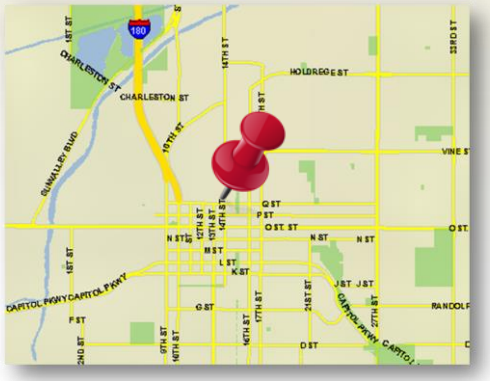
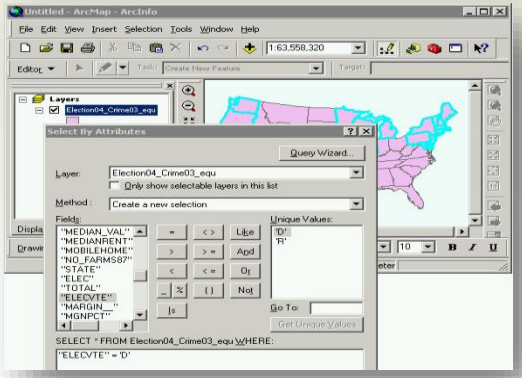
Working with tabular data in GIS

Patricia Carbajales-Dale
clemsongis.org

Overview



Geocode



Query

INPUT

OBJECT ID#	Landuse Code
1	2
2	0
3	1

Join Fields

+

Landuse Code	Landuse Type
0	Unclassified
1	shrub
2	water

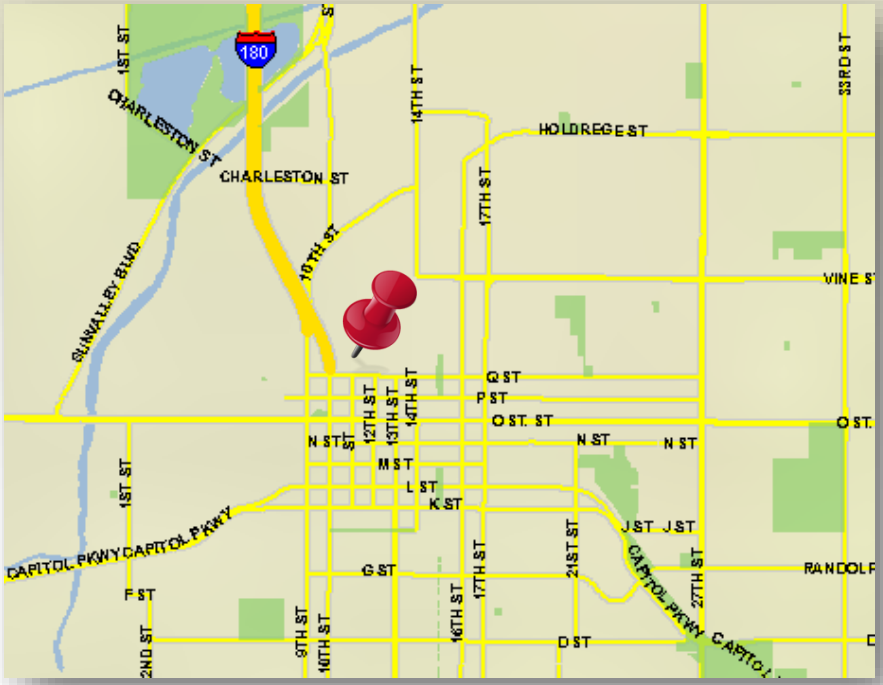
OUTPUT

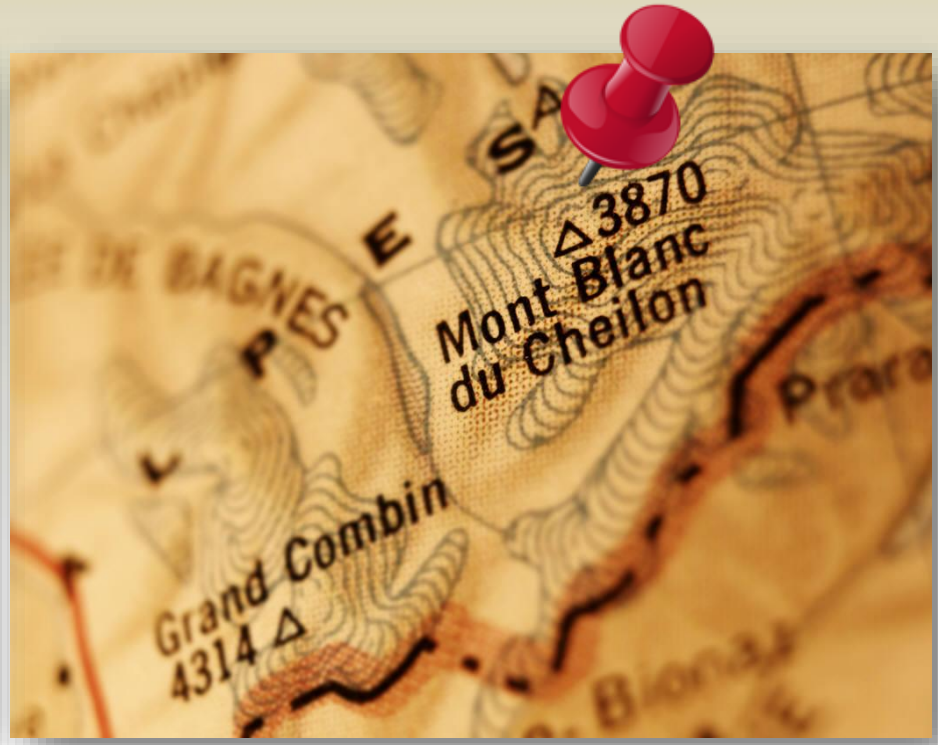
OBJECT ID#	Landuse Code	Join Table Landuse Code	Join Table Landuse Type
1	2	2	water
2	0	0	Unclassified
3	1	1	shrub

Join

What is Geocoding?

