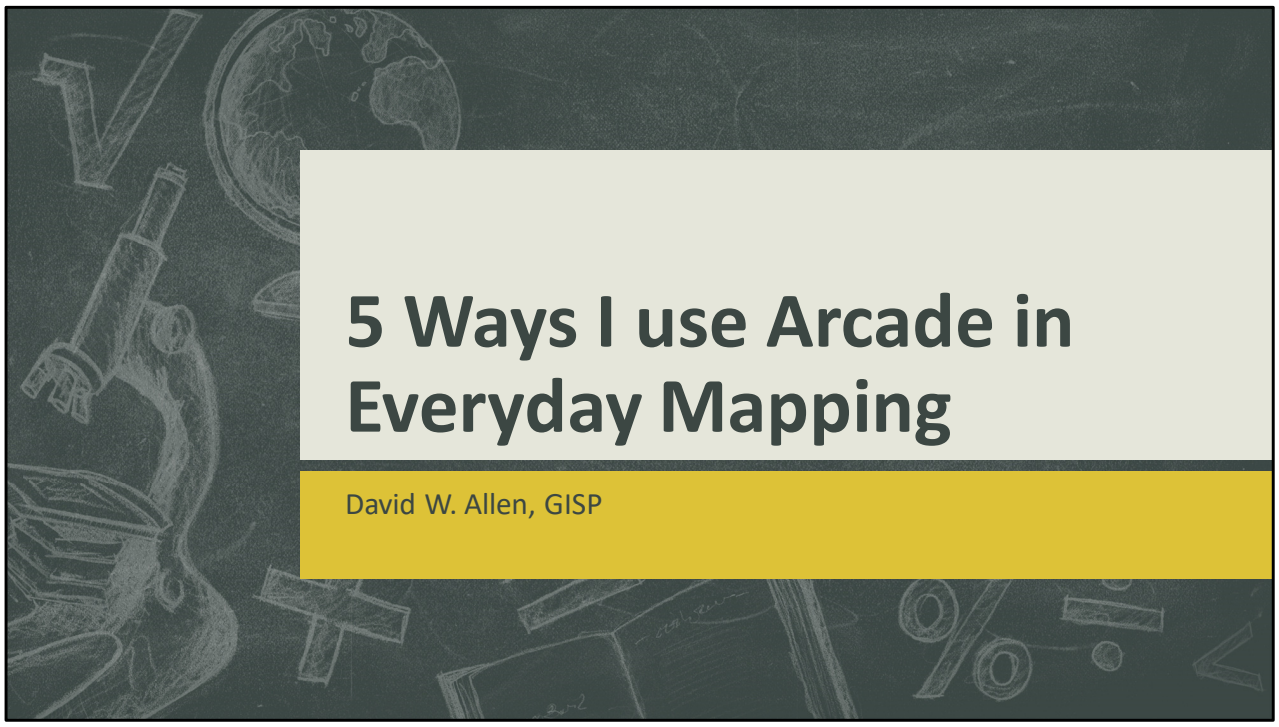


5 Ways I use Arcade in Everyday Mapping

David W. Allen, GISP

5 Ways I use Arcade in Everyday Mapping

David Allen

34 years with the City of Euless

Taught GIS at Tarrant County College for almost 20 years

Author of many GIS textbooks including:

'Writing Arcade Expressions for ArcGIS Pro'

5 Ways I use Arcade in Everyday Mapping

Arcade

Esri invented a new scripting language with ArcGIS Pro. The main feature of this language is that the scripts written in Pro will work seamlessly in ArcGIS Online without changing a single bit of code. (and vice-versa)

Any map you wish to use as a web service or hosted feature layer can use Arcade, and when you consume the layer in a web app the Arcade you wrote will still work.

