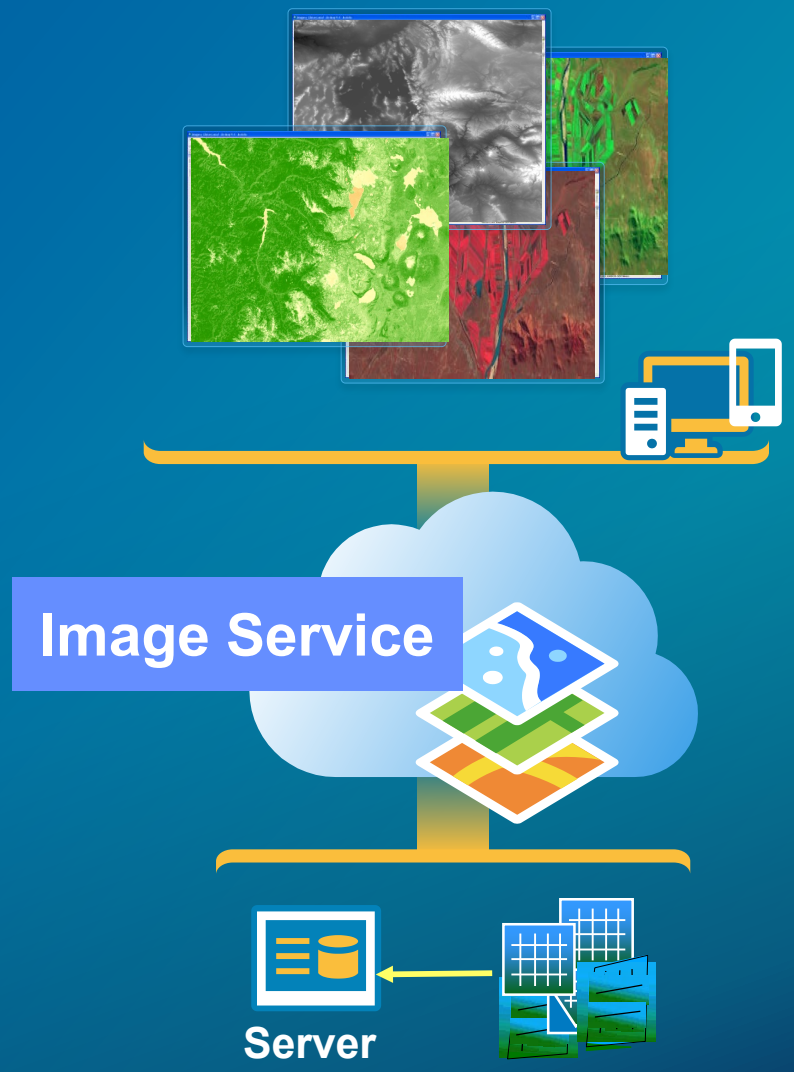
- Cover core concepts that allow you to leverage the power of raster data in your web applications.
- Provide an overview of the various ways to create Imagery Apps on the ArcGIS platform.
- What might be the best combination for you.

What we are talking about today

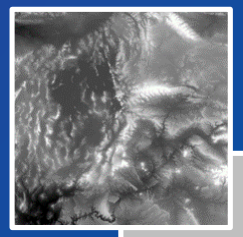
- Image Services
- ArcGIS API for JavaScript
- **Map viewer in ArcGIS Enterprise**
- **Web AppBuilder**
- **Configurable App Templates**

Image Services

- **Data**
  - shared through ArcGIS Server
  - accessed by any device that supports connecting to a web service
  - Single image or a collection of images
- **Visualization**
  - Fast dynamic display
- **Processing**
  - Well-known server-side raster functions
  - Publish processing models with the service



ImageServices



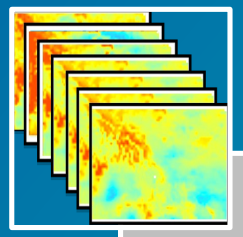
Raster Dataset

- Format
- Interpolation
- Rendering Rule

- GetRasterAttributeTable
- GetKeyProperties

Mosaic Dataset (MD)

- +
- MosaicRule
- QueryVisibleRasters



MultiDimensional MD

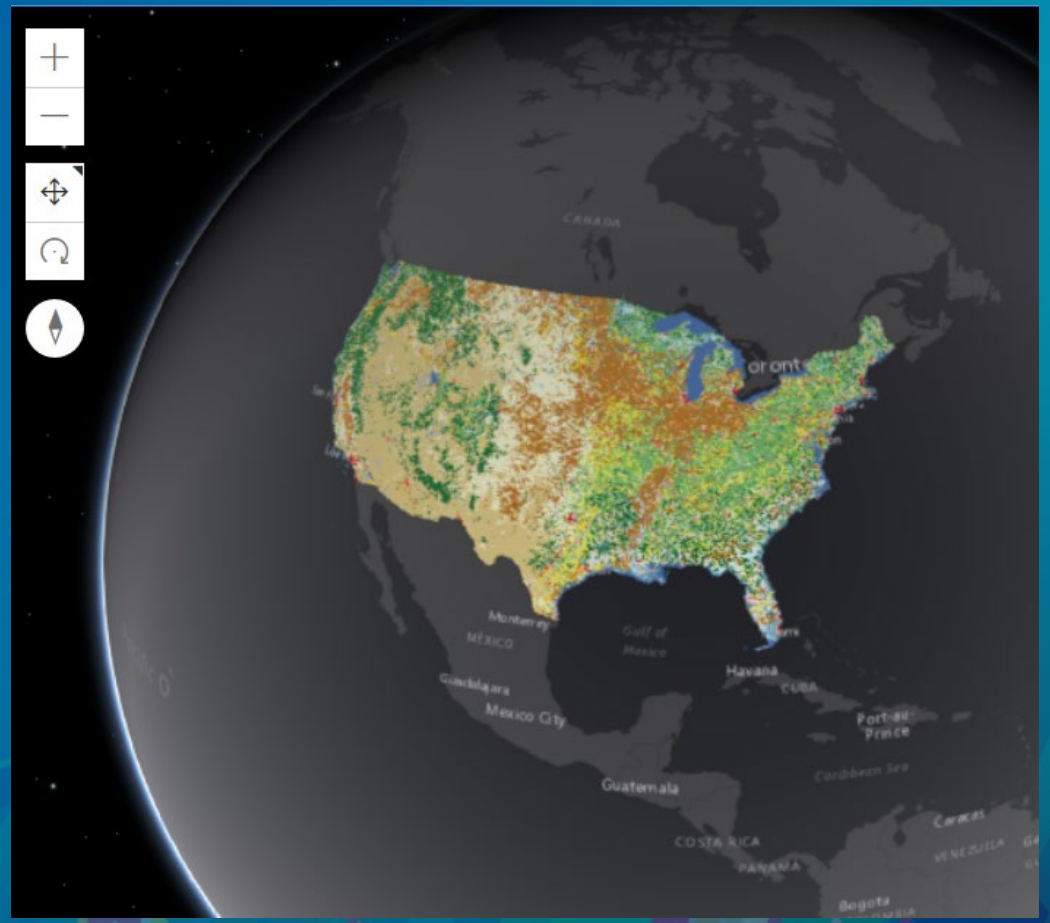
- +
- MultiDimensional Info
- Dimensional Definition

