ArcGIS API for JavaScript: Using Arcade with your Apps

Kristian Ekenes & David Bayer

2018 **Esri DEV**Summit **Conference** | Palm Springs, CA

Session Goals

Overview of Arcade
- What is Arcade
- Why use Arcade

Arcade Language
- Variables, Functions, Loops, Conditional Statements

Arcade and the ESRI JavaScript API
- Using Arcade in Custom Apps

What is Arcade?

- **A new expression language for the ArcGIS Platform**
  - ArcGIS Pro
  - Runtime SDKs
  - JavaScript API
  - Web apps
- **Primarily designed for ArcGIS web maps and web scenes**

Arcade Purpose

Arcade is NOT intended to be a Full Programming / Scripting Language

- Goal : Lightweight and Simple
- Equivalent to a Spreadsheet Cell Calculation

Embedded Expressions

- Labeling, Rendering, Symbol Variation, ….
- Sharing

Not a replacement for Python for geoprocessing and Automation

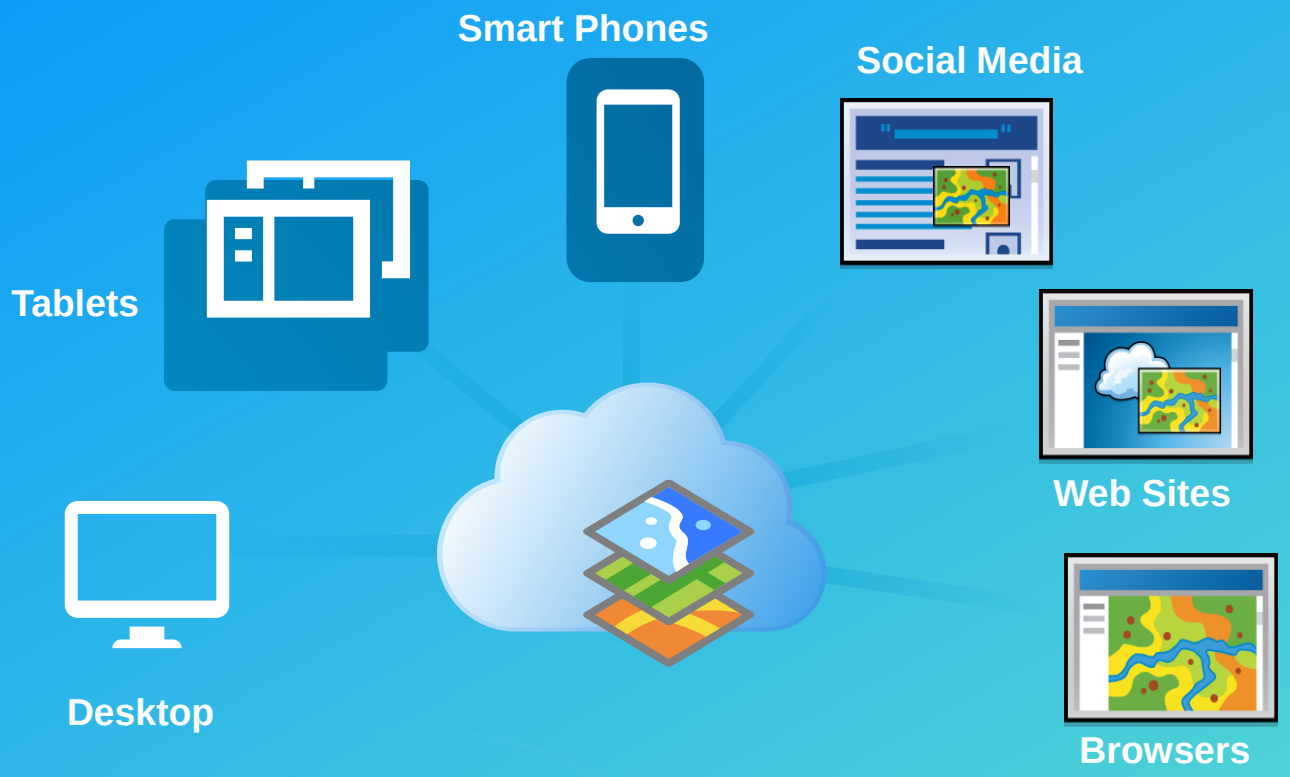
Hello World

Why is it called Arcade?

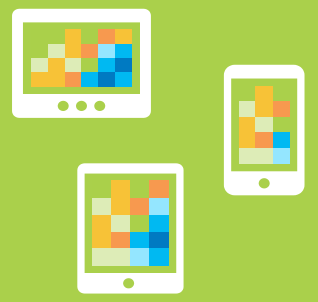
Why do I need another language?