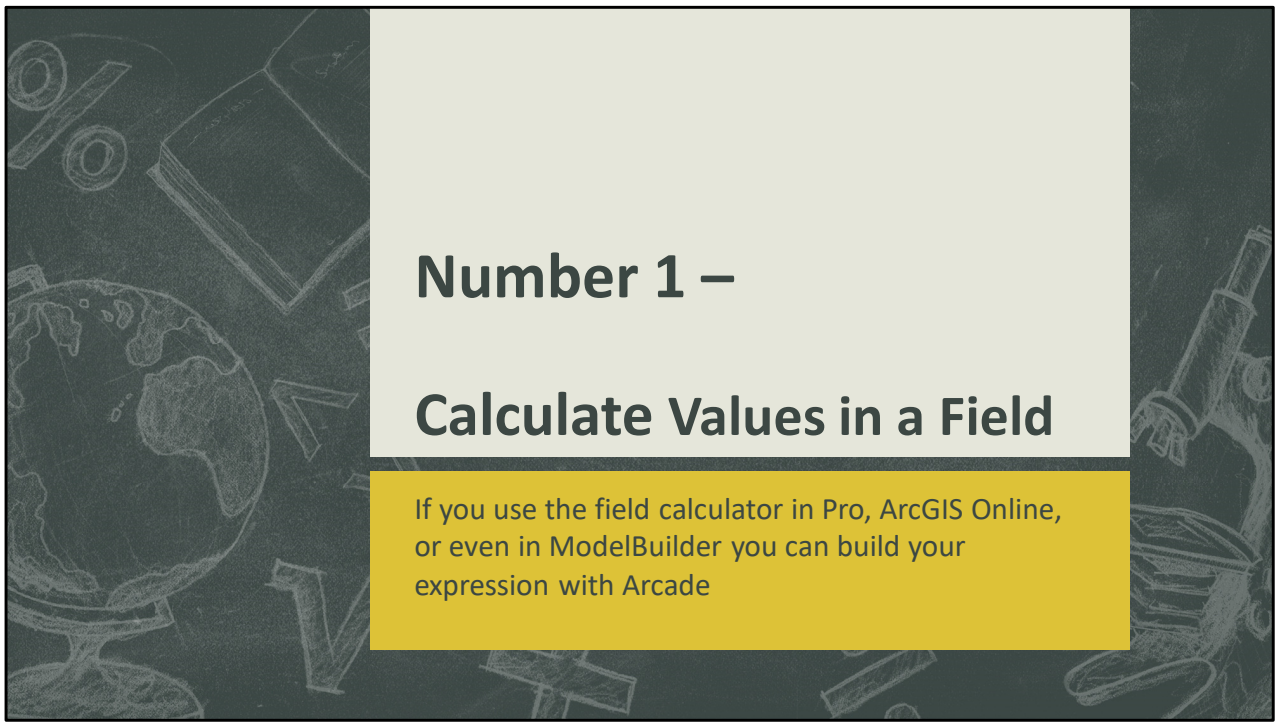
(can't say that with Python)

5 Ways I use Arcade in Everyday Mapping

Arcade has also become the default scripting language within many of the Esri products, and enhancements are being made with every release.

In this presentation I'll demonstrate ways that I use Arcade in everyday mapping – and you can too!

(if you get these slides later they will include all of the scripts in the slide's notes)


Number 1 –
Calculate Values in a Field
If you use the field calculator in Pro, ArcGIS Online, or even in ModelBuilder you can build your expression with Arcade

Calculate values in a field

Arcade has all the common field operators like you would expect -

+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment by one
--	Decrement by one

==	Equal to
!=	Not equal to
>	Greater than
>=	Greater than or equal
<	Less than
<=	Less than equal to

||	Logical or
&&	Logical and
!	Logical not

All the Arcade operators can be found on this page:
https://developers.arcgis.com/arcade/guide/operators/

Calculate values in a field

Arcade also has text handling functions.

- Concatenate
- Count
- Find
- FromCharCode
- FromCodePoint
- Guid
- Left
- Lower
- Mid
- Proper
- Replace
- Right
- Split
- StandardizeGuid
- Text
- ToCharCode
- ToCodePoint
- ToHex
- Trim
- Upper

All the Arcade operators can be found on this page:
https://developers.arcgis.com/arcade/guide/operators/

Calculate values in a field

The tax data I get has the values for City and State in the same field, but I need it in separate fields. The Arcade text functions can be used to split them out.

Owner_CityState
EULESS, TX
HURST, TX
DALLAS, TX
EULESS, TX
EULESS, TX
EULESS, TX
LOS ANGELES, CA
TYLER, TX
EULESS, TX
EULESS, TX
SOUTHLAKE, TX
EULESS, TX

If I can find the index number of the comma in the string, then I can split out all the text to the left into the City field. The characters are indexed starting at the left with 0.

The Arcade function Find() will return the index number of the comma.

```
var commaLoc = Find($feature['TadData_2022_10.Owner_CityState'], ',')
```

The Arcade command reference is here:
https://developers.arcgis.com/arcade/function-reference/

var commaLoc = Find($feature['TadData_2022_10.Owner_CityState'], ',')

Calculate values in a field

Next the Left() function can be used to pull a certain number of characters from the left end of the input string. Since the comma location is indexed starting with 0, the commaLoc value is in fact the number of characters to fetch.

```
var commaLoc = Find($feature['TadData_2022_10.Owner_CityState'], ',')
var cityName = Left($feature['TadData_2022_10.Owner_CityState'],commaLoc)
return cityName
```