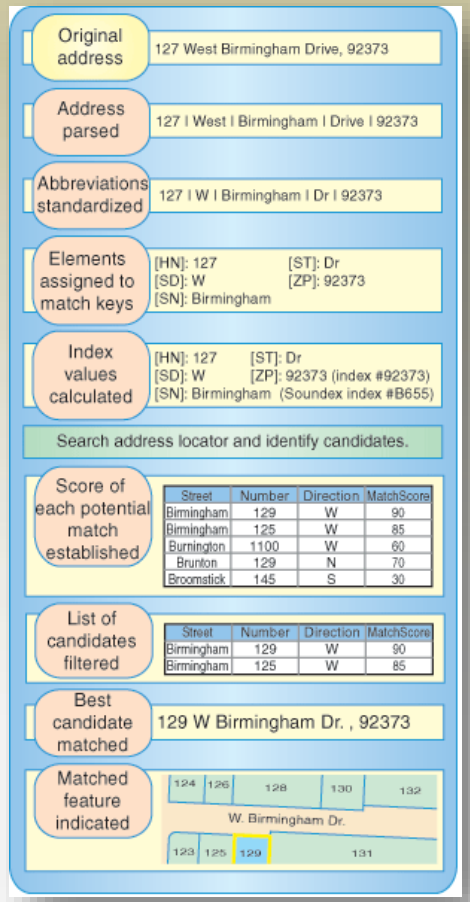
- Description of a Location → Location on a Map
  - Street Address, Intersection
  - Place name (park, monument)
  - Buildings, zip codes, ip addresses

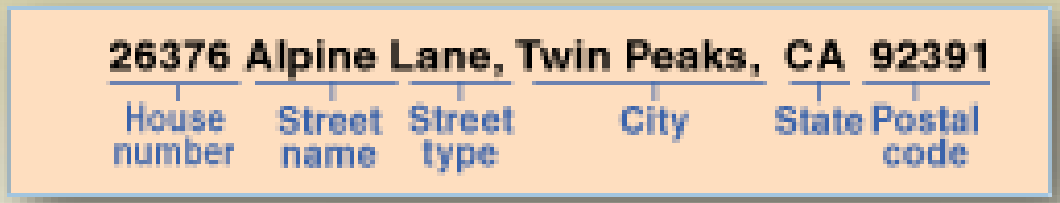


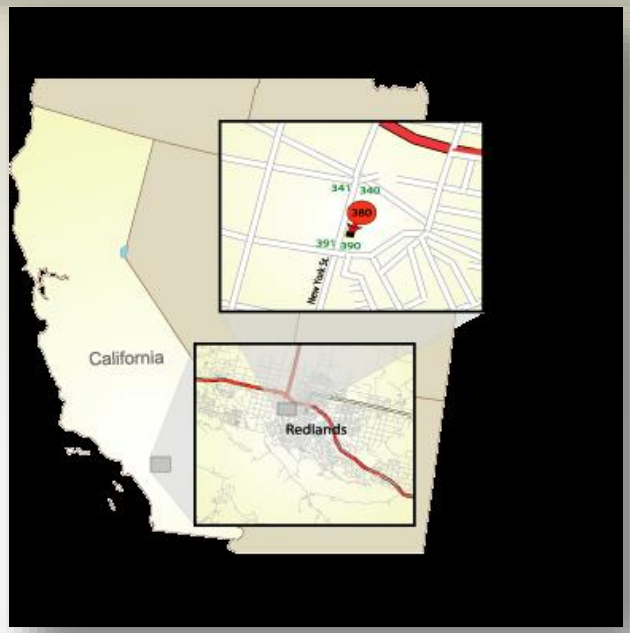


Geocoding

Geocoding is the process of **assigning a location**, usually in the form of coordinate values (points), to an **address** by comparing the descriptive location elements in the address to those present in the reference materia