ArcGIS API for JavaScript

4x

Overview

- Build 2D and 3D web apps
- Integrate seamlessly with your web GIS
- New architecture with views and models
- Build better user experiences
- Modern browser support: IE11+



ImageryLayer

- Data source is an image service
- Renders image data
- Supports 2D and 3D visualization
- Retrieves data as pixels, enabling client side processing
- Server side processing using Raster Functions
- Provides Multi-dimensional API to view Scientific data
- View popups and query capabilities

Using an ImageryLayer

esri/layers/ImageryLayer

- Data source is an image service
- Image service parameters: compressionQuality, format, opacity

```
var layer = new ImageryLayer({
  url: "https://myServer.arcgisonline.com/arcgis/rest/services/NLCDLandCover/ImageServer"
});

/*************************
 * Add image layer to map
 *************************/

var map = new Map({
  basemap: "gray",
  layers: [layer]
});

var view = new MapView({
  container: "viewDiv",
  map: map,
  center: [-100, 40],
  zoom: 5
});
```

ImageryLayer: MosaicRule

esri/layers/support/MosaicRule

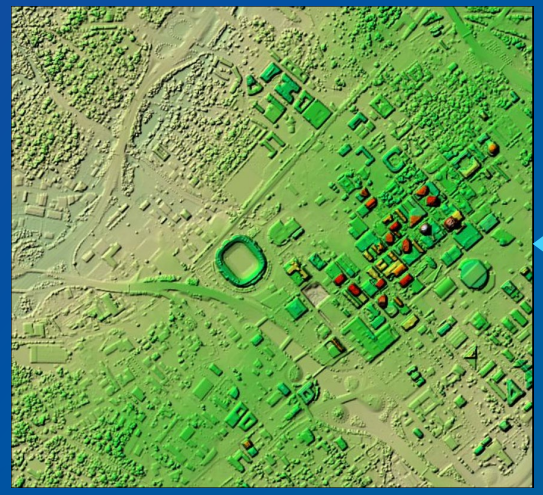
- MosaicRule defines how overlapping images are mosaicked together
- Method: none | center | nadir | viewpoint | attribute | lock-raster | northwest | seamline
- Select a subset of images using the where clause
- Sort using sort field and value

```
// Define the way overlapping images are mosaicked together
var mosaicRule = new MosaicRule({
  ascending: true,
  method: "center",
  operation: "last"
});
```

ImageryLayer: Server side processing

esri/layers/support/RasterFunction

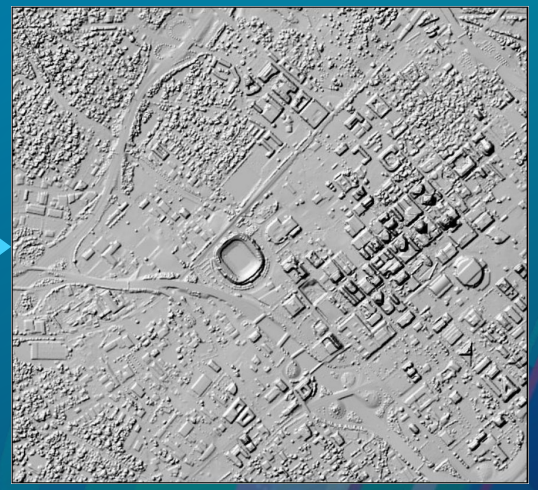
- Applied on the on the fly as renderingRule
- Wide range of supported server raster functions
- Can be used to together to perform complex processing
- See list http://esriurl.com/ServerRasterFunctions



Shaded Relief

fx

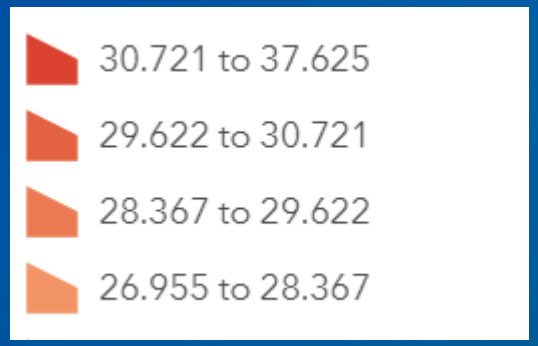
fx

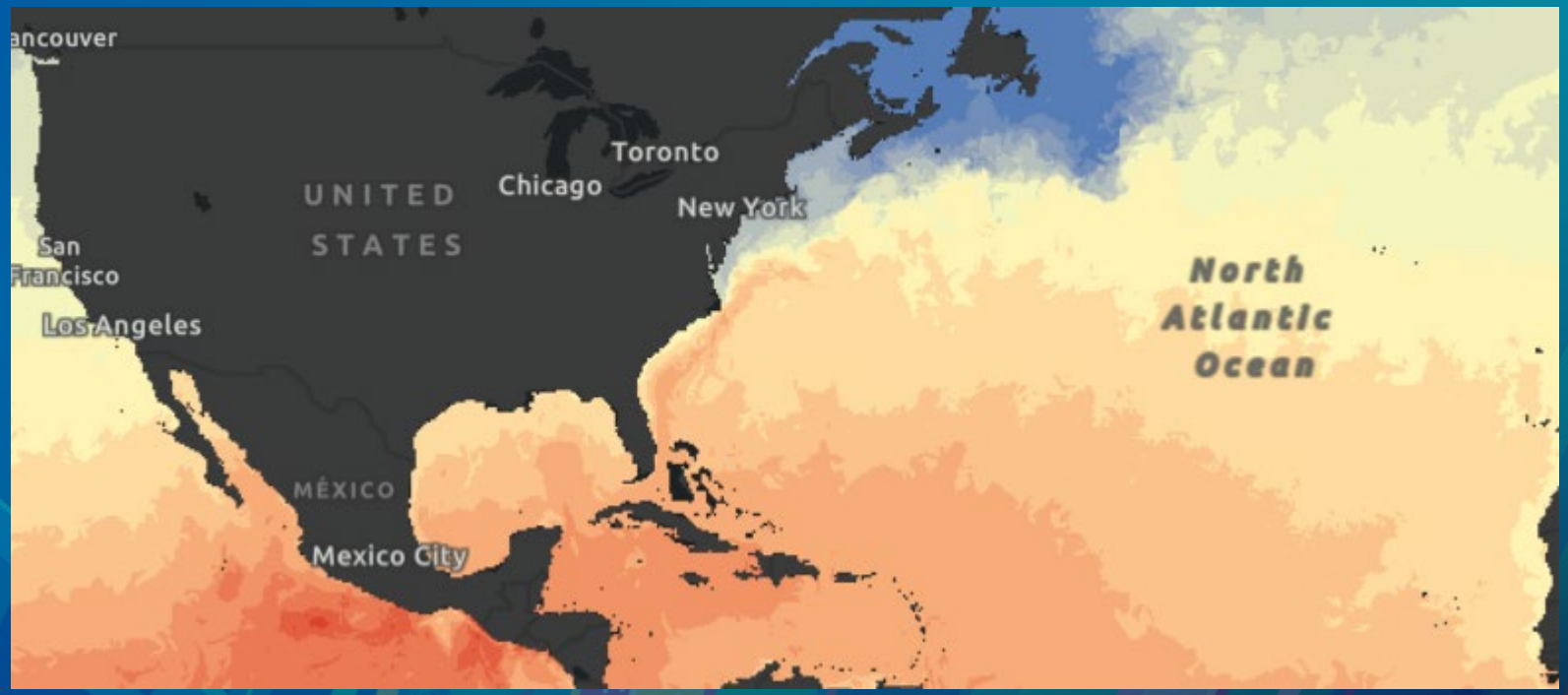


Hillshade

ImageryLayer: Renderer

- Being added in 4.11, upcoming release
- Renderers define how the layer should be rendered
- Available renderers:
  - Stretch Renderer: new in 4.11
  - Class Breaks Renderer
  - Unique Value Renderer





