

---

# COMMON LAND UNIT BOUNDARIES

API Documentation 2020

---

[API Portal](#)

[GitHub Repo](#)

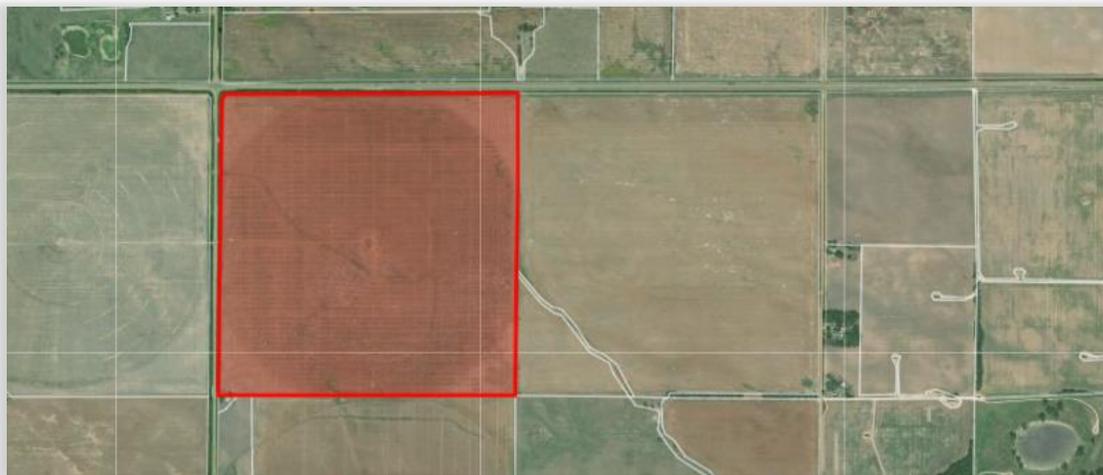
[Jupyter Notebook](#)

[Purchase](#)

## Service Overview

Common Land Unit (CLU) boundaries are farm or field boundaries used by the United States Department of Agriculture (USDA) in the delivery of certain farm programs. It is not uncommon for more than one crop to be grown on a CLU. These CLU boundaries are derived from the last publicly available USDA distribution from 2008. A single CLU is approximately interpreted as a "field". A Common Land Unit (CLU) is the smallest unit of land that has a permanent, contiguous boundary, a common land cover and land management, a common owner and a common producer in agricultural land associated with USDA farm programs.

The Ag-Analytics collection of the 2008 USDA Farm Service Agency (FSA) CLU Boundaries provide a service which a user can pass an extent (bounding box) and retrieve field boundaries in geojson or json format. To our knowledge, this is the only CLU field boundary data service in the market. It is a frequently requested dataset and useful for researchers who seek pre-made field boundaries in order to conduct representative analyses, as well as other apps that wish to serve 'starter' field boundaries.



*CLU Boundaries shown in FarmScope*

## POST Request

POST Request Example – application/x-www-form-urlencoded

```
where=1=1&objectIds=&time=&geometry={"xmin":-89.97802734375, "ymin":
39.07890809706475, "xmax":-89.9560546875, "ymax":39.095962936305476,
"spatialReference":{"wkid":4326}}&geometryType=esriGeometryEnvelope&inSR=4326&spat
ialRel=esriSpatialRelIntersects&distance=&units=esriSRUnit_Foot&relationParam=&out
Fields=*&returnGeometry=true&maxAllowableOffset=&geometryPrecision=6&outSR=4326&gd
bVersion=&returnDistinctValues=false&returnIdsOnly=false&returnCountOnly=false&ret
urnExtentOnly=false&orderByFields=&groupByFieldsForStatistics=&outStatistics=&retu
rnZ=false&returnM=false&multipatchOption=&resultOffset=&resultRecordCount=&f=pjson
```

## Header Parameters

Ocp-Apim-Subscription-Key: Subscription keys are given upon purchase - [Purchase APIs](#) 