Owner City
EULESS
EULESS
EULESS
HURST
HURST
SAN DIEGO
EULESS
EULESS
EULESS
COLLEYVILLE
EULESS
SAN DIEGO

var commaLoc = Find($feature['TadData_2022_10.Owner_CityState'], ',')
var cityName = Left($feature['TadData_2022_10.Owner_CityState'],commaLoc)
return cityName

Calculate values in a field

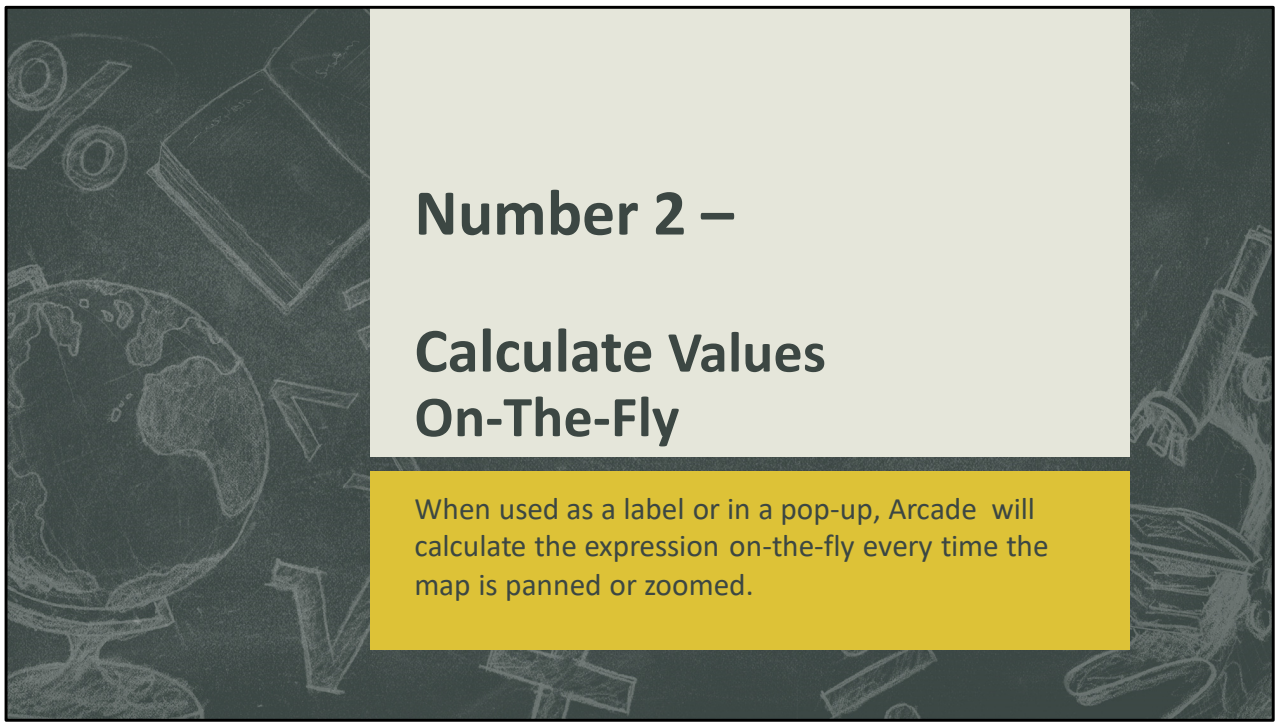
The Right() function can pull the two character state abbreviation off the right side of the string.

```
right($feature['TadData_2022_10.Owner_CityState'],2)
```

The result splits the data from one field into two.