ImageryLayer: Client side processing

- Perform processing or custom rendering using the pixelFilter and PixelBlock API
- Create highly interactive experiences
- Consideration: cannot be saved with your webmap

PixelBlock

- Width: number of columns
- Height: number of rows
- Pixels: two dimensional array
- PixelType: s8 | s16 | s32 | u8 | u16 | u32 | f32 | f64
- Mask: array of 0 or 1
- Statistics: array of objects containing the min and max values

ImageryLayer: MultidimensionalData

esri/layers/support/MultidimensionalDefinition

- Filters data based on slices or ranges in one or more dimensions
- A multidimensional layer has multidimensionalInfo

ArcGIS API for JavaScript

3x

Layer Types

Raster

- Renders image data
- Retrieves data as pixels (PNG, JPEG, TIFF, LERC etc.)
- Processing using raster functions
- Processing using pixel filters

…

ArcGISImageService

- Renders image data
- Retrieves data as an image (PNG, JPEG, TIFF, etc.)
- Processing using raster functions

…

ArcGISImageServiceVector

- Renders vector data
- Retrieves data as pixels
- Supports scientific data
- Define Symbology

…

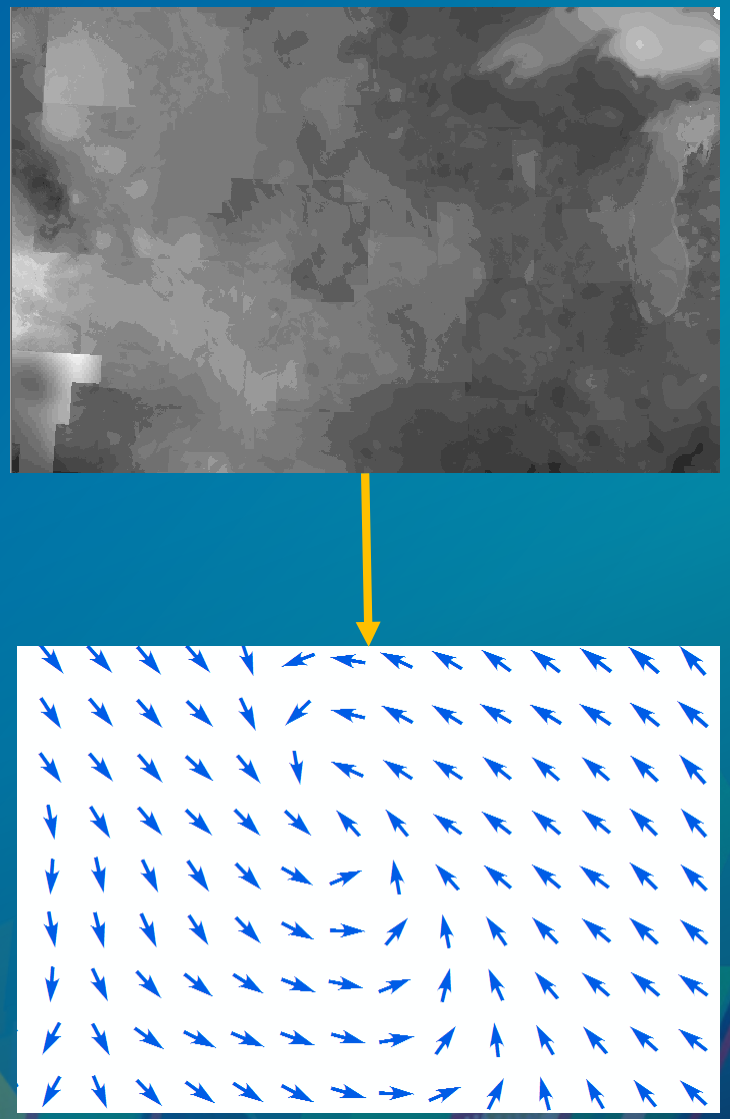
WCS

- Renders raster data using OGC WCS specs
- Processing using client side filters

ImageServiceVectorLayer

- Visualize vector data: magnitude and direction
  - Wind, Ocean current
- Renders raster data as vectors using VectorFieldRenderer
  - *Single Arrow, Simple Scalar, Wind Barbs, Beaufort Wind, Ocean Current*
- Uses a pixelFilter for conversion of U-V to Magnitude-Direction

```
Var layer = new ArcGISImageServiceVectorLayer(url,
    {
      imageServiceParameters: params,
      symbolTileSize: 60,
      rendererStyle: "single_arrow"
    });
```



WCS Layer

- Enable Web Coverage Service on image services
- Get server info: coverages, formats, interpolations, versions, etc.
- Support for identify
- Client side pixel filters too!

```
var wcsUrl = "//sampleserver6.arcgisonline.com/arcgis/services/ScientificData/MODIS_Landcover/ImageServer/WCSServer";
var wcsLayer = new WCSLayer(wcsUrl, {
  version: "1.0.0",
  pixelFilter: colorizer }
);
```