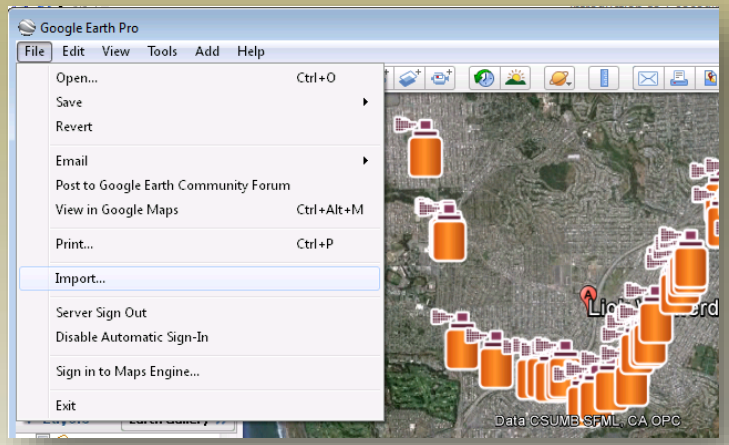


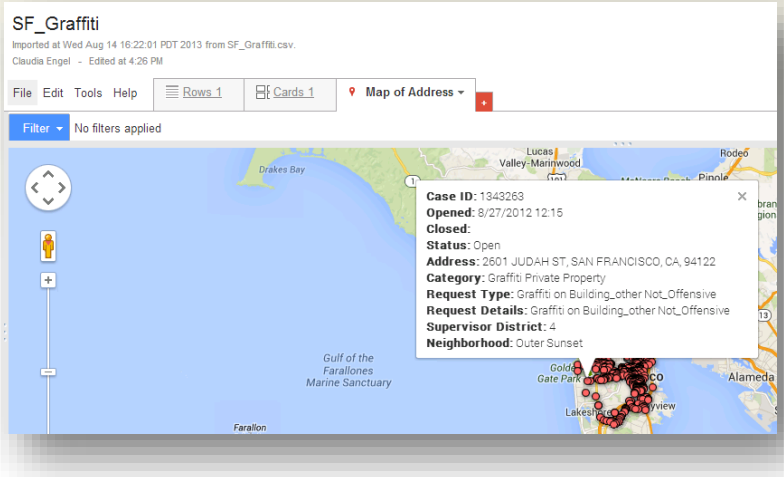




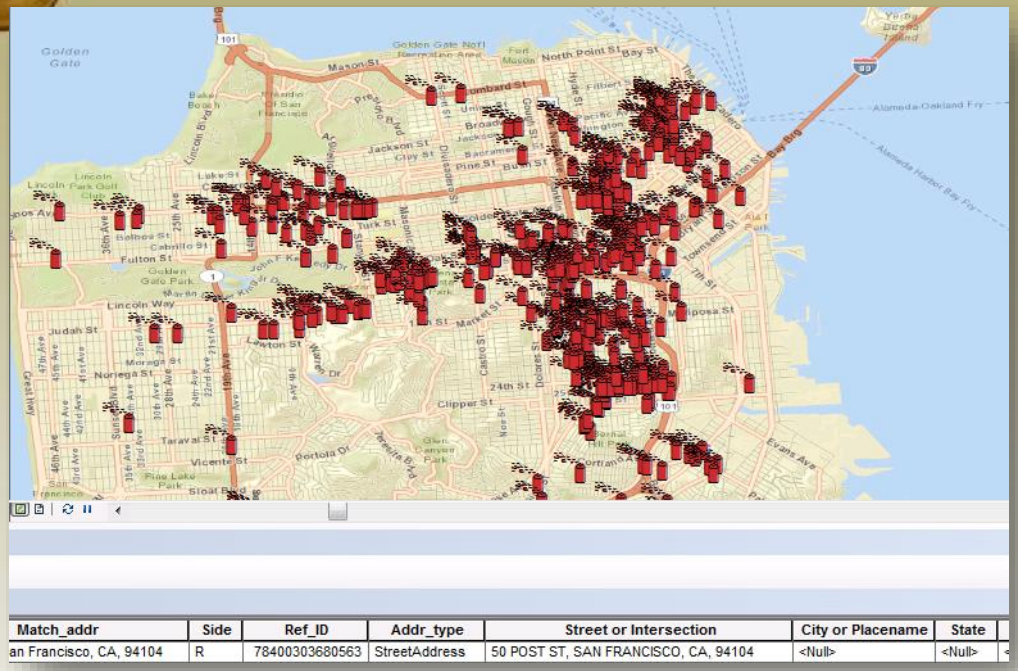
Geocoding Options

- Google
  - Google Earth Pro
  - Google Fusion Tables
  - Google Maps API
- ArcGIS
  - ArcMap
  - ArcGIS Online
  - Building your own address locator
- Automated Processes
- Web Services





Geocoding in ArcGIS



Match_addr	Side	Ref_ID	Addr_type	Street or Intersection	City or Placename	State
an Francisco, CA, 94104	R	78400303680563	StreetAddress	50 POST ST, SAN FRANCISCO, CA, 94104	<Null>	<Null>

Specifications

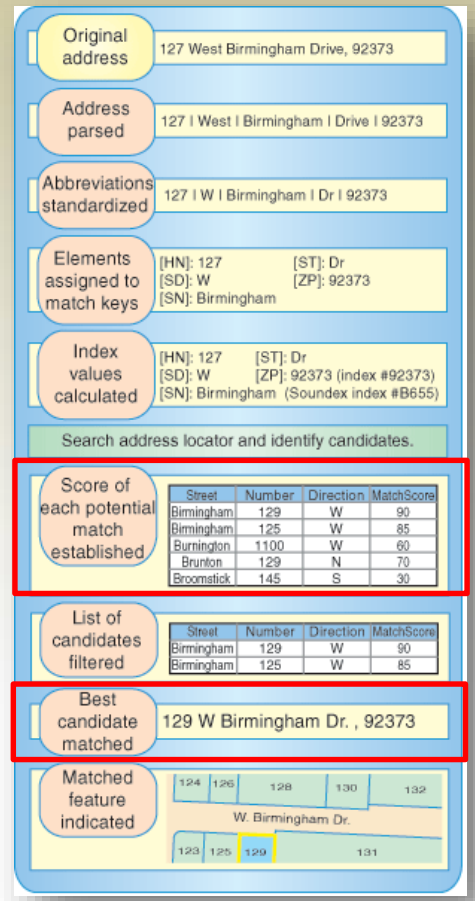
- Import .csv or Excel files
- No limit on number available
- Max. control on the engine
- Flexible entry format
- No spaces or special characters on field names

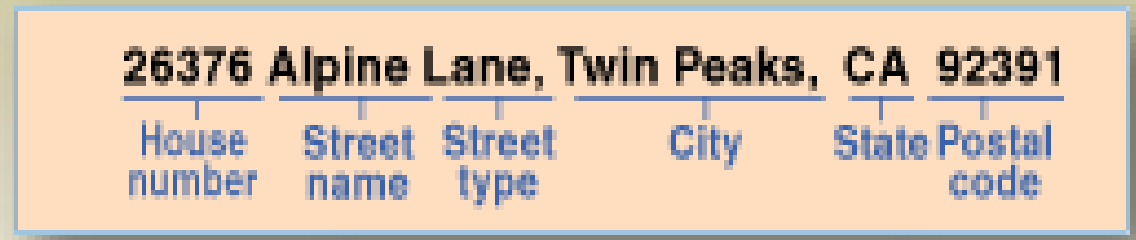
Results (out of 500 records):

- 496 records geocoded in place
- 4 records rematched manually

How does it work?

- Standardize your Address Data
- Compare to Reference Data (geographic data)
- Return best match of results