ArcGIS: Make, Share, Anywhere

ArcGIS Environments and Platforms



ArcGIS: Make, Share, Anywhere - Challenges



Mobile and Native Apps

- Size of Executable
- Security
  - Native Access

Browser based Apps

- Page Size
- Security
  - XSS
- No Binaries

Traditional Development / Scripting Languages

Arcade Goals

Portable

- Write an Expression in Pro, have it run in a Browser, or on a Device

Secure

- Ensure that Expressions or Scripting do not compromise Security

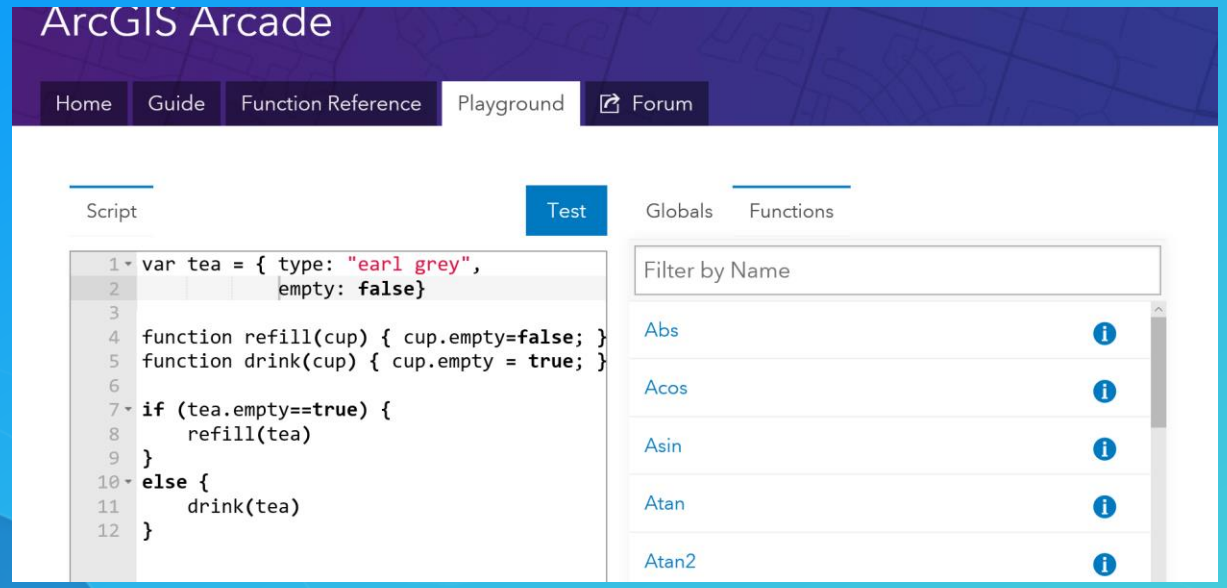
Lightweight

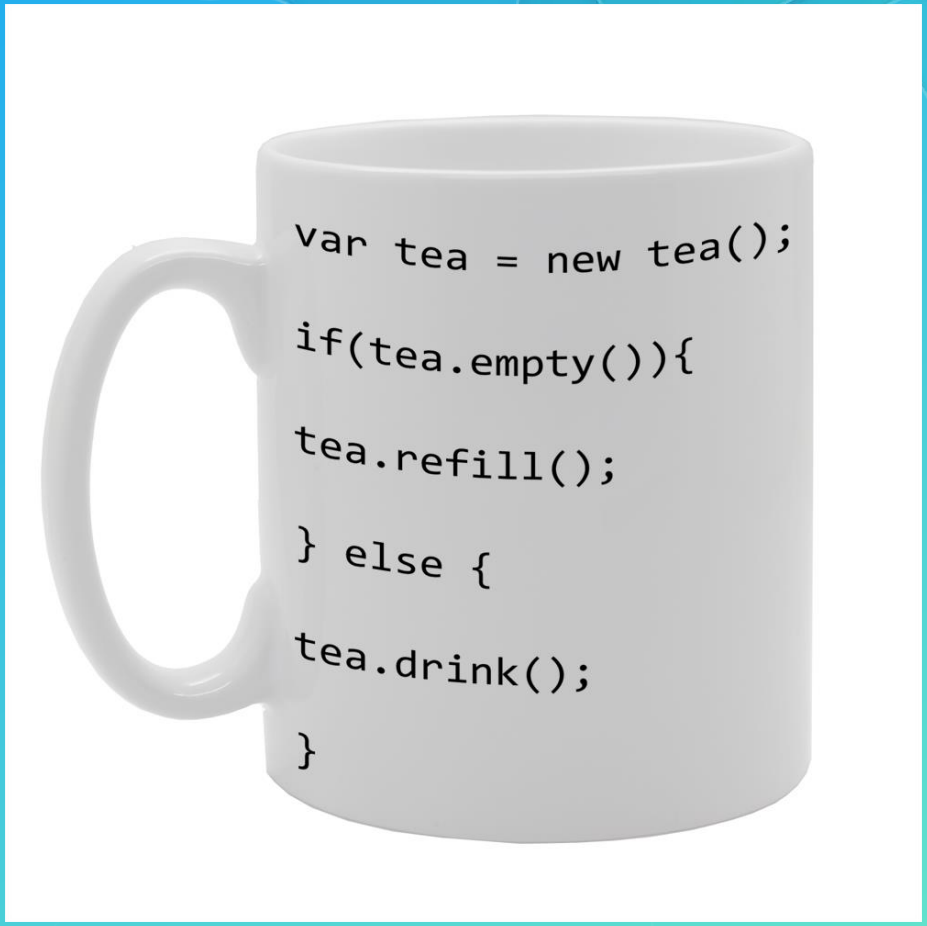
- Most requirements for Scripts and Expressions are Lightweight. The language should be small, and fast

Geospatial

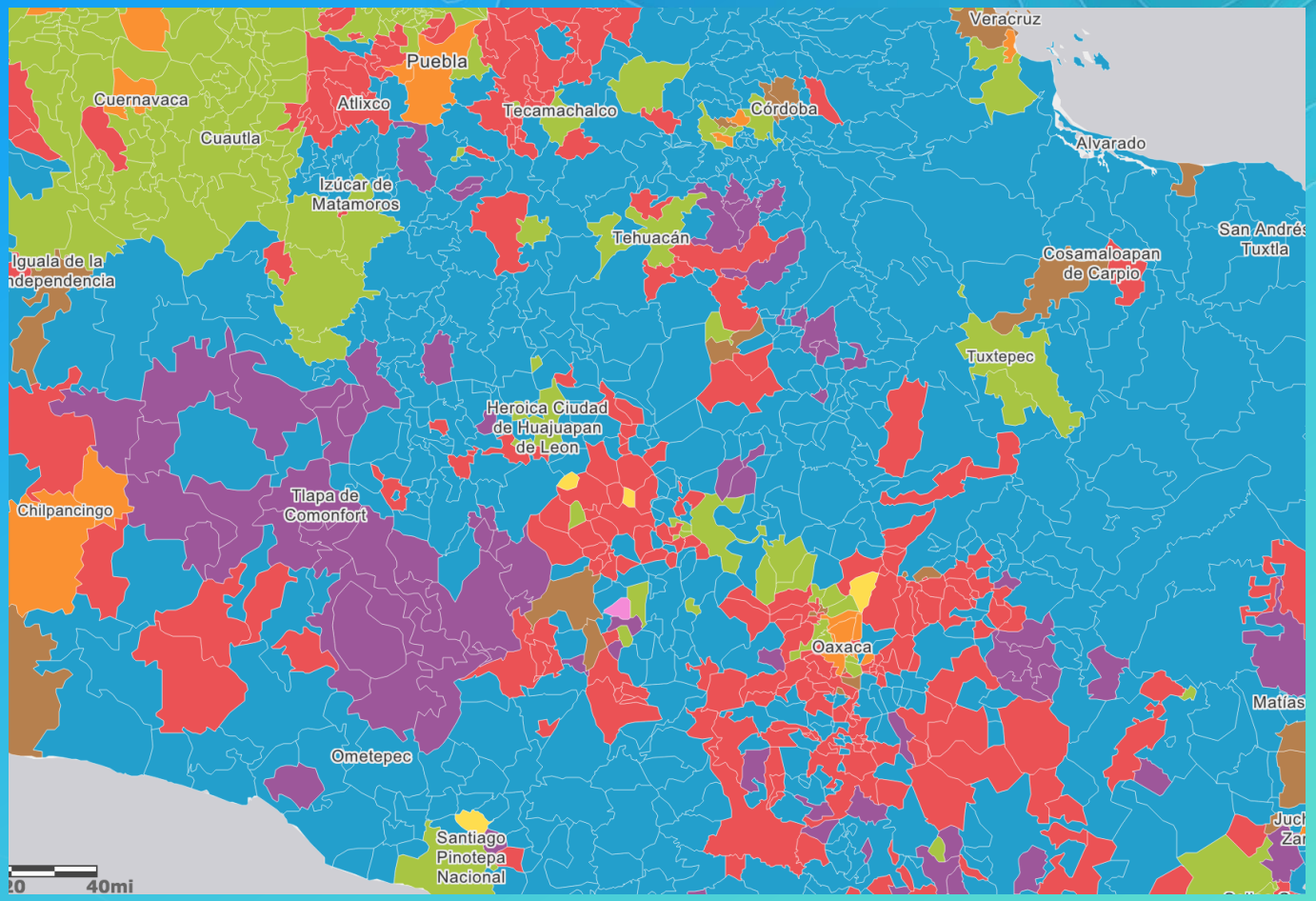
- Scripts should treat Geospatial as First Class Members

What can I do with it?