Owner State
TX
TX
TX
TX
TX
CA
TX
TX
TX
CA
TX
CA

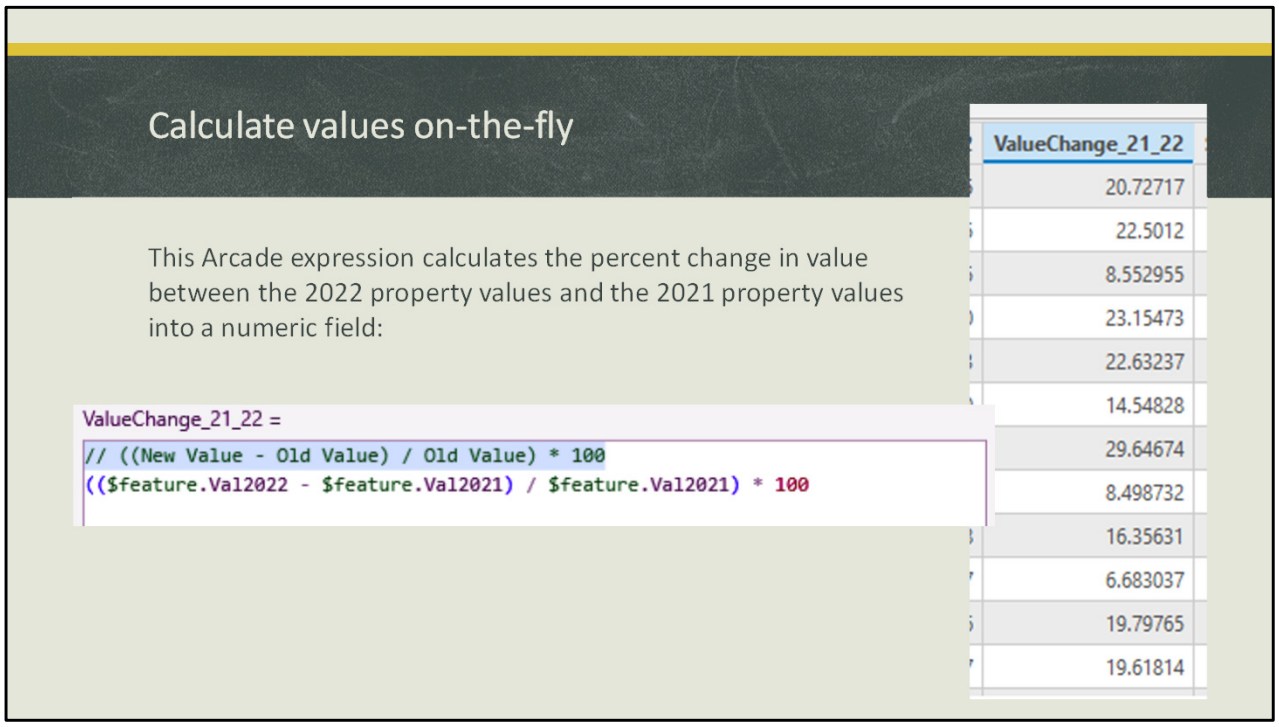
All the Arcade operators can be found on this page:
https://developers.arcgis.com/arcade/guide/operators/



Number 2 –

Calculate Values On-The-Fly

When used as a label or in a pop-up, Arcade will calculate the expression on-the-fly every time the map is panned or zoomed.



Calculate the change in value between two numbers.
((New Value – Old Value) / Old Value) * 100

(($feature.Val2022 - $feature.Val2021) / $feature.Val2021) * 100

Calculate values on-the-fly

The same Arcade expression used before could also cast the results as text and be used to format a label on the map display, or included in a pop-up window. The results wouldn't have to be stored in a field but can be generated on-the-fly:

Expression

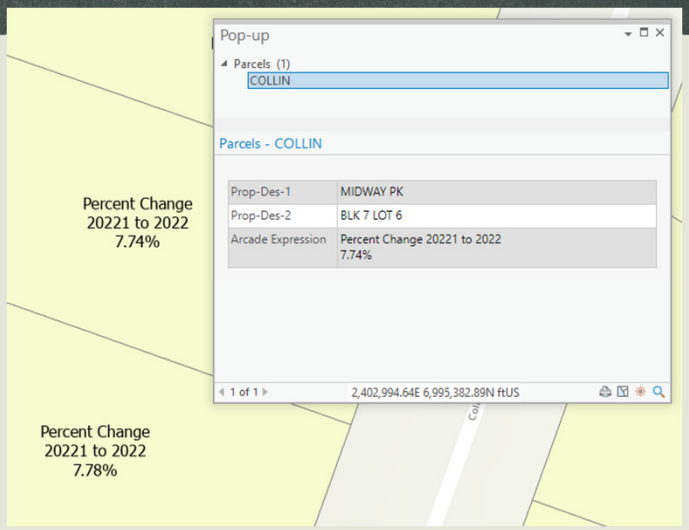
```
'Percent Change 20221 to 2022' +
TextFormatting.NewLine +
text(Round((($feature.Val2022 - $feature.Val2021) / $feature.Val2021) * 100,2)) + '%'
```