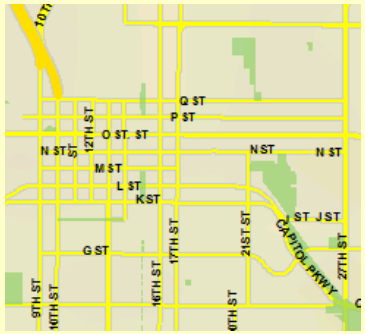




Address Locator

Address Locator

Reference Data



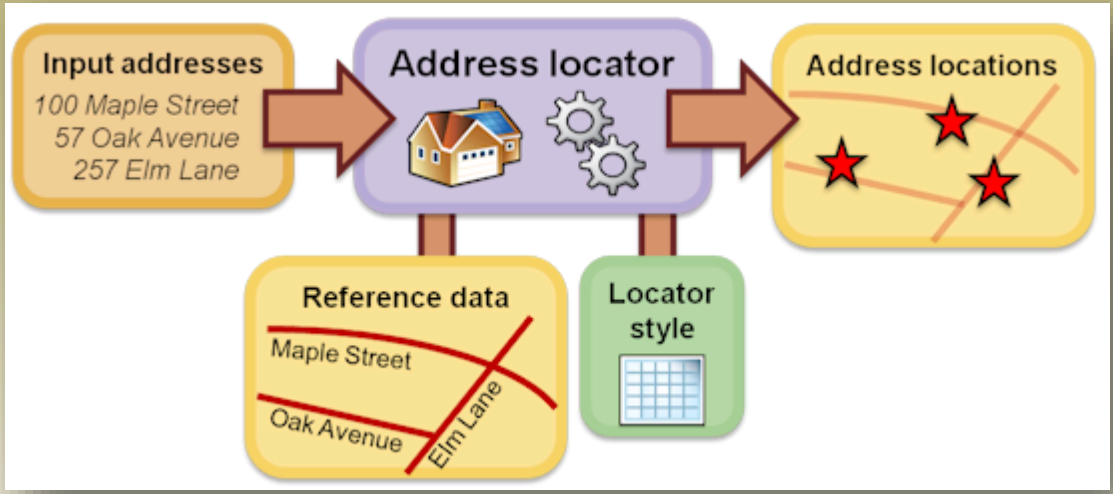
Rules

Even house numbers are on the right side of the street

Street Components

Avenue = Ave
Street = St
Highway = Hwy

Geocoding Workflow in ArcGIS

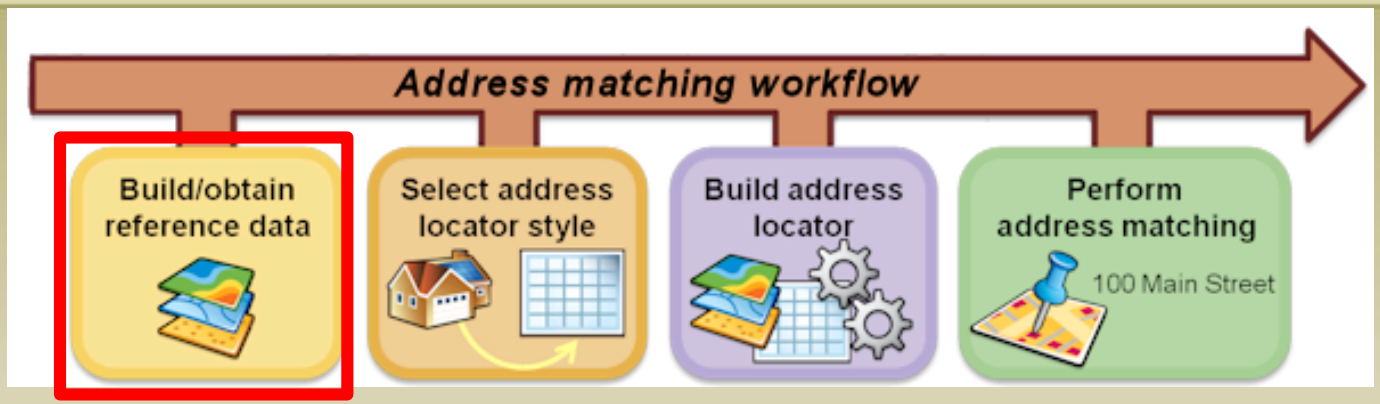


Build or obtain reference data

Determine address locator style

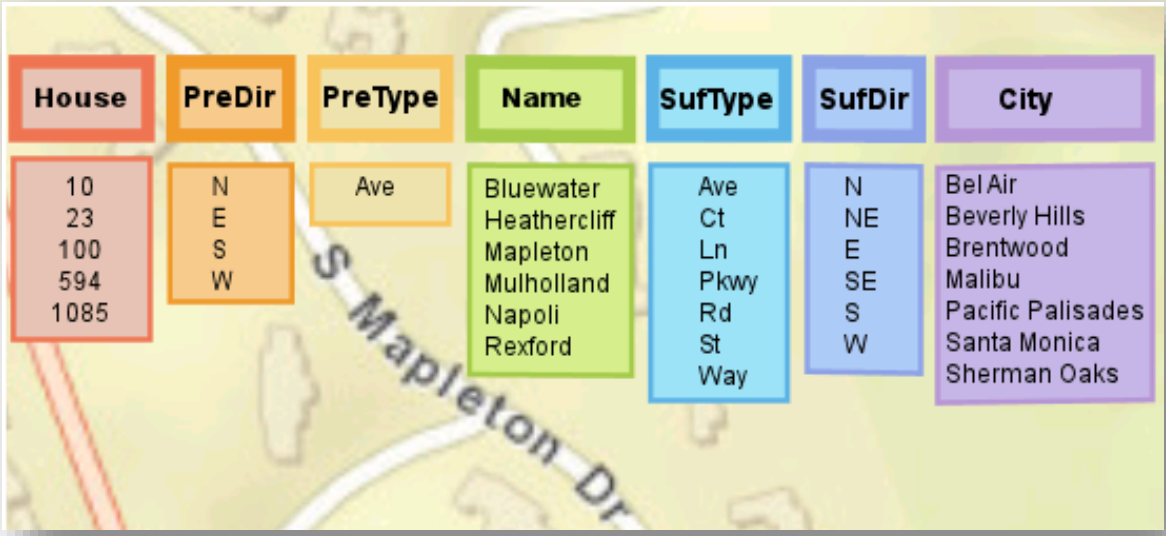
Locate addresses & rematch remaining addresses

Reference Data



Snapshot of Geographic Information

- ESRI Street Data
- US Census TIGER roads
- Parcels
- Gazetteers
- General Place names
- Zip Codes/ZCTAs
- NavTec, TeleAtlas



Querying Tables

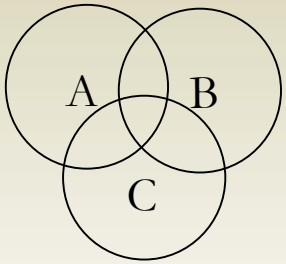
- Queries are the most common operations in a spatial database
  - used to **find features** that meet certain criteria
  - used to **isolate features** for future analysis (subset or filter data)
- Structured Query Language (**SQL**)
  - standard language for retrieving and updating information in a database
- Most common operation is the **SELECT**
  - retrieves data from a table or multiple related tables
  - SELECT * FROM "*Table*" WHERE "*Field = Value*"
  - returns a subset of records, e.g. restricts records based on some condition
- Can be simple or complex
  - simple: "area" > 20
  - complex: ("area" > 20) AND ("area" < 50)

SQL

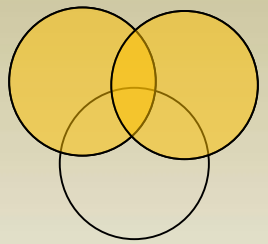
- **Set Algebra**
  - basis for simple query expressions
    - = (equal)
    - > (greater than)
    - < (less than)
    - <> (not equal to)
  - can be used on both strings and numbers
- **Boolean Algebra**
  - basis for complex query expressions
    - **NOT**: accepts value of one input and outputs opposite value
    - **AND**: accepts two values as input and outputs the intersection of both
    - **OR**: accepts two values as input and outputs the sum of both
  - parentheses may be required and the order of precedence is important