Arcade Examples



Language Overview

Arcade : Language Features

Designed for Simple Expressions

- Single Line - Implicit Returns
- Case Insensitive Language
- Dictionary Syntax for Field Access

Has Language Features to Support more Complex Scripts

- Type System
- Implicit and Explicit Type Casting
- Flow Control Statements: IF, FOR
- Variable Declaration and Assignments
- User Defined Functions

Arcade Scripts run and return a value. They are self contained and cannot alter their environment.

Implicit Returns

```
(10 + 100) * 30

Equivalent to:

return (10 + 100) * 30;
```

Implicit Returns

- Last executable statement that is a value, will automatically be considered the return value.
- Enables very simple Single Line Expressions

Case Insensitive

```
UpPeR(“Hello”)
$FeAtUrE.FiElD1
```

Language is Case Insensitive

- Quicker to Author
- Simpler for working with Data, and Field Names

Functions

Data Functions

- Dictionary
- DomainCode
- DomainName
- Feature
- First
- HasKey
- IndexOf
- Reverse
- Sort
- TypeOf

Date Functions

- Date
- DateAdd
- DateDiff
- Millisecond
- Minute
- Hour
- Month
- Weekday
- Year
- Day
- Now
- Today

Logical Functions

- IsEmpty
- DefaultValue
- When
- Decode
- Iif
- Boolean

Mathematical Functions

- Abs
- Acos
- Asin
- Atan
- Atan2
- Average
- Ceil
- Cos
- Count
- Exp
- Floor
- Log
- Min
- Max
- Number
- Pow
- Random
- Round
- Sin
- Sqrt
- Stdev
- Sum
- Tan
- Variance

Text Functions

- Concatenate
- Console
- Find
- Lower
- Left
- Mid
- Proper
- Replace
- Right
- Split
- Text
- Trim
- Upper

> Functions are the Power of Arcade

Functions : Geometry

Geometry Functions

- Geometry
- Multipoint
- Extent
- Polyline
- Polygon
- Area
- AreaGeodetic
- Buffer
- BufferGeodetic
- Centroid
- Clip
- Contains
- Crosses
- Cut
- Difference
- Disjoint
- Distance
- Equals
- Intersection
- Intersects
- Length
- LengthGeodetic
- MultiPartToSinglePart
- Overlaps
- Relate
- SymmetricDifference
- Touches
- Union
- Within

Global Variables and Profiles

Rendering

- Use expressions to provide values required for different types of Renderers
- Global Variables
  - $feature
  - $view

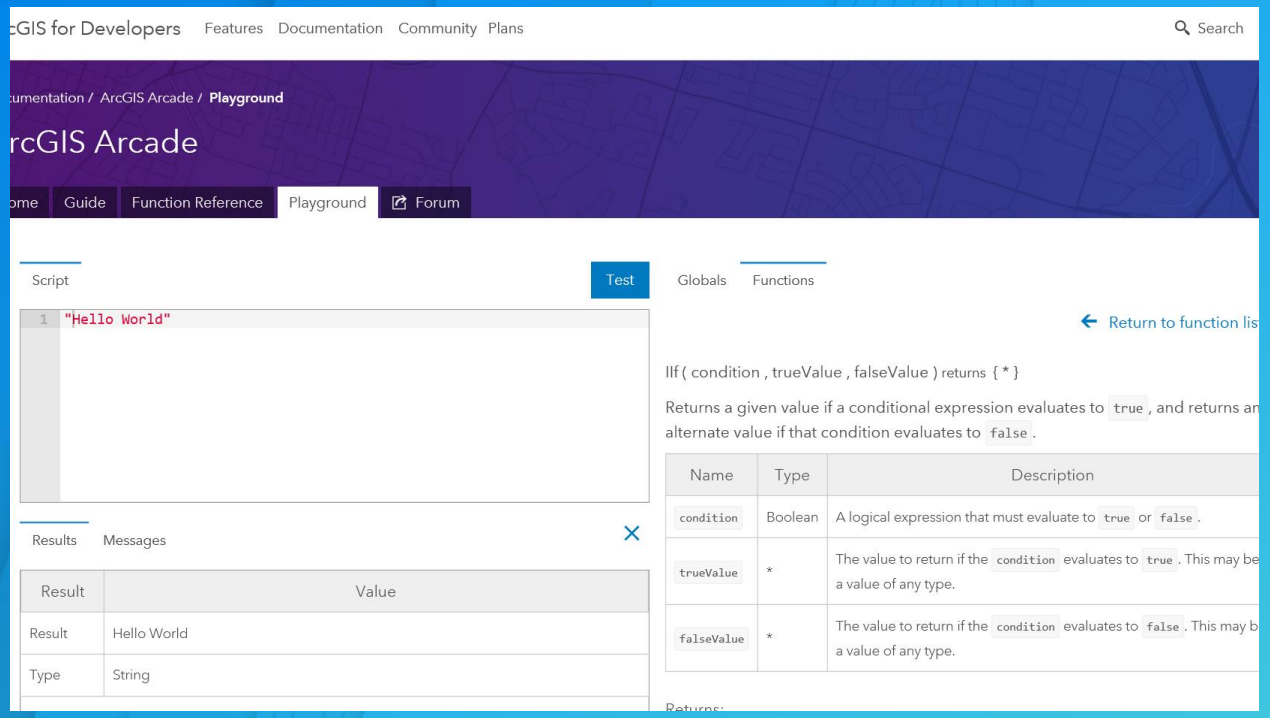
Labeling

- Use expressions to provide values required for different types of label classes
- Global Variables
  - $feature
  - $view

Popup

- Use expressions to provide values for popup tables and other content elements
- Global variables
  - $feature

> Profiles provide the purpose for executing an Arcade Expression



Exploring Profiles and Functions in the Playground

Type System

- Simple Types
  - Numbers
  - Booleans
  - Dates
  - Strings
- Object Types
  - Dictionary
  - Feature
  - Array
  - Point
  - Line
  - Polygon
  - Multipoint

```
var myNumber = 10;
var myText = “Hello”;
var myDate = Date(2015,1,1);
var myBool = true;
var myDictionary = { “key1”: 10 };
var myFeature = { “geometry”:  {…},
                  “attributes” {“key1”: 10 }};
var myArray = [1,2,3];
var myPoint = Point({….});
```

Dates, Dictionary, Feature, Point, Line… all have overloaded constructors.

Implicit and Explicit Type Casting

Implicit Casting

- For Function Parameters
- For Expressions

Explicit Casting

- Functions cast between types
- Number, Text, Date, Boolean

```
Implicit Casting
return 10 + “Hello”

Explicit Casting:
return text(10) + “Hello”
```

If Statement

- Traditional IF Statements are supported
- For simple single line expressions, consider using Functions
  - IIF
  - Decode
  - When

```
if (myvalue==true) {
 return 1;
}
else {
  return 2;
}
```

Equivalent to:

```
IIF(myvalue==true, 1, 2)
```

For Statement

For Statements

- Same syntax as JavaScript

For In Statements

- Iterates over indices of an array, or field names of a Dictionary or Feature

Break, Continue and Return Statements Supported inside Block

```
for (var z=1; z<100; z++ ) {
 k+=z;
}

for(var k in myArray) {
    n+=myArray[k];
}
```

User Defined Functions

```
function MultiplyNumber(thenumber) {
 return thenumber*10;
}

MultiplyNumber(10);
```

Function Keyword used to Declare Functions

- Must be Declared ahead of Use
- Variables defined inside of Function, have Local Scope
- Functions cannot be declared inside of Functions.