Documentation Resources

- **ArcGIS API for JavaScript:**
  https://developers.arcgis.com/javascript/3/

- **ArcGIS API for JavaScript 4.x**
  https://developers.arcgis.com/javascript

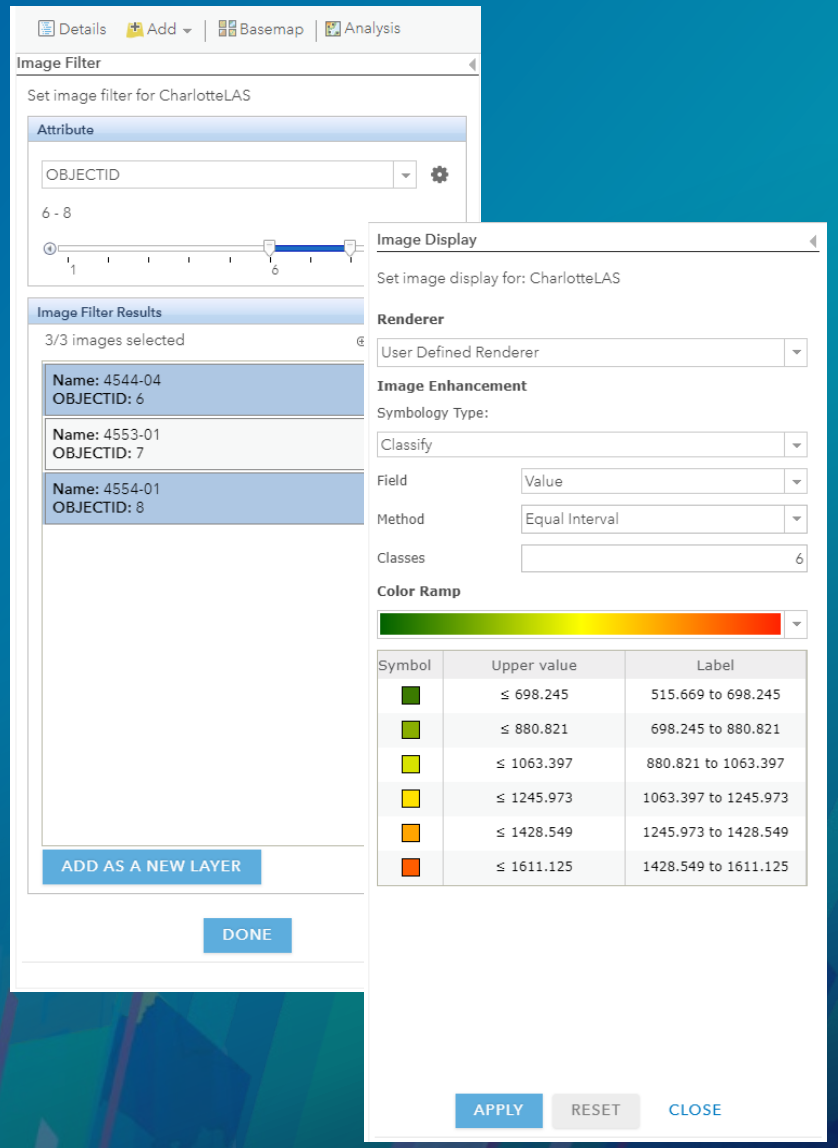
What are we talking about today…

- Image Services
- ArcGIS API for JavaScript
- **Map viewer in ArcGIS Enterprise**
- **Web AppBuilder**
- **Configurable App Templates**

ArcGIS Enterprise Map Viewer

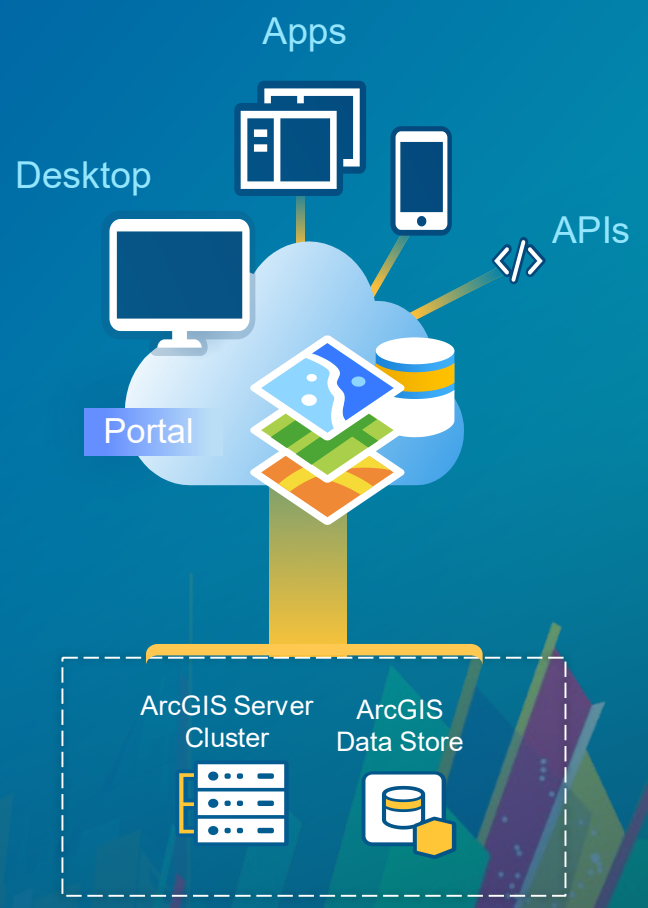
Map Viewer Capabilities

- **Image Filter**
  - Select imagery based on an attribute in a specific geographic area of interest
  - Create a new layer with the results
- **Symbology**
  - Stretch, Classify and Unique Values renderers
  - Leverages the rendering capability in ArcGIS JavaScript API
  - Access from the Image Display pane
- **Image Processing**
  - Dynamic processing (on the fly): Server processing templates
  - Persisted output: Raster Analysis (ArcGIS Enterprise)



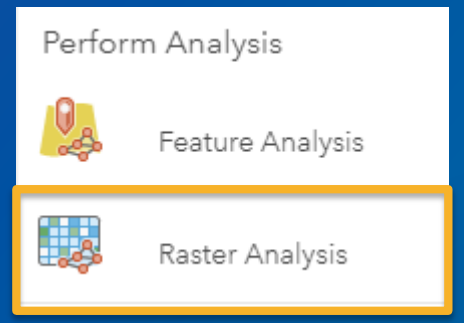
Raster Analysis

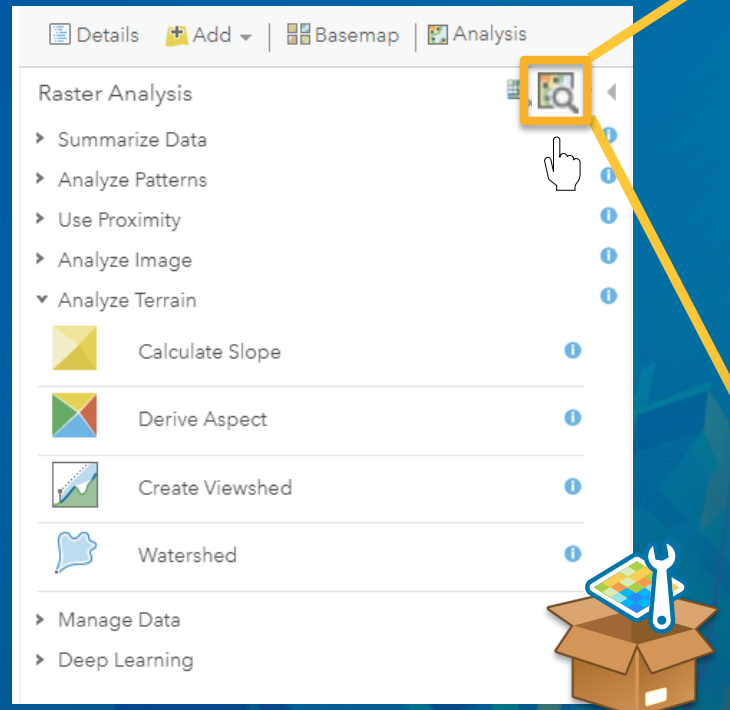
- Quickly extract information using spatial analysis and raster models by leveraging distributed processing and storage on ArcGIS Server
- Optimized raster format: CRF (Cloud Raster Format)
  - Multi-band, block based, multiple readers and writers
- Can be accessed by different clients (ArcGIS Pro and Map viewer) and have a REST API to access services
- Once completed, results immediately available in your Web GIS