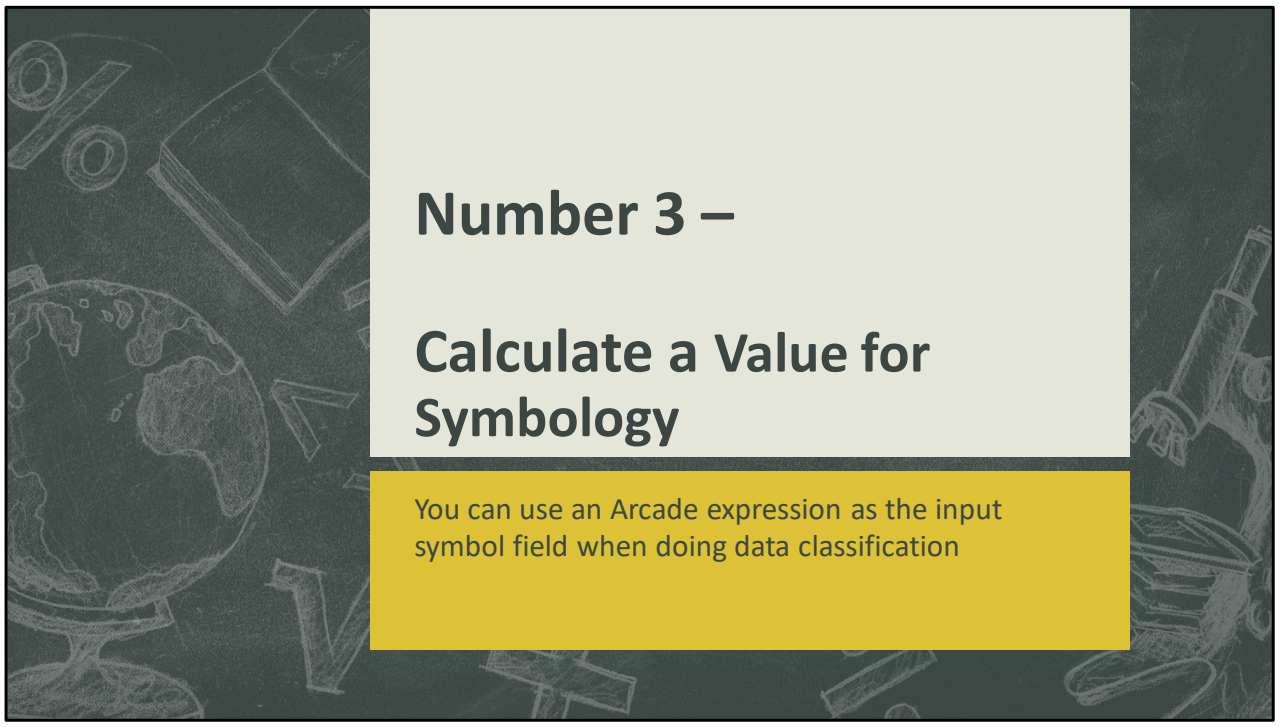
Cast the results as Text and round to two decimal places

```
'Percent Change 20221 to 2022' +
TextFormatting.NewLine +
text(Round((($feature.Val2022 - $feature.Val2021) / $feature.Val2021) * 100,2)) + '%'
```

Calculate values on-the-fly

The label and pop-up display were both formatted with an Arcade expression




Number 3 –
Calculate a Value for Symbology
You can use an Arcade expression as the input symbol field when doing data classification

Calculate a value for symbology

The symbology generator allows an Arcade expression to be used as the classification field. In this use the guys in charge of our reclaimed water system wanted to track their valve inspections. So I made a map show the time since the last inspections and color coded by year.

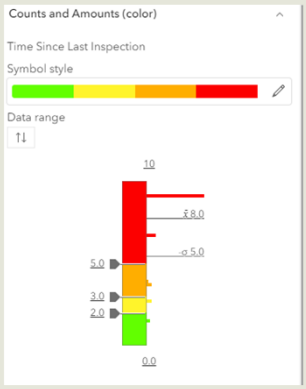
To do this, I use an Arcade function called DateDiff() to find the difference between today and the last inspection, then round it to whole years.

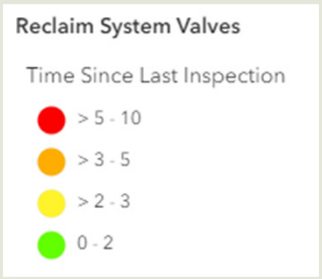
```
Round(DateDiff(Today(),Date($feature.InspDate),'years'),1)
```