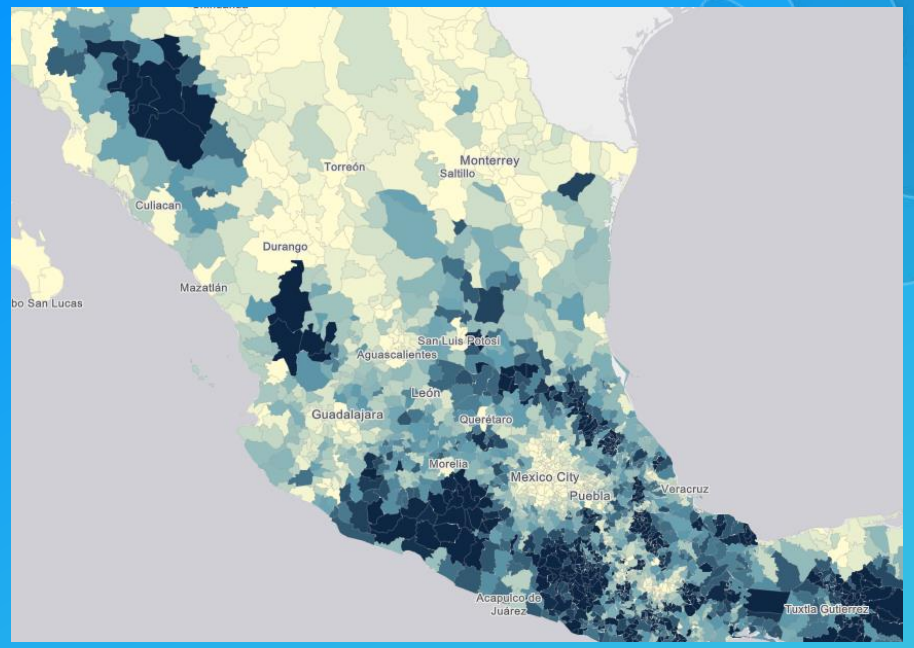
Variable Assignment

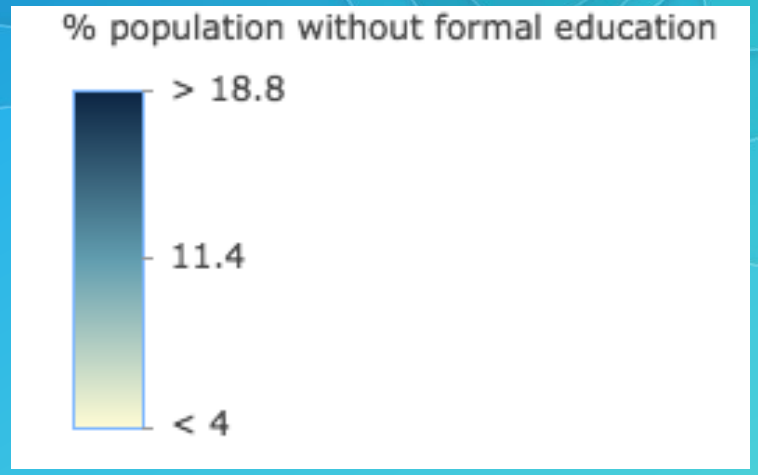
```
++myNumber
i=i+1;
myArray[10]=11;
myFeature.Field1 = “ddd”;
```

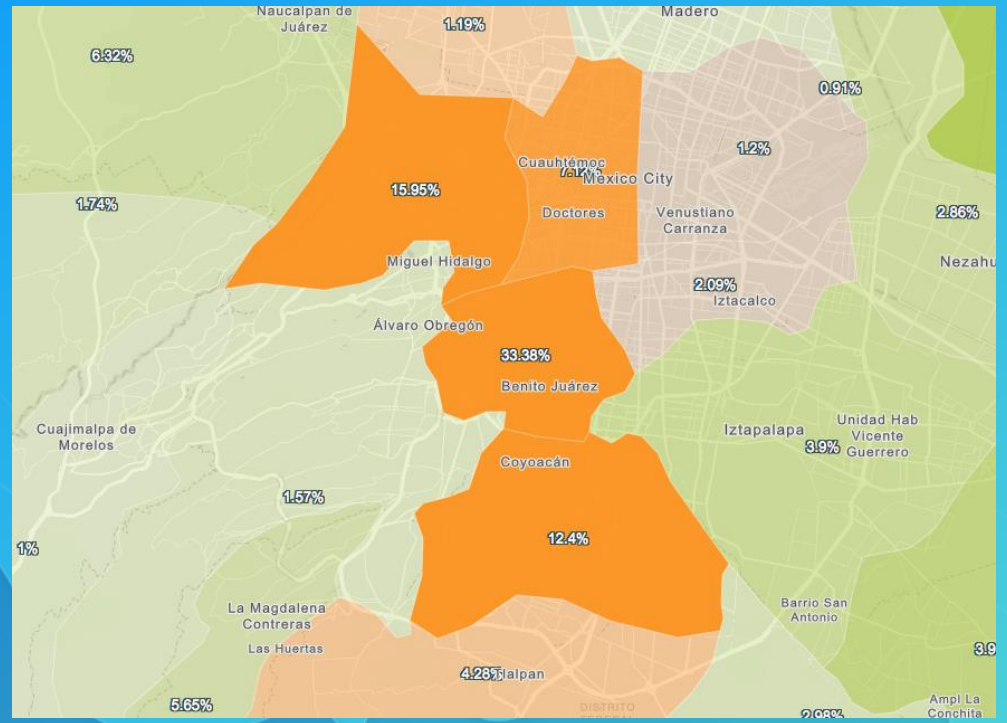
Variables can have their values reassigned.

- Objects Types maybe Immutable, if passed into Arcade.
- Arrays are not Sparse. They must be sequential.
- Geometry Types are immutable inside. You cannot change the points in a path of a line.