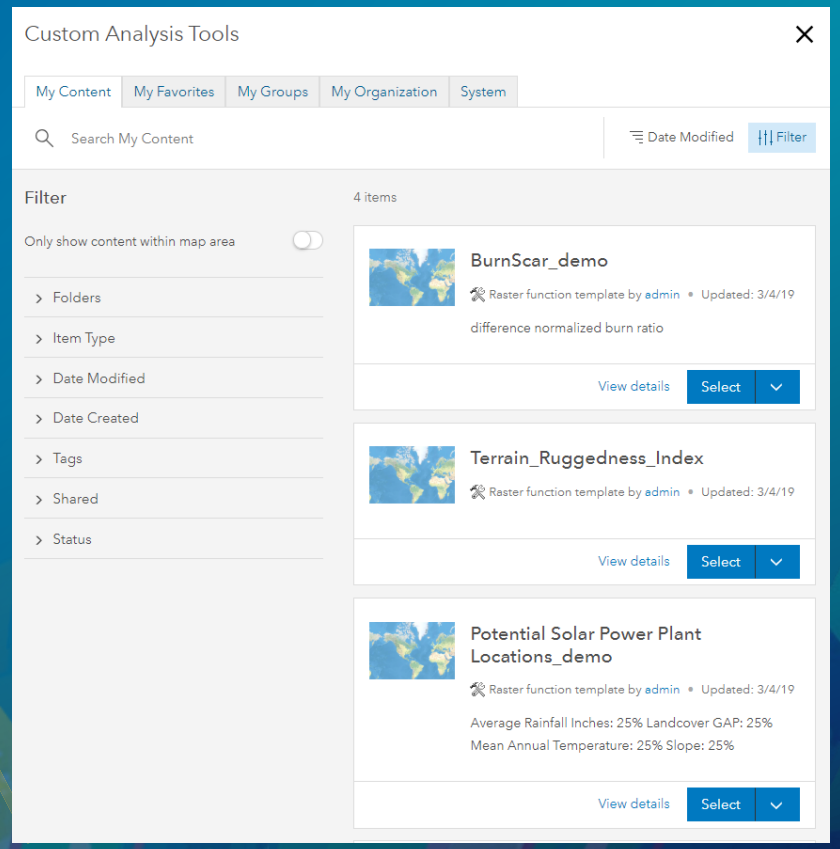


Raster Analysis using ArcGIS Enterprise Map Viewer

- 17 out of the box tools available
- Ability to **share** and **apply** Raster Function Templates
- **Build** and **edit** custom templates