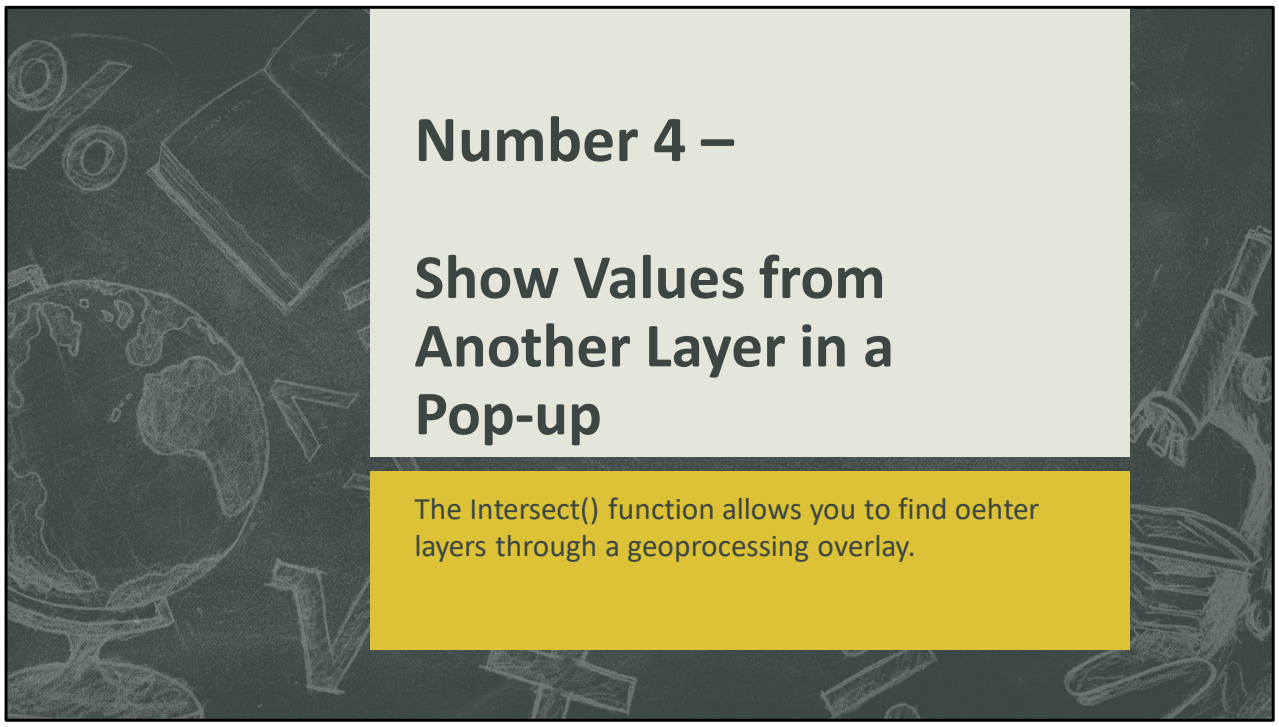
Round(DateDiff(Today(),Date($feature.InspDate),'years'),1)

Calculate a value for symbology

The symbol (style in ArcGIS Online) generator only sees the results of the calculation – so it sees a bunch of single digit years.






Number 4 –
Show Values from Another Layer in a Pop-up
The Intersect() function allows you to find oehter layers through a geoprocessing overlay.

Show values from another layer in a pop-up

I have a layer of vacant land that is the hub of activity during high grass and weeds season – but that data doesn't have any links to the ownership data. So I used an Arcade function to intersect a layer that does have a link, and use that link to retrieve the ownership in the pop-up.

Vacant Parcel Index Number : 127

Tax Roll Lookup	40800938
Mowing Contractor	
Prop-Des-1	BEAR CREEK ESTATES-EULESS
Prop-Des-2	BLK 10 LOT 1

Basic Information

OWNER
ZKB PROPERTIES LLP
500 TOWNE COVE
IRVING, TX 75061-9307

TYPE
Real

Show values from another layer in a pop-up

The Arcade function FeatureSetByPortalItem() identifies which layer to use, and the Intersect() function does the feature overlay. Then a loop looks for the desired field value in the overlay layer.

```
var intersectLayer = Intersects(
FeatureSetByPortalItem(Portal('https://maps3.eulesstx.gov/portal/'),
'a75db68350e14e28be4a67399e0cf4b2', 18),$feature)

for (var f in intersectLayer){
   return f.SKEY
}
```


```
var intersectLayer =
Intersects(FeatureSetByPortalItem(Portal('https://maps3.eulesstx.gov/portal/'),
'a75db68350e14e28be4a67399e0cf4b2', 18),$feature)

for (var f in intersectLayer){
   return f.SKEY
}
```

Show values from another layer in a pop-up

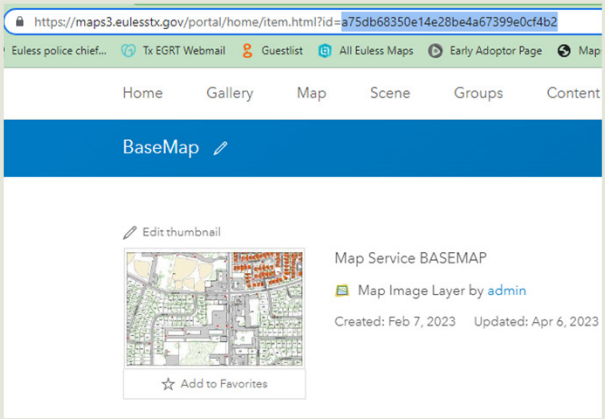
Let's break that down … First define your portal location:

Portal('https://maps3.ci.eules.tx.us')

Within that portal is a map service that contains the data I want …

in this case point data from the parcel ownership. And I want to record

id=a75db68350e14e28be4a67399e0cf4b2



Show values from another layer in a pop-up

Then within the Map Service Layer, I need to know the index number of the layer I want.