Boolean Algebra

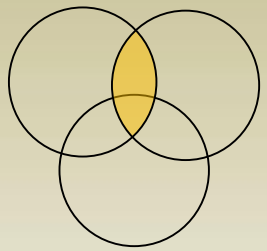
Regions



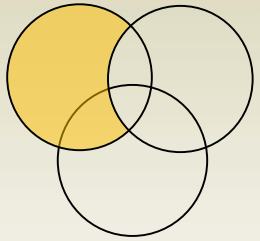
A ***OR*** B



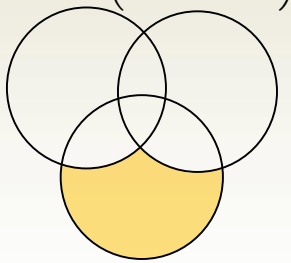
A ***AND*** B



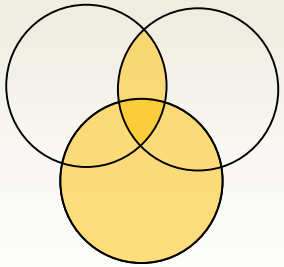
A ***NOT*** B



NOT (A ***OR*** B)



A ***AND*** B ***OR*** C



Simple Queries

ID	Area	Landuse	Municipality
1	10.5	Urban	City
2	330.3	Farm	County
3	2.4	Suburban	Township
4	96.0	Suburban	County
5	22.1	Urban	City
6	30.2	Farm	Township
7	4.4	Urban	County

Find all records with ***Area*** greater than 20.0 ⟵ Area > 20.0

Find all ***Urban Landuse*** ⟵ Landuse = Urban

Find all records with ***Area*** less than or equal to 55.0 ⟵ Area <= 55.0

Find all ***Non-Urban Landuse*** ⟵ Landuse <> Urban

Compound Queries

Find all the municipalities that are not urban cities

NOT [(Landuse = Urban) AND (Municipality = County)]

- Landuse = Urban ⟵ Set1
- Municipality = County ⟵ Set2
- Set1 ***AND*** Set2 ⟵ Set3
- ***NOT*** (Set3)

ID	Area	Landuse	Municipality
1	10.5	Urban	City
2	330.3	Farm	County
3	2.4	Suburban	Township
4	96.0	Suburban	County
5	22.1	Urban	City
6	30.2	Farm	Township
7	4.4	Urban	County

Compound Queries

Excluding counties, which municipalities are not urban?

[NOT (Landuse = Urban)] AND [NOT (Municipality = County)]

- Landuse = Urban ⟵ Set1
- ***NOT*** (Set1) ⟵ Set2
- Municipality = County ⟵ Set3
- ***NOT*** (Set3) ⟵ Set4
- Set2 ***AND*** Set4

ID	Area	Landuse	Municipality
1	10.5	Urban	City
2	330.3	Farm	County
3	2.4	Suburban	Township
4	96.0	Suburban	County
5	22.1	Urban	City
6	30.2	Farm	Township
7	4.4	Urban	County

Querying Tables

ID	Area	Landuse	Municipality
1	10.5	Urban	City
2	330.3	Farm	County
3	2.4	Suburban	Township
4	96.0	Suburban	County
5	22.1	Urban	City
6	30.2	Farm	Township
7	4.4	Urban	County

Find all **Suburban** and **Urban** Landuse **greater than 20.0** and **less than 5.0**
Find all **County Non-Urban** Landuse **less than 100.0**
Find all **Urban Cities greater than 20.0**

Other SQL Statements

- IN
  - search for **several strings or values** in a field; alternative to multiple ORs

```
state IN ('Alabama', 'Alaska', 'California', 'Florida')
```

- LIKE
  - used with **wildcards** to build a **partial string search**

```
state LIKE 'Mi%'
```

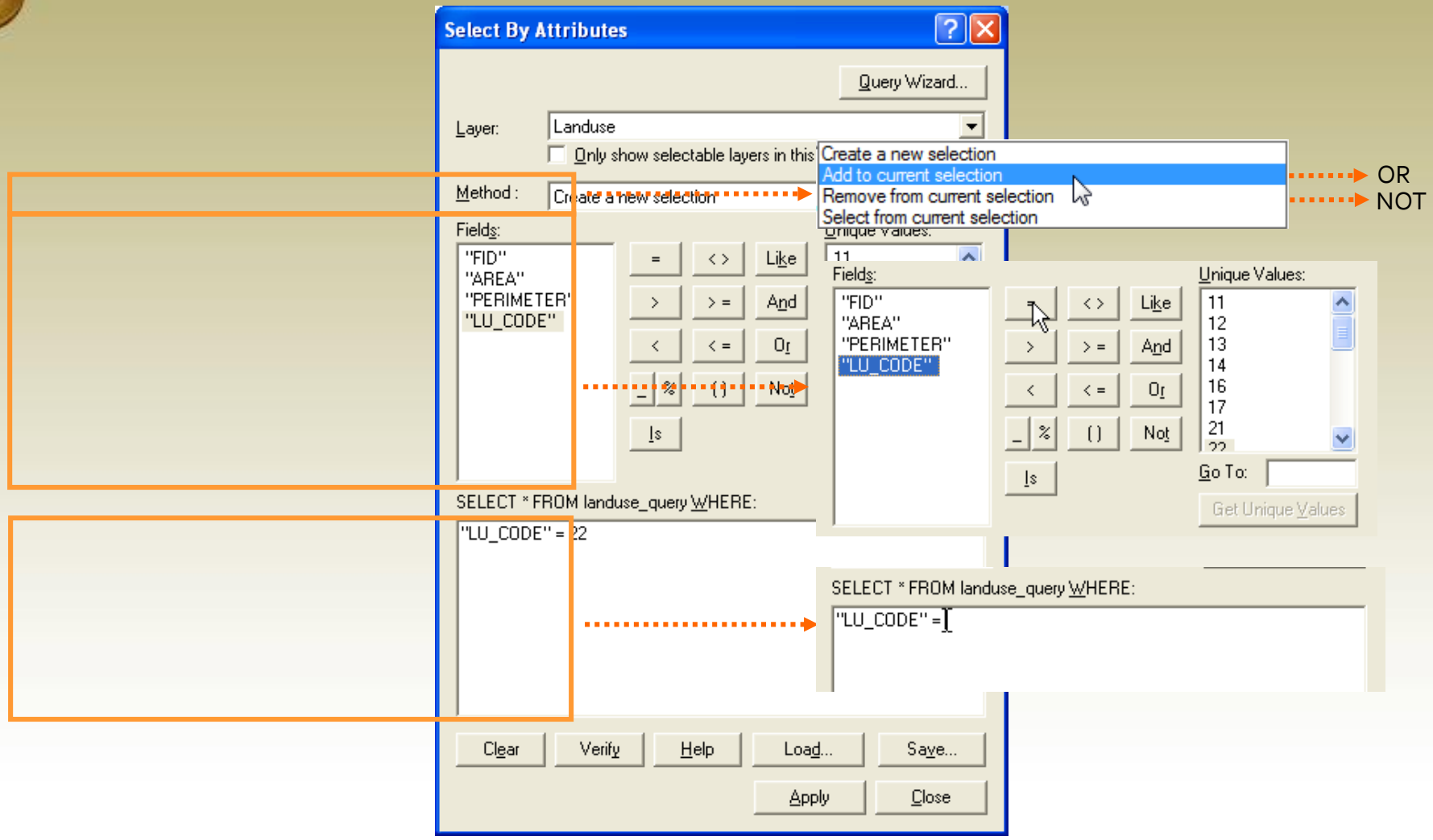
- % (shapefile); * (geodatabase)
  - **wildcard** representing any **multiple** characters; used with LIKE

```
place_name LIKE '%ort%'
```