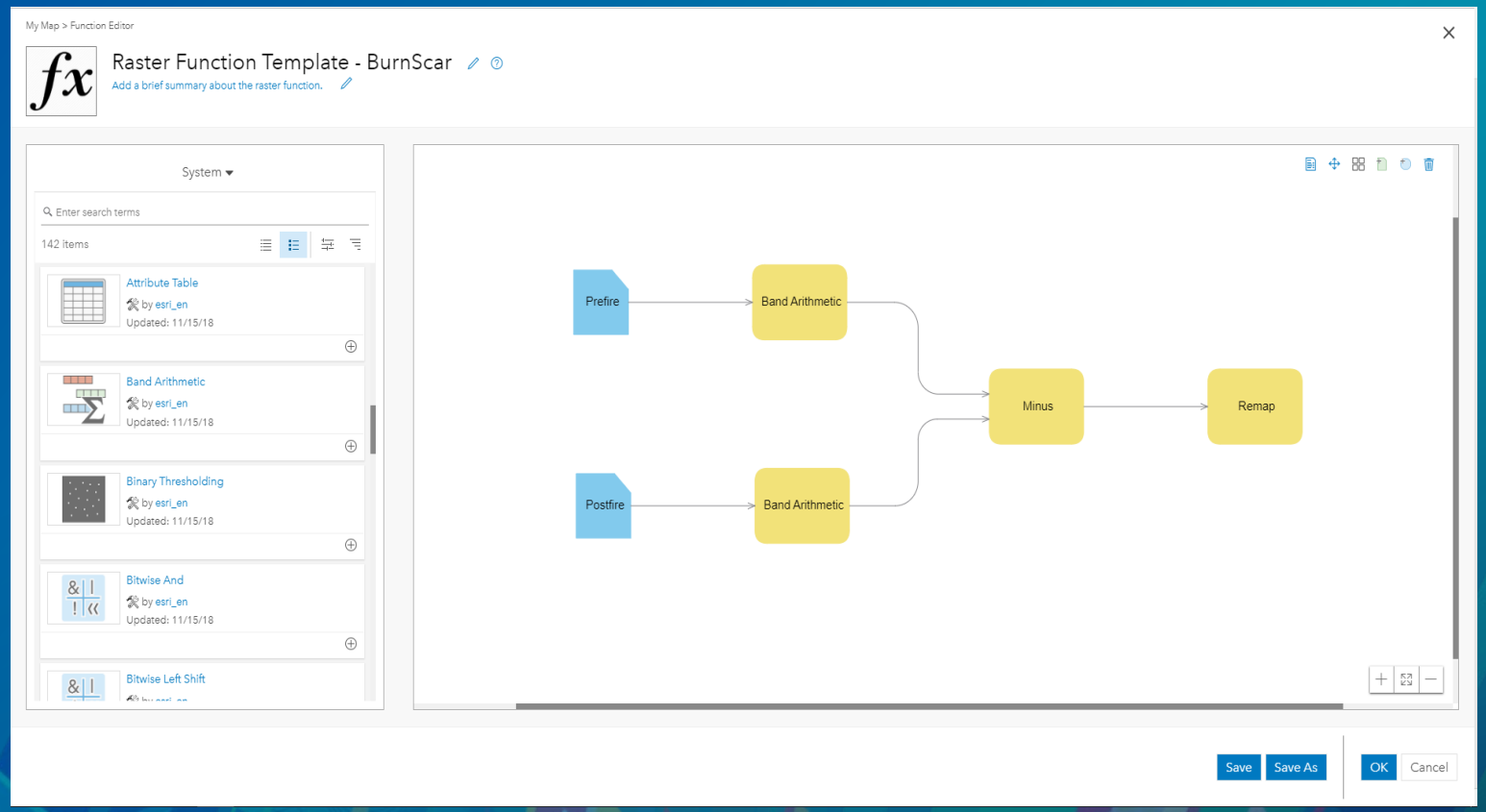


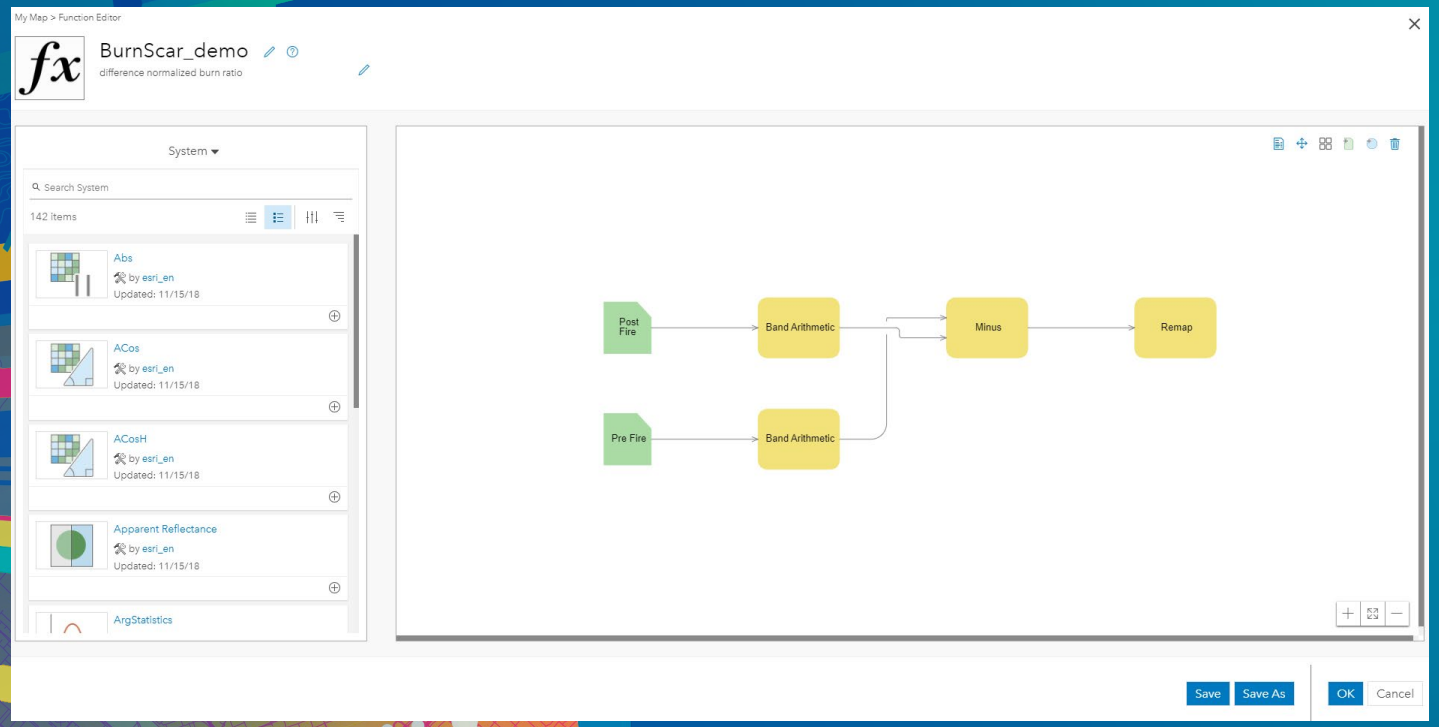




Raster Function Editor

- **View**, **edit** and **build** Raster Function Templates in Portal
- **Save** and **share** new or updated templates
- **Run** as an analysis operation to produce a persisted output
- Available in 10.7 release





Raster Function Editor Demo

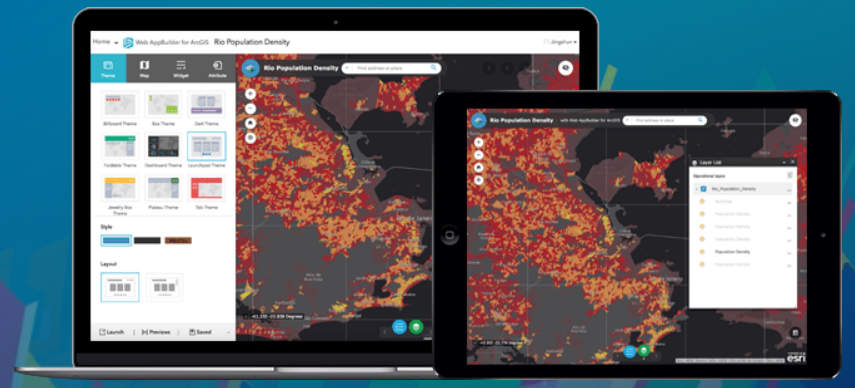
BurnScar

Image Services in

Web AppBuilder for ArcGIS

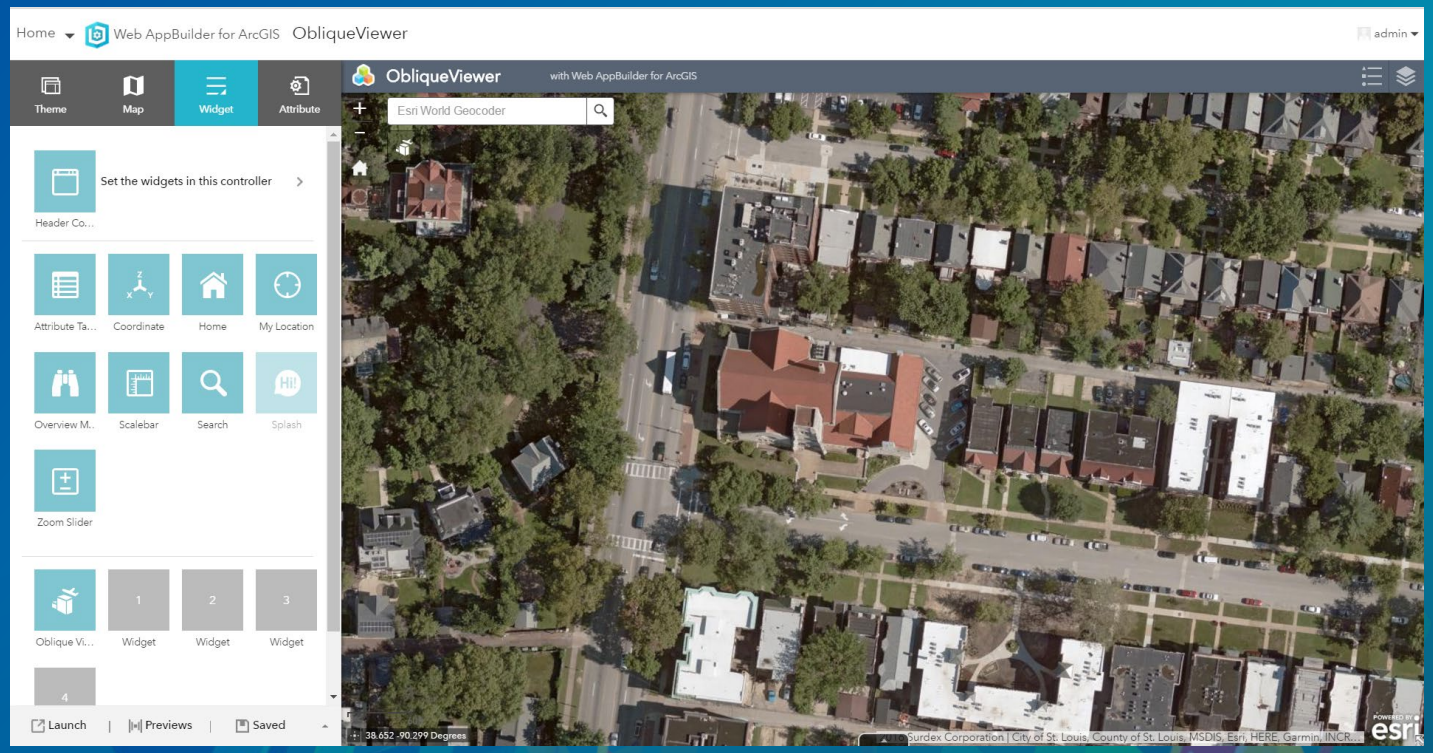
Web AppBuilder for ArcGIS

- Create apps with ease (no coding required)
- Extensible framework with the developer edition
- Fully integrated with the ArcGIS platform
- Apps run on any device