In this case, it's **18**

For a Hosted Feature Layer, the index is in the ID

id=f3ee658376c24548a0d2e532769ef5c2**&sublayer=3**

Show values from another layer in a pop-up

Now put it all together.

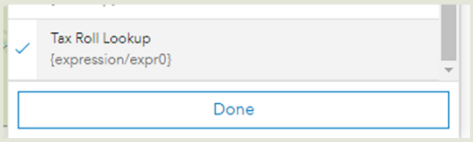
```
Var intersectLayer =
Intersects(                                   ## Arcade overlay function
FeatureSetByPortalItem(                       ## Arcade Function to get the layer
Portal('https://maps3.eulesstx.gov/portal/')  ## Identifies the Portal location
'a75db68350e14e28be4a67399e0cf4b2', 18)       ## Identifies the layer ID and index number
$feature)                                     ## Tells the Intersect to use selected feature
for (var f in intersectLayer){                ## Loops through the interest feature to find
   return f.SKEY                              ## The value of the field SKEY
}
```

```
var intersectLayer = Intersects(
FeatureSetByPortalItem(Portal('https://maps3.eulesstx.gov/portal/'),
'a75db68350e14e28be4a67399e0cf4b2', 18),$feature)

for (var f in intersectLayer){
   return f.SKEY
}
```

Show values from another layer in a pop-up

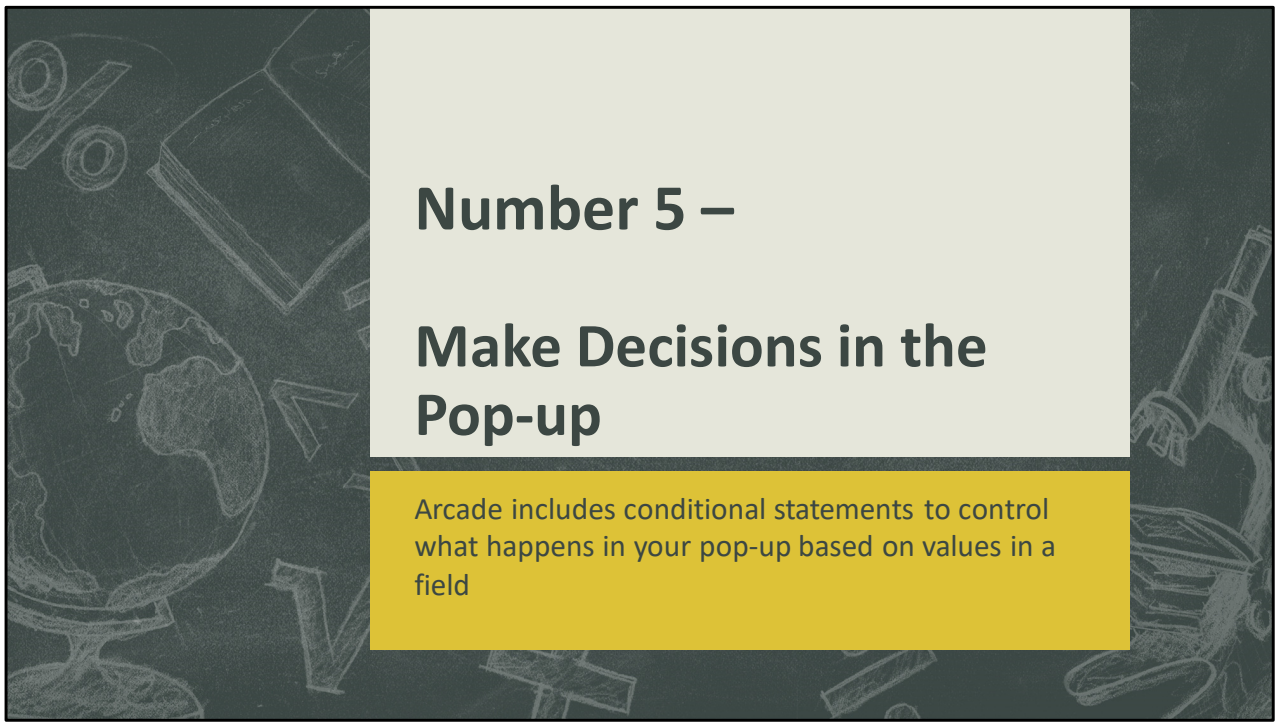
As far as the pop-up config window knows, this is just a regular field available for use.