- _ (shapefile); ? (geodatabase)
  - **wildcard** representing any **single** character; used with LIKE

```
first_name LIKE '_arry'
```

Querying Tables



Joins

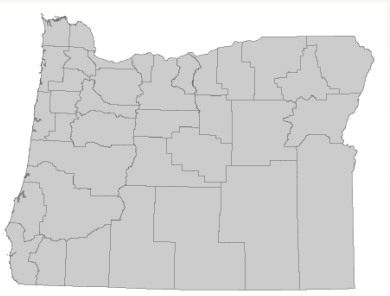
Joins allow us to **append** new data to existing geometry or tables, extending their informational range.

Geometry and attribute data are not always stored together.

Any data that shares a common attribute with a spatial dataset can be mapped.

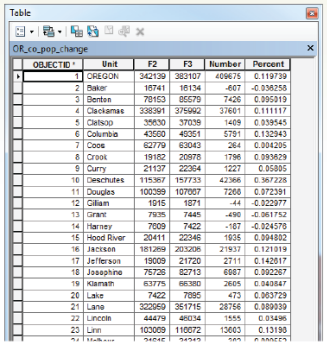
- Map new attributes to existing features
- Extend analytical range of spatial data
- Based on common attributes between tables
- Everything is spatial!

Boring Geometry



+

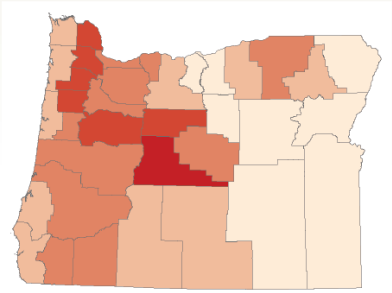
Boring Statistics



OBJECTID *	Unit	F2	F3	Number	Percent
1	OREGON	342139	383107	409675	0.119739
2	Baker	16741	16134	-607	-0.036258
3	Benton	78153	85579	7426	0.095019
4	Clackamas	338391	375992	37601	0.111117
5	Clatsop	35630	37039	1409	0.039545
6	Columbia	43660	49351	5791	0.132643
7	Coos	62779	63043	264	0.004205
8	Crook	19182	20978	1796	0.093629
9	Curry	21137	22364	1227	0.05805
10	Deschutes	115367	157733	42366	0.367228
11	Douglas	100399	107667	7268	0.072381
12	Gilliam	1915	1871	-44	-0.022977
13	Grant	7935	7445	-490	-0.061752
14	Harney	7609	7422	-187	-0.024576
15	Hood River	20411	22346	1935	0.094802
16	Jackson	181269	203206	21937	0.121019
17	Jefferson	19009	21720	2711	0.142617
18	Josephine	76728	82713	6687	0.092267
19	Klamath	63775	66380	2605	0.040847
20	Lake	7422	7895	473	0.063729
21	Lane	322959	351715	28756	0.089039
22	Lincoln	44479	46034	1555	0.03496
23	Linn	103069	116672	13603	0.13198

=

Awesomeness!



Joins

Join one table to another table or layer based a common field.

Attributes of the join table are appended to the input table.

A match is made when the input field and output field values are equal.

- Common identifier or "key"
- Field names do not need to match
- Field type must be the same (numbers, text, etc.)
- Field data should be the same
- The join is dynamic