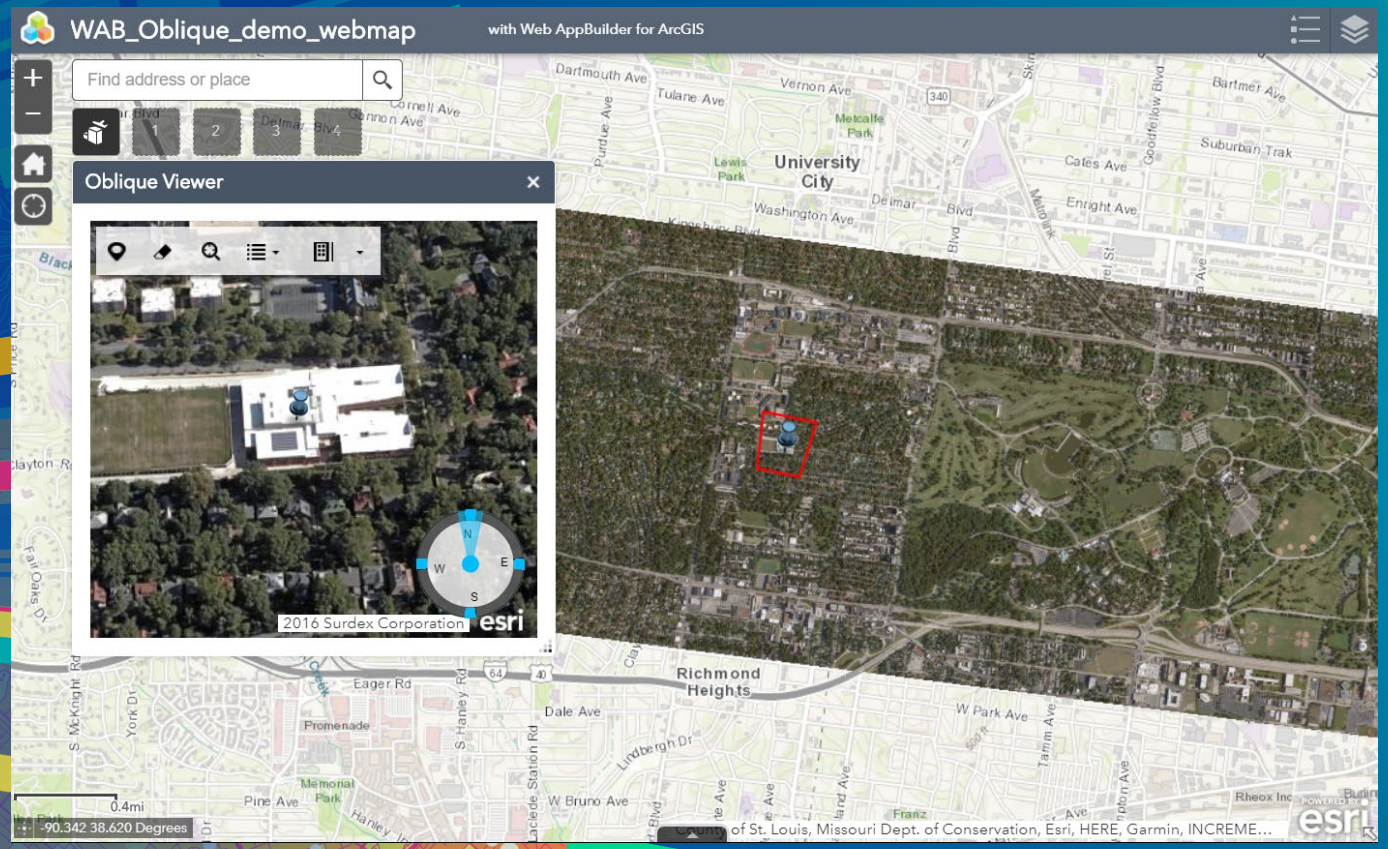


Imagery support in Web AppBuilder for ArcGIS

- Layer List
- Legend
- Query
- Attribute Table
- Popups
- Image Measurement*
- Oblique viewer*



** Specifically designed to work with image services*



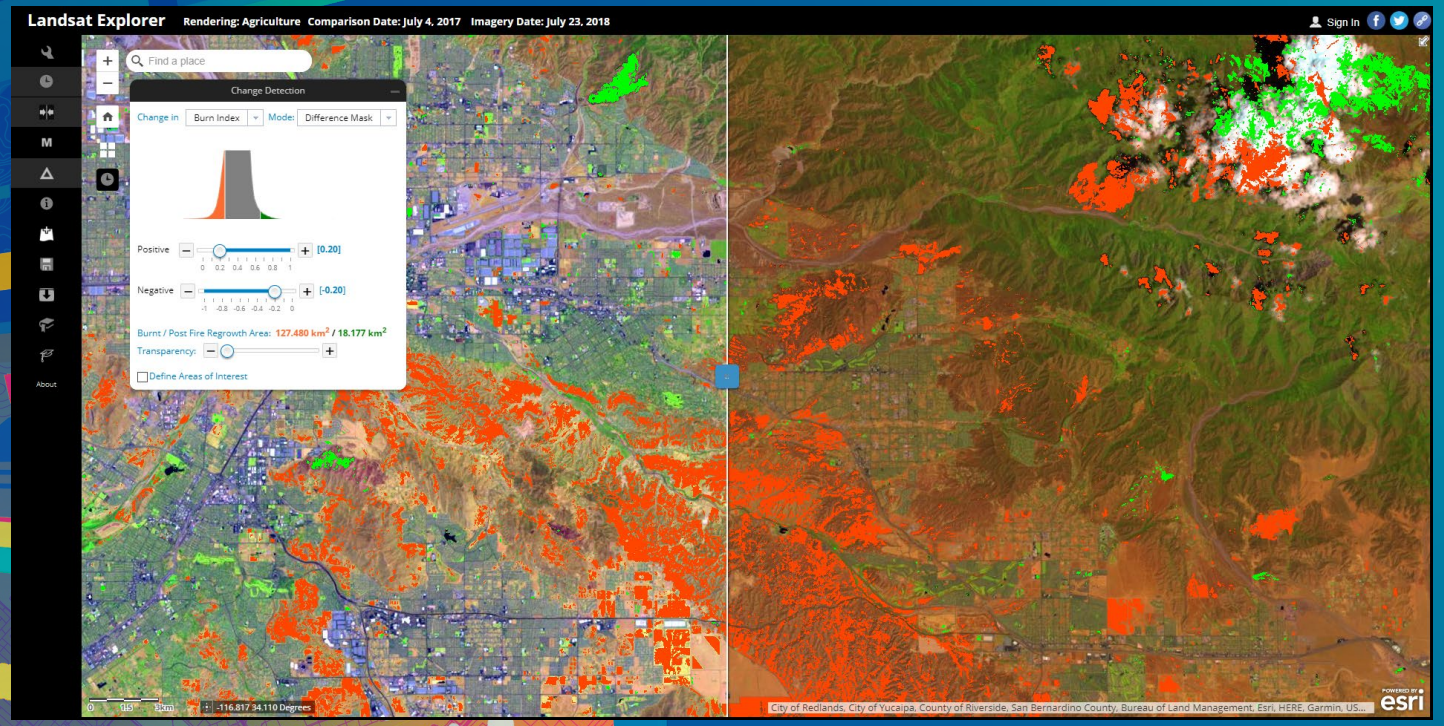
Web AppBuilder Demo

Oblique Viewer widget

Web AppBuilder widgets for Image Services (WABIS)