And this can pull data from any Portal you can

Identify, INCLUDING the Esri Living Atlas

– and the layer doesn't have to be in the map.


Number 5 –
Make Decisions in the Pop-up
Arcade includes conditional statements to control what happens in your pop-up based on values in a field

Make decisions in the pop-up

In the Tarrant Appraisal District data there's a field called SB247 Flag which denotes citizens that have asked for their personal info to be screened on public websites. We have to abide by that, so if the SB247 Flag is anything but 'N' we can't show a link to the ownership data.

Using the Arcade When() function I can check to see what the flag is and change the display on-the-fly.

If it's OK to show owner info, the link is
https://taxonline.tarrantcounty.com/TaxPayer/Accounts/AccountDetails?taxAccountNumber=000{SKEY}

If it's restricted, the link is
https://maps3.eulesstx.gov/icons/securedata.png

Make decisions in the pop-up

The Arcade When() function accepts a condition (or conditions), then returns a result based on that condition.

When the flag field equates to N it send back the URL to the tax website.

Otherwise it displays an icon saying the information is secured.

```
return When($feature.SB247_Flag == 'N',
'https://taxonline.tarrantcounty.com/TaxPayer/Accounts/AccountDetails?taxAccountNumber=000' +
$feature.SKEY, 'https://maps3.eulesstx.gov/icons/SecureData.png')

```

return When($feature.SB247_Flag == 'N',
'https://taxonline.tarrantcounty.com/TaxPayer/Accounts/AccountDetails?taxAccountNumber=000' +
$feature.SKEY, 'https://maps3.eulesstx.gov/icons/SecureData.png')

Make decisions in the pop-up

This is also done for the other ownership fields.

```
return when($feature.SB247_Flag == 'N',$feature.Owner_Name,'Data Secured')
```

```
return when($feature.SB247_Flag == 'N',$feature.Owner_Address,'Data Secured')
```

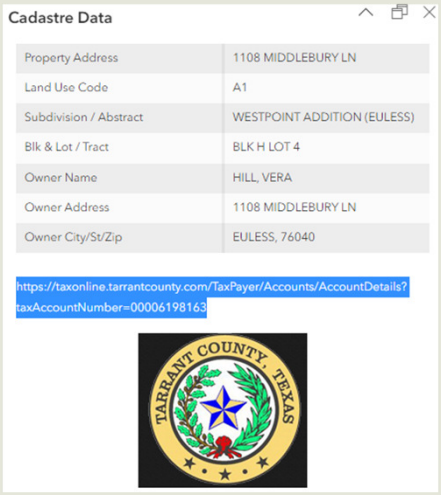
```
return when($feature.SB247_Flag == 'N',$feature.Owner_City + ', ' + $Feature.Owner_State +
' ' + $feature.Owner_Zip,'Data Secured')
```

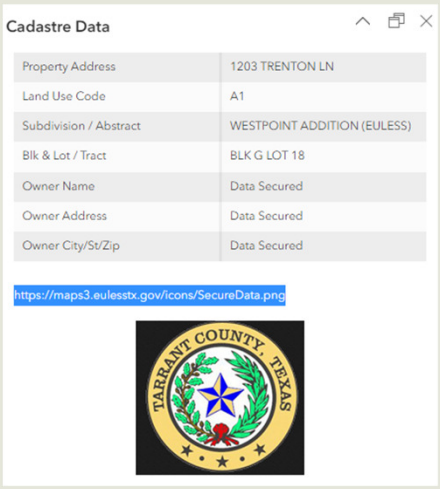
return when($feature.SB247_Flag == 'N',$feature.Owner_Name,'Data Secured’)

return when($feature.SB247_Flag == 'N',$feature.Owner_Address,'Data Secured’)

return when($feature.SB247_Flag == 'N',$feature.Owner_City + ', ' +
$Feature.Owner_State +
' ' + $feature.Owner_Zip,'Data Secured')

Make decisions in the pop-up





Calculate values in a field

You don’t have to use impressively long Arcade expressions to make a difference. These simple implementations make a big difference in how data is created and displayed.

Thanks for hanging around and seeing a little Arcade.

Be sure to get the slides from SCAUG because all of the code is in the slide notes.

David Allen, GISP

You contact me at Info@GISGuidebooks.com