Demos







% population without formal education Edit

Script | Test | Globals | Functions

```
var noSchool = $feature.EDUC01_CY;
var total = $feature.EDUCA_BASE;
Round((noSchool/total)*100,2);
```

$feature.AVGHHSZ_CY
(2014 Average Household Size (MBR))

$feature.EDUC01_CY
(2014 Pop 12+/Educ: Without Education (MBR))

$feature.EDUC02_CY
(2014 Pop 12+/Educ: Preschool or Kindergarten (MBR))

$feature.EDUC03_CY
(2014 Pop 12+/Educ: Incomplete Primary (MBR))

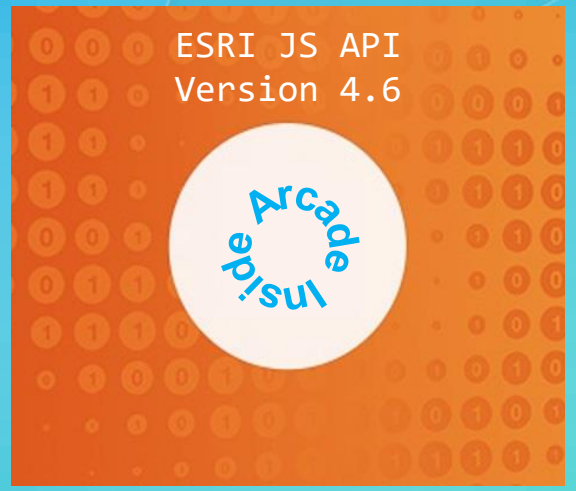
$feature.EDUC04_CY
(2014 Pop 12+/Educ: Primary Completed (MBR))

$feature.EDUC05_CY
(2014 Pop 12+/Educ: Incomplete Secondary (MBR))

$feature.EDUC06_CY

OK Cancel

Arcade and the ESRI JS API



“ Arcade Interpreter available in both 3.23 and 4.6 JavaScript API

ESRI JS API and Arcade Profiles

Arcade Expressions set as Properties

Classes that support Arcade Expressions

- `ClassBreaksRenderer`
- `UniqueValueRenderer`
  - visualVariables
    - `Color`
    - `Size`
    - `Opacity`
    - `Rotation`
- `LabelClass`
- `PopupTemplate`

Supported for FeatureLayer and SceneLayer

ESRI JS API : Arcade Expressions

```
var renderer = {
 type: “class-breaks”,
 valueExpression: "( $feature.TOT_VOTES / $feature.REG_VOTERS ) * 100",
 valueExpressionTitle: "Title to Display"
 classBreakInfos: [ ... ],
};
```

Expression Properties are set as Literal Strings

Arcade Interpreter will Execute valueExpression for every Feature

ESRI JS API : Arcade Expressions in JS API 3.23

Arcade Expressions can be used For Labeling

```
var labelClass = new LabelClass({
    labelExpressionInfo: {
      expression: "$feature.STATION_NAME"
    },
    labelPlacement: "below-right",
    minScale: 2500000
  });
  nameClass.symbol = new TextSymbol();
  // set the label class to the feature layer
  featureLayer.setLabelingInfo([ labelClass ]);
```

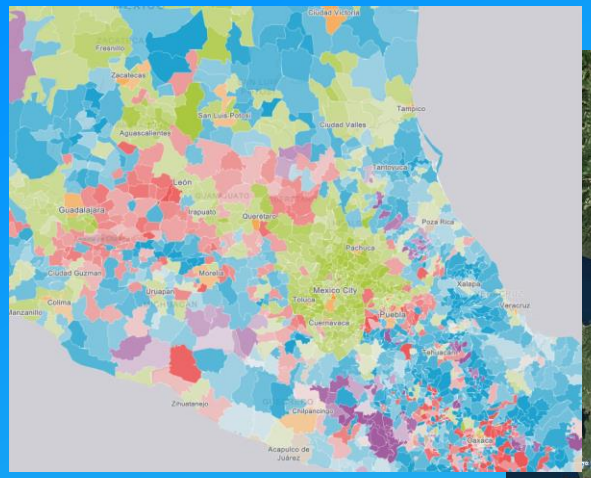
Practicalities: Scripts in Script

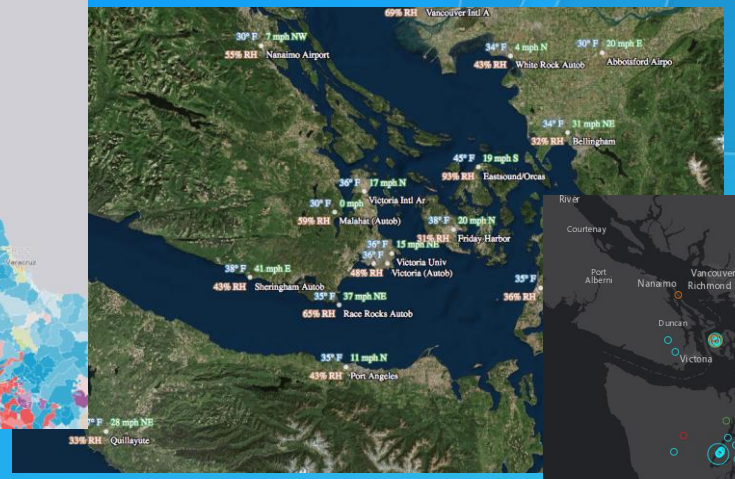
```
<script type="text/plain" id="myScript">
 Proper($feature.CityName)
</script>
...
var arcadescript = document.getElementById(“myScript").text;
```

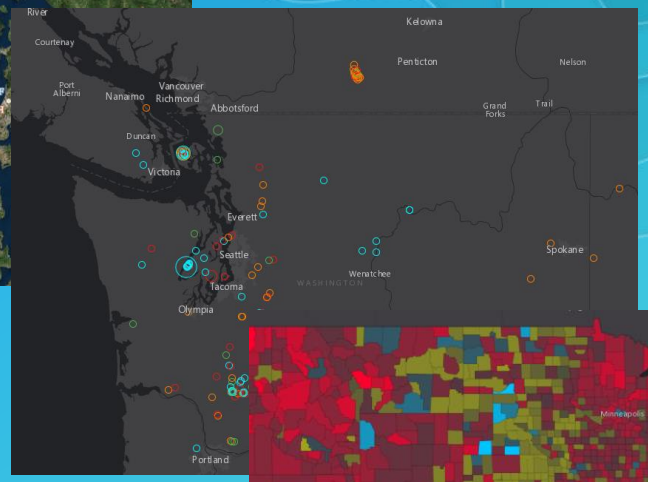
Scripts within Scripts

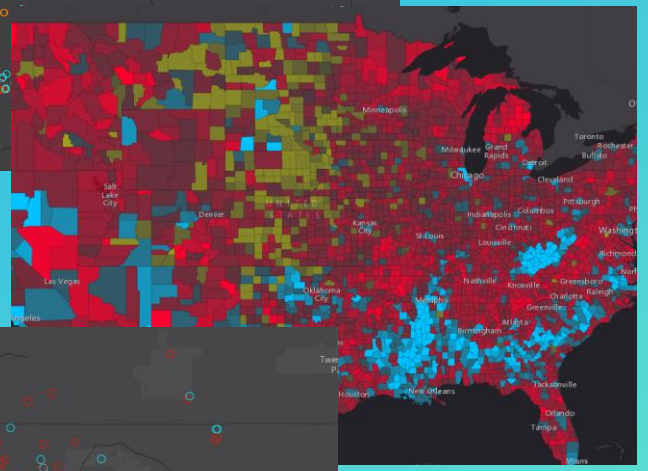
- Lots of String Escaping.
- Embed Script in HTML using MIME type text/plain