- Custom widgets to be used in Web AppBuilder
- Designed to work with image services
- Requires ArcGIS Online account and WAB Developer Edition
- IS Image Selector, Layers, Change detection, Renderer, Mask, Export, etc.
- Process imagery, export and share your results

https://github.com/Esri/WAB-Image-Services-Widgets



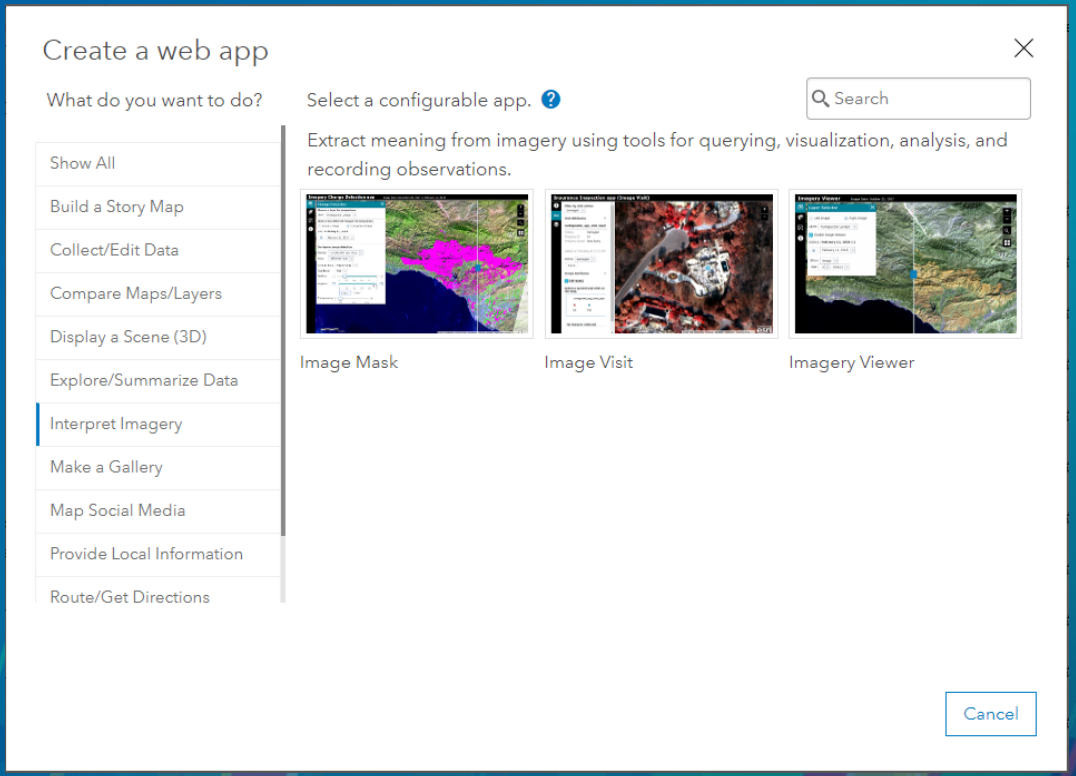
Custom Imagery Widgets Demo

Landsat Explorer

http://landsatexplorer.esri.com/

Configurable App Templates

- Image Mask
  - Calculate and visualize change between two images in an imagery layer
- Image Visit
  - Review the attributes of a predetermined sequence of locations in imagery
- Imagery Viewer
  - Visualize and interpret imagery layers through time and space



What should I use…

ArcGIS API for JavaScript	• Latest features, the most flexibility • Create, customize, and optimize full-featured 2D/3D web apps • Your app needs to be hosted on your own server, or third-party resources like Amazon AWS or Microsoft Azure
Web AppBuilder	• No coding required • Ready-to-use widgets and configurable themes • Host your apps online or run them on your own server • Create custom app templates
Configurable App Templates	• No coding required • The fastest and easiest way to go from a map to a shareable app • Many configurable apps to choose from • Easily share your app with others. Your app can be hosted on ArcGIS Online, it can reside on your organization's servers, or you can use third-party resources

Road ahead

- In ArcGIS API for JavaScript (4)
  - Support for visualizing vector data
  - Non-blocking decoding of pixel blocks
  - Time support
- Map viewer
  - Support for creating custom processing chains for on the fly processing

To conclude…

- Leveraging image services using the ArcGIS API for JavaScript
  - Mosaic rule, server side processing, client side processing and rendering
- Imagery capabilities in Map Viewer
  - Raster analysis
  - Symbology, filtering capabilities
  - Author your layers and use across the platform!
- Imagery in Web AppBuilder
  - Out of the box widgets
  - WABIS
- Configurable App Templates

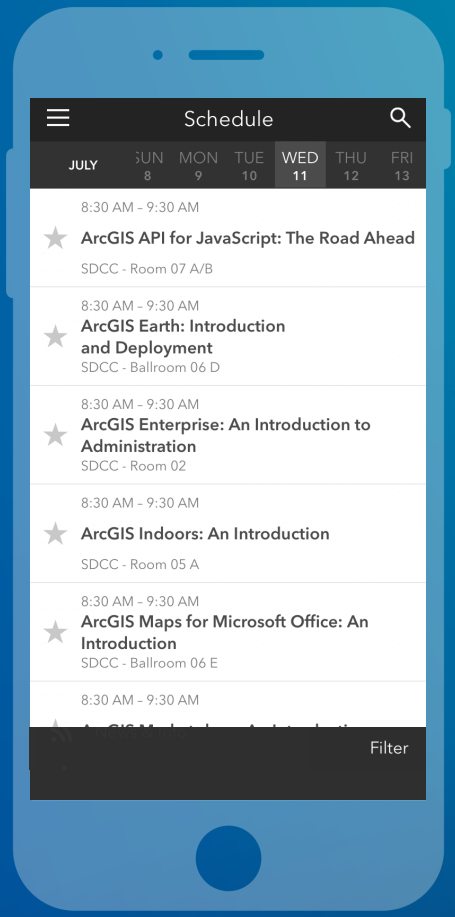
esri

THE SCIENCE OF WHERE

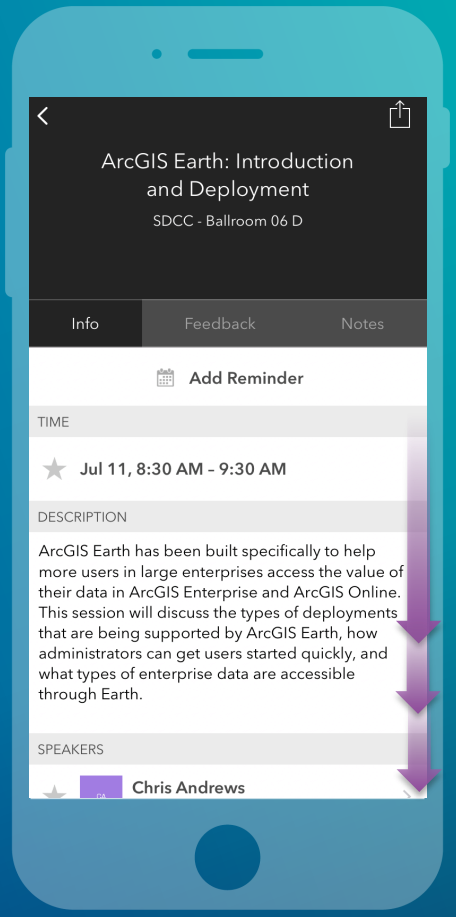
Please Take Our Survey on the App

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”