INPUT

OBJECT ID#	Landuse Code
1	2
2	0
3	1

Join Fields

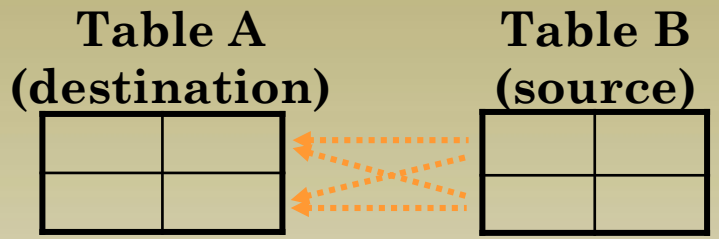
\+

Landuse Code	Landuse Type
0	Unclassified
1	shrub
2	water

OUTPUT

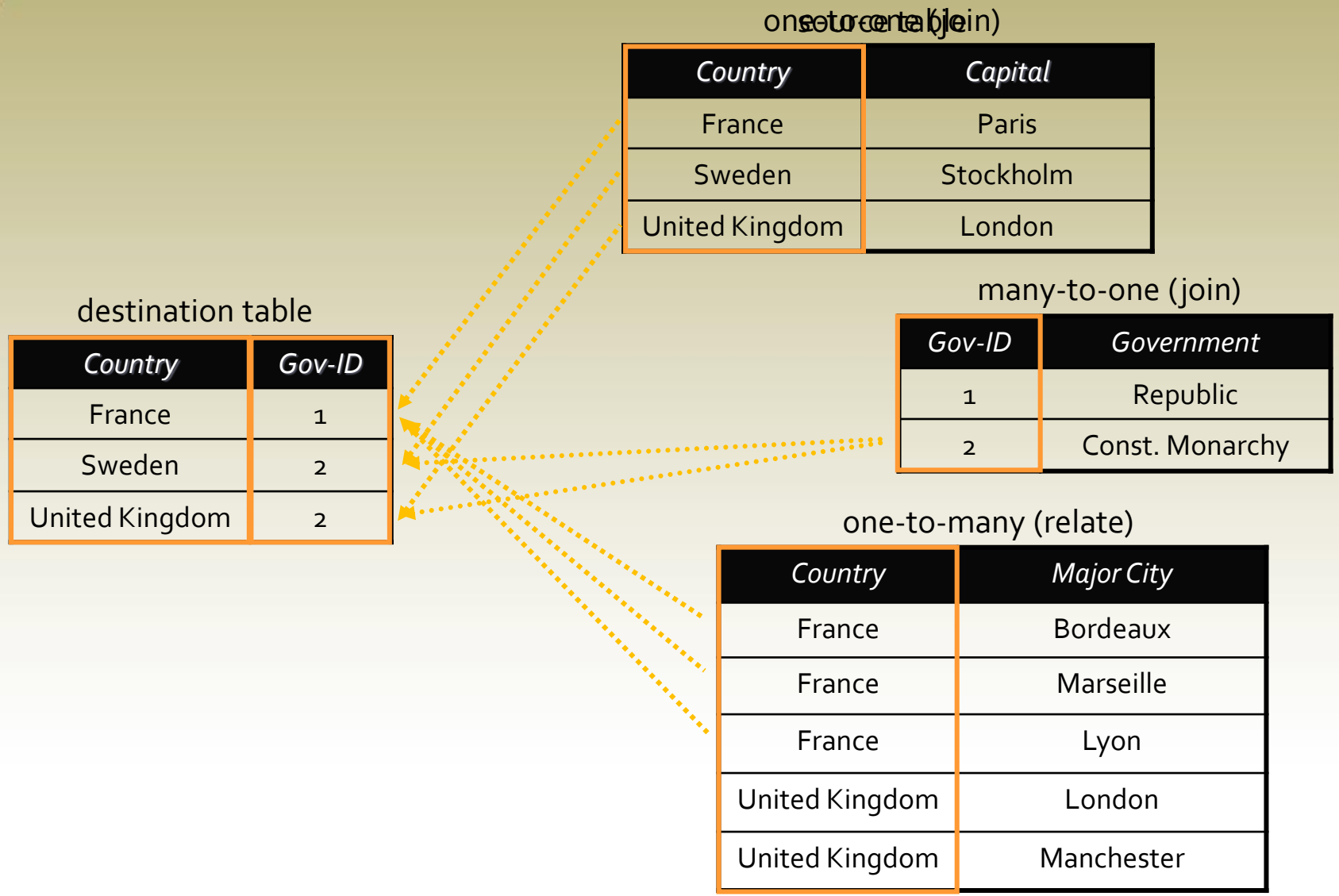
OBJECT ID#	Landuse Code	Join Table Landuse Code	Join Table Landuse Type
1	2	2	water
2	0	0	Unclassified
3	1	1	shrub

Cardinality of Relationships

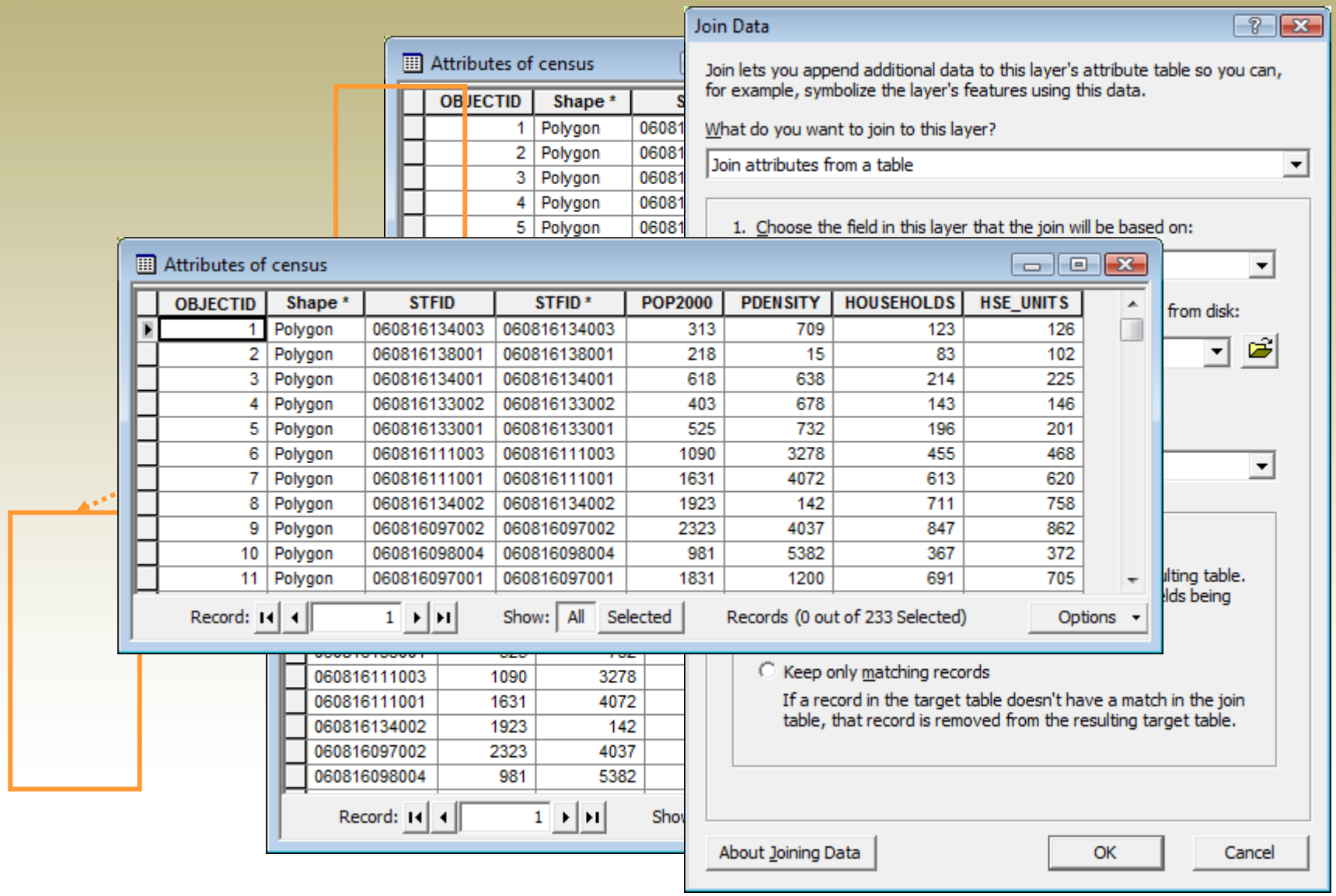


- ***One-to-one (join)***
  - every record in A matches exactly one record in B; every record in B matches exactly one record in A.
- ***Many-to-one (join)***
  - every record in table B matches one or more records in table A.
- ***One-to-many (relate)***
  - one or more records in table B matches exactly one record in table A.
- ***Many-to-many (relate)***
  - every record in A matches one or more records in B; every record in B matches one or more records in A.