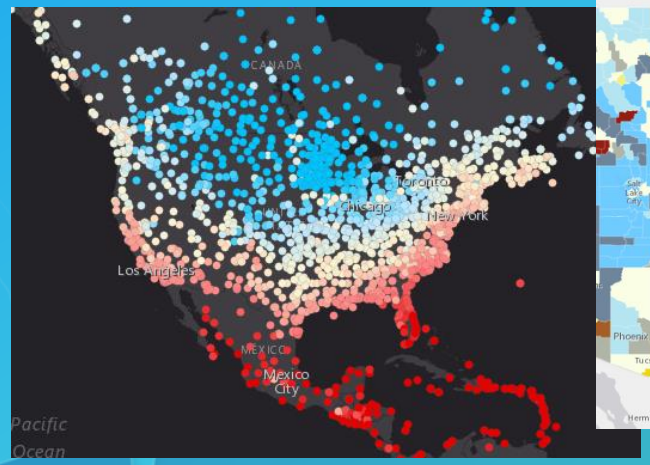
Demos

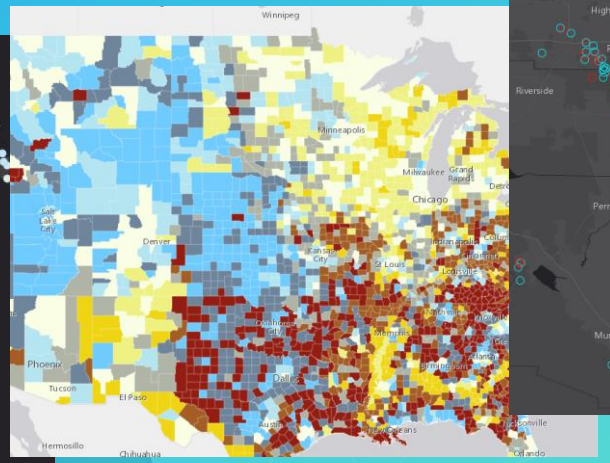


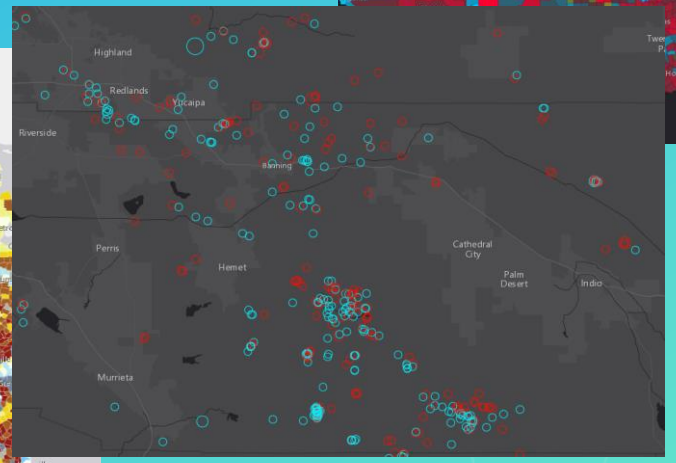

Nanaimo Airport
White Rock Autob
Abbotsford Airpo
Bellingham
Eastsound/Orcas
Victoria Intl Ar
Malahat (Autob)
Friday Harbor
Victoria Univ
Victoria (Autob)
Sheringham Autob
Race Rocks Autob
Port Angeles
Quillayute


Kelowna
Penticton
Nelson
Courtenay
Port Alberni
Nanaimo
Vancouver
Richmond
Abbotsford
Grand Forks
Trail
Duncan
Victoria
Everett
Seattle
Tacoma
Olympia
Spokane
Wenatchee
Portland


Minneapolis
Milwaukee
Chicago
Detroit
Cleveland
Salt Lake City
Denver
Kansas City
St. Louis
Indianapolis
Columbus
Pittsburgh
Cincinnati
Louisville
Nashville
Knoxville
Greensboro
Raleigh
Charlotte
Atlanta
Birmingham
Memphis
Las Vegas
Oklahoma City
New Orleans
Jacksonville
Orlando
Tampa
Miami
Toronto
Rochester
Buffalo


CANADA
Toronto
Chicago
New York
Los Angeles
MEXICO
Mexico City
Pacific Ocean


Winnipeg
Minneapolis
Milwaukee
Grand Rapids
Chicago
Salt Lake City
Denver
Kansas City
St. Louis
Indianapolis
Columbus
Cincinnati
Memphis
Dallas
Phoenix
Tucson
El Paso
Hermosillo
Chihuahua
Orlando

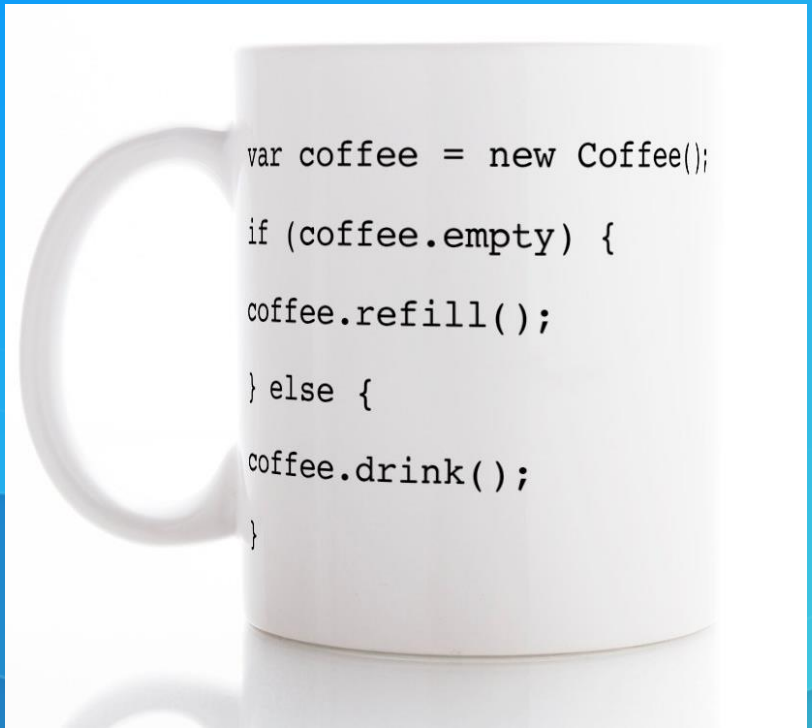

Highland
Redlands
Yucaipa
Riverside
Banning
Perris
Hemet
Cathedral City
Palm Desert
Indio
Murrieta

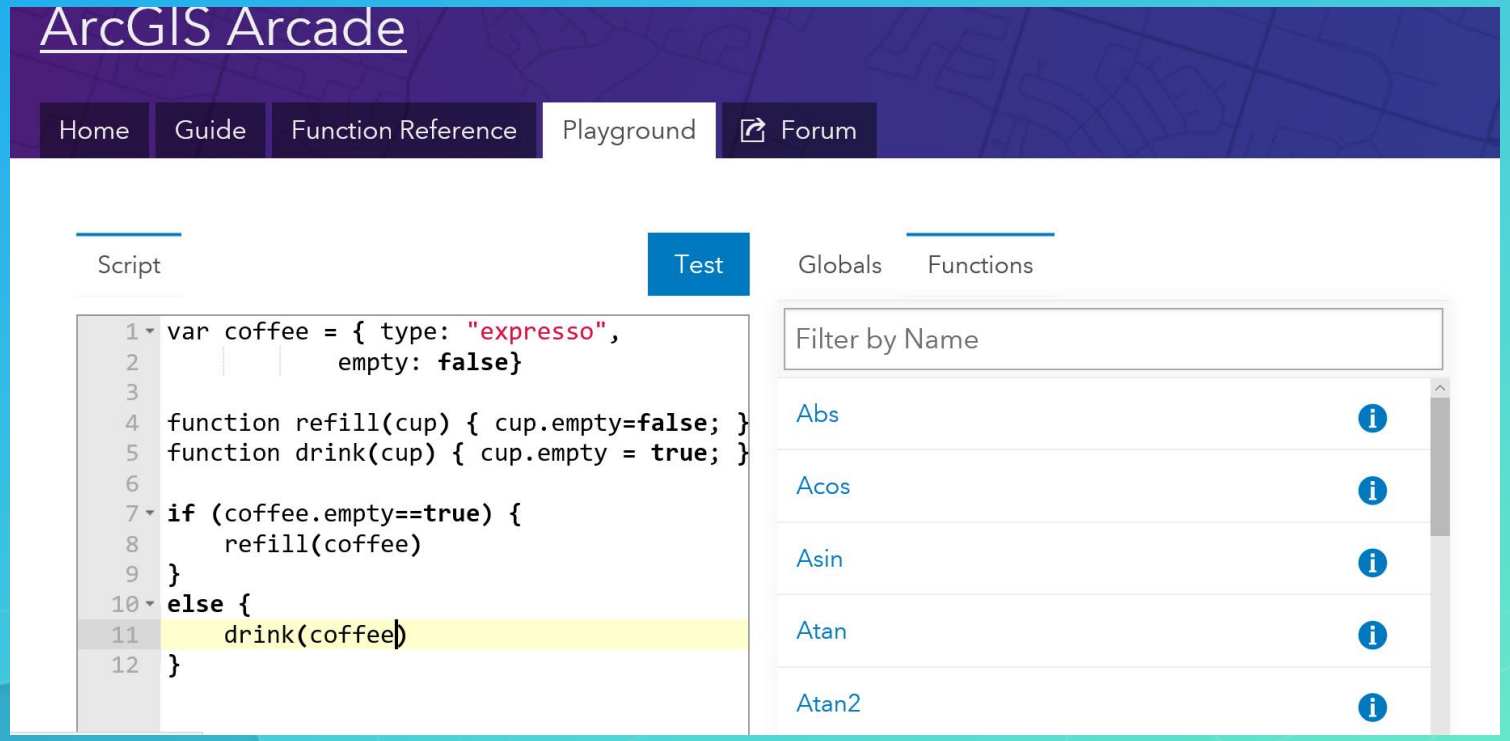
Expression sharing

Future

More uses of Arcade

- **Field Calculators**

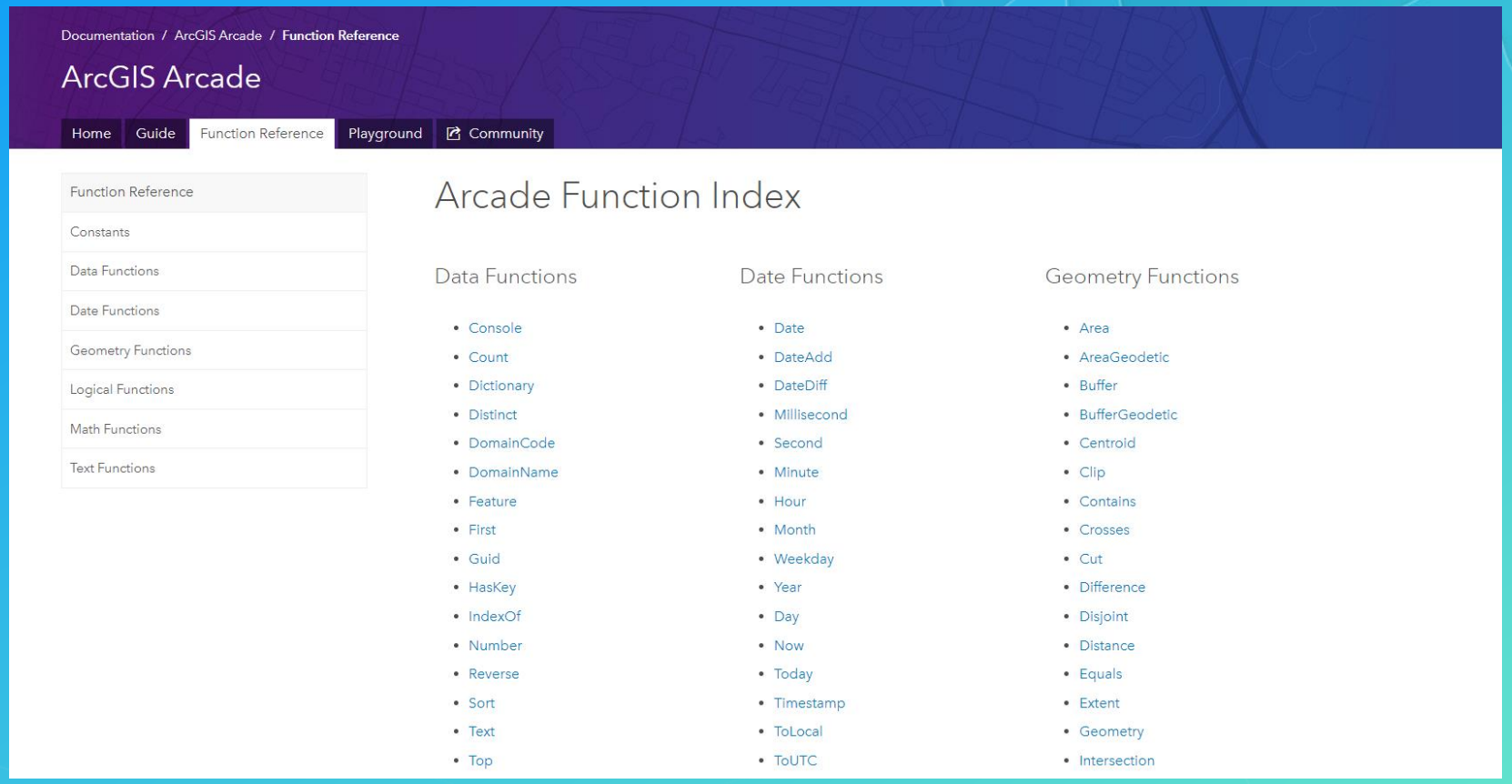




Adding Functions & Profiles

Identifying common operations and use cases

- Let us know what you need!