Joining and Relating Tables



Joining Tables



Join Validation

Check for field names that **start** with an invalid character:

`~@#$%^&*()-+=|\\,<>?/{}.!'[]:;_0123456789

Check for field names that **contain** an invalid character:

`~@#$%^&*()-+=|\\,<>?/{}.!'[]:;

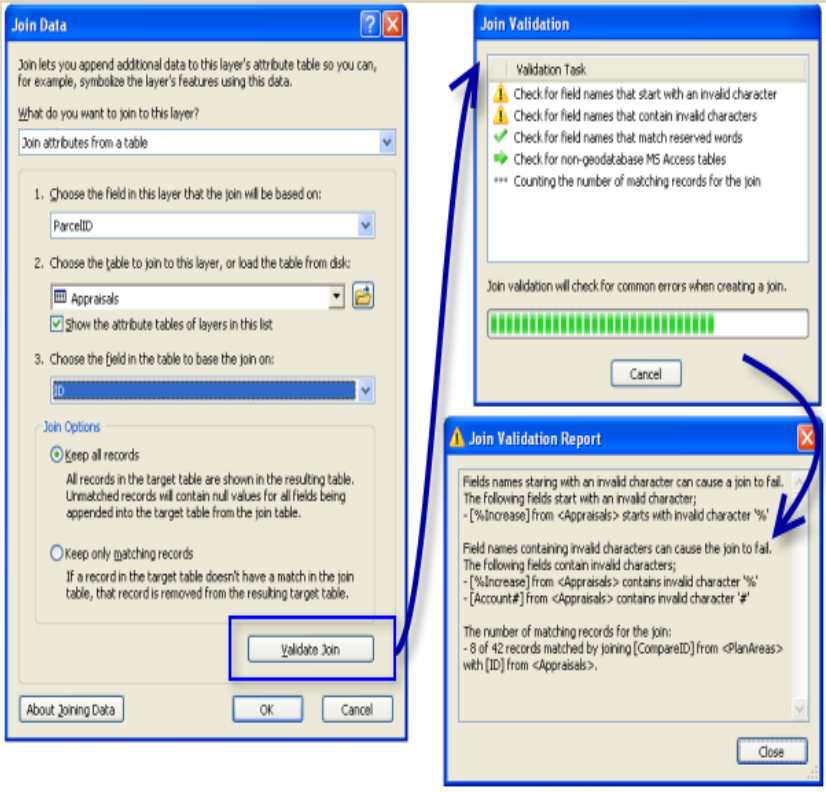
Check for field names that match:

"California" <> " California"
"California" <> "california"
1 <> "1"

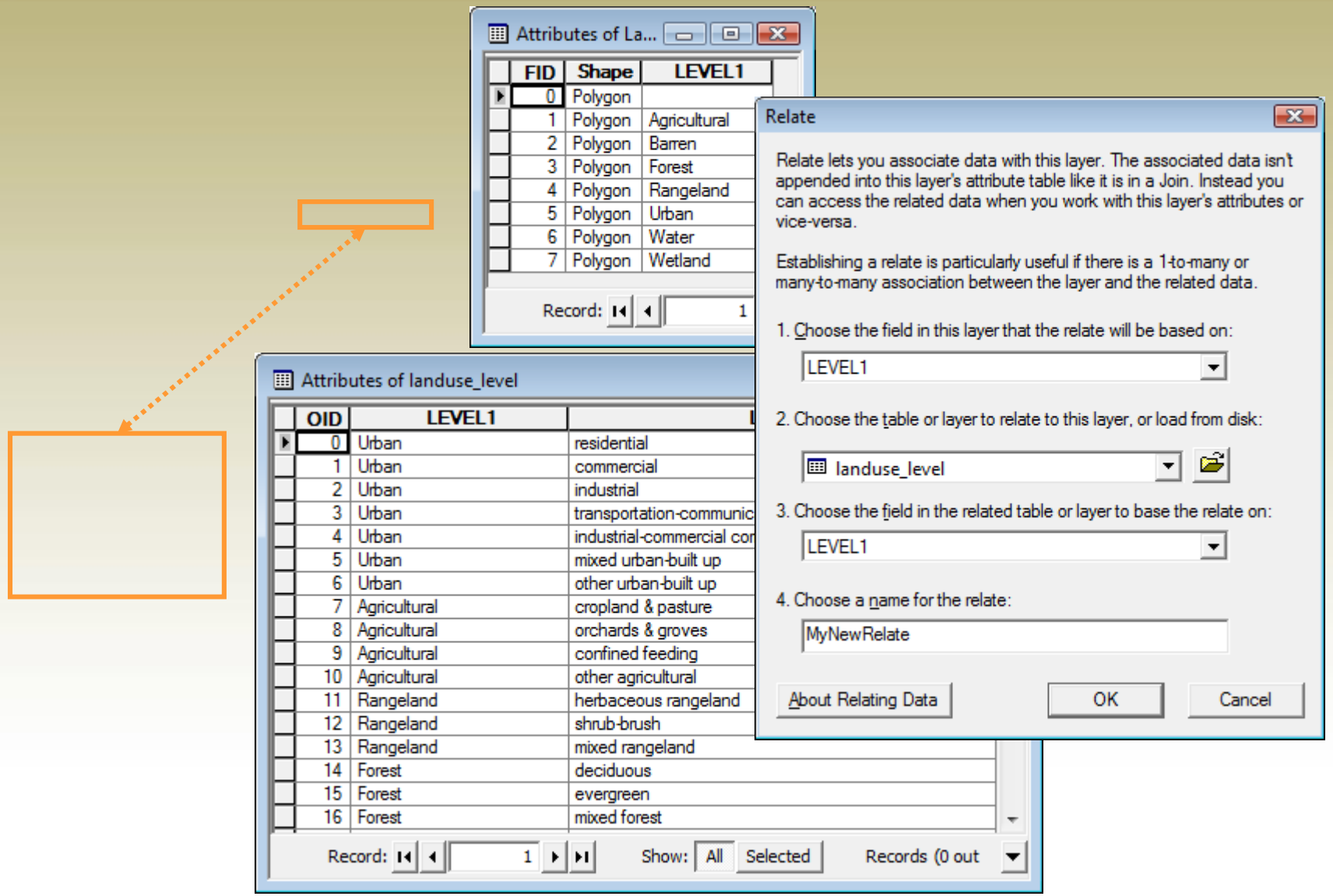
Check for field names in the table that Are Microsoft **Access reserved words**

date, day, month, table, text, user, when, where, year, and zone

Use **Validate Join** tool!



Relating Tables



Next workshop: Field Data Collection

THANK YOU!!

clemsongis.org