Finding Arcade Resources

Resources

For Arcade Documentation
https://developers.arcgis.com/arcade

Playground
https://developers.arcgis.com/arcade/playground/

Using Arcade with JS API

https://developers.arcgis.com/javascript/latest/guide/arcade/index.html

Blogs and Posts
https://blogs.esri.com/esri/arcgis/2016/12/19/introducing-arcade/

https://blogs.esri.com/esri/arcgis/2016/12/15/use-arcade-expressions-to-map-your-ideas/

http://odoe.net/blog/hello-arcade/

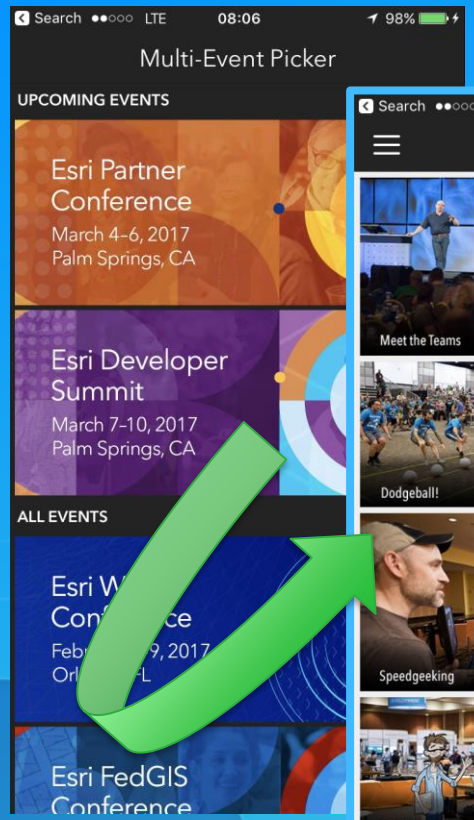
https://blogs.esri.com/esri/arcgis/2017/01/10/using-arcade-expressions-in-web-apps/

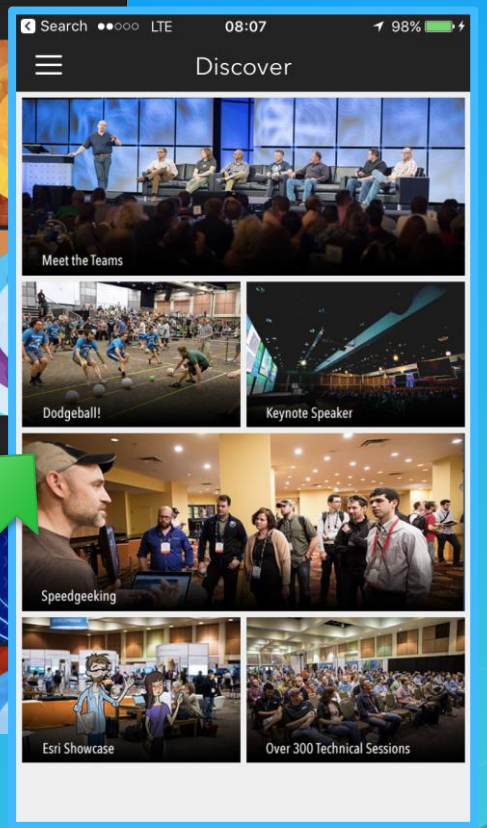
https://blogs.esri.com/esri/arcgis/2017/10/18/unwinding-the-clock-visualizing-time-with-arcade/

https://blogs.esri.com/esri/arcgis/2017/05/23/creating-a-predominance-visualization-with-arcade/

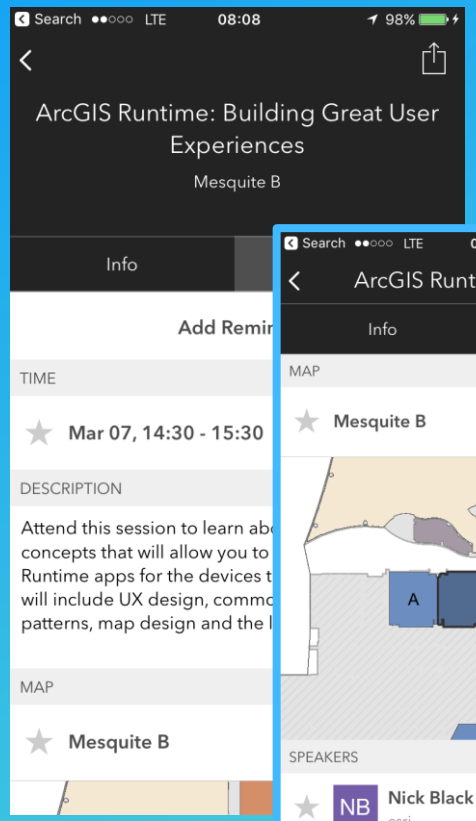
Please Take Our Survey!

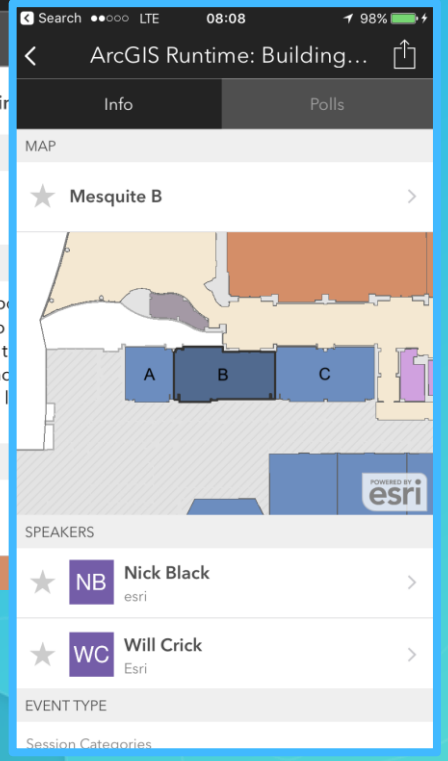
Download the Esri Events app and go to DevSummit





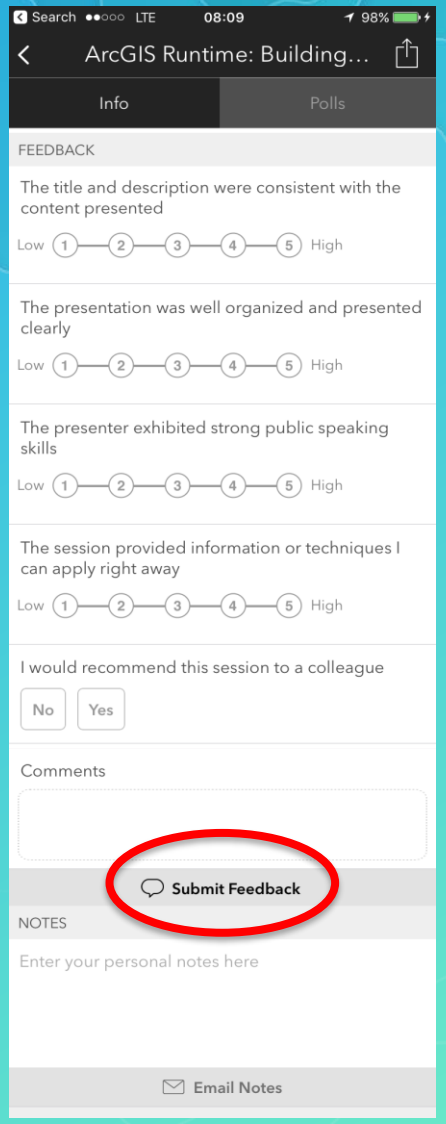
Select the session you attended





Scroll down to the "Feedback" section

Complete Answers, add a Comment, and Select "Submit"



esri

THE SCIENCE OF WHERE