## Request Parameters

Parameter	Information
<b>geometryType</b>	<p>The type of geometry specified by the geometry parameter. The geometry type can be an envelope, a point, a line, or a polygon. The default geometry type is an envelope.</p> <p>Values: esriGeometryPoint   esriGeometryMultipoint   esriGeometryPolyline   esriGeometryPolygon   esriGeometryEnvelope</p>
<b>inSR</b>	<p>The spatial reference of the input geometry. The spatial reference can be specified as either a well-known ID or as a spatial reference JSON object. If the inSR is not specified, the geometry is assumed to be in the spatial reference of the map.</p>
<b>distance</b>	<p>The buffer distance for the input geometries. The distance unit is specified by units. For example, if the distance is 100, the query geometry is a point, units is set to meters, and all points within 100 meters of the point are returned. The geodesic buffer is created based on the datum of the output spatial reference if it exists. If there is no output spatial reference, the input geometry spatial reference is used. Otherwise, the native layer spatial reference is used to generate the geometry buffer used in the query. This parameter only applies if supportsQueryWithDistance is true.</p>
<b>spatialRel</b>	<p>This parameter applies for hosted feature services running on a spatiotemporal datastore if sqlParserVersion contains ES. Point layers require either an envelope or polygon input geometry and supports the following values:</p> <p>Values: esriSpatialRelIntersects   esriSpatialRelEnvelopeIntersects   esriSpatialRelIndexIntersects   esriSpatialRelWithin</p> <p>Polygon or polyline layers support the following values:</p> <p>Values: esriSpatialRelIntersects   esriSpatialRelContains  </p>



	<p>esriSpatialRelEnvelopeIntersects   esriSpatialRelIndexIntersects   esriSpatialRelWithin</p>
<b>units</b>	<p>The unit for calculating the buffer distance. If unit is not specified, the default will be esriSRUnit_Foot when querying feature services in ArcGIS Enterprise, and esriSRUnit_Meter when querying feature services in ArcGIS Online This parameter only applies if supportsQueryWithDistance is true.</p> <p>Values: esriSRUnit_Meter   esriSRUnit_StatuteMile   esriSRUnit_Foot   esriSRUnit_Kilometer   esriSRUnit_NauticalMile   esriSRUnit_USNauticalMile</p>
<b>relationParam</b>	<p>The spatial relate function that can be applied while performing the query operation. An example for this spatial relate function is "FFFTT***". For more information on this spatial relate function, see the documentation for the spatial relate function.</p>
<b>outFields</b>	<p>The list of fields to be included in the returned result set. This list is a comma delimited list of field names. You can also specify the wildcard "*" as the value of this parameter. In this case, the query results include all the field values.</p> <p>Example</p> <pre>//Standard usage outFields=AREANAME,ST,POP2000  //Wildcard usage outFields=*</pre>
<b>returnGeometry</b>	<p>If true, the result includes the geometry associated with each feature returned. The default is true.</p> <p>Values: true   false</p>
<b>maxAllowableOffset</b>	<p>The maximum allowable offset, only applicable for layers that are not editable.</p>
<b>geometryPrecision</b>	<p>This option can be used to specify the number of decimal places in the response geometries returned by the Query operation. This applies to X and Y values only (not m or z-values).</p> <p>Example</p> <pre>geometryPrecision=3</pre>
<b>outSR</b>	<p>The spatial reference of the returned geometry. The spatial reference can be specified as either a well-known ID or as a spatial reference JSON object. If outSR is not specified, the geometry is returned in the spatial reference of the map.</p>



<p><b>gdbVersion</b></p>	<p>The geodatabase version to query. This parameter applies only if the <code>isDataVersioned</code> property of the layer is true. If this is not specified, the query will apply to the published map's version.</p> <p>Syntax: <code>gdbVersion=&lt;version&gt;</code></p> <p>Example: <code>gdbVersion=SDE.DEFAULT</code></p>
<p><b>returnDistinctValues</b></p>	<p>If true, it returns distinct values based on the fields specified in <code>outFields</code>. This parameter applies only if the <code>supportsAdvancedQueries</code> property of the layer is true. This parameter can be used with <code>returnCountOnly</code> to return the count of distinct values of subfields.</p> <p>Note: Make sure to set <code>returnGeometry</code> to false when <code>returnDistinctValues</code> is true. Otherwise, reliable results will not be returned.</p> <p>Values: true   false</p>
<p><b>returnIdsOnly</b></p>	<p>If true, the response only includes an array of object IDs. Otherwise, the response is a feature set. The default is false. When <code>objectIds</code> are specified, setting this parameter to true is invalid.</p> <p>While there is a limit to the number of features included in the feature set response, there is no limit to the number of object IDs returned in the ID array response. Clients can exploit this to get all the query conforming object IDs by specifying <code>returnIdsOnly=true</code> and subsequently requesting feature sets for subsets of object IDs.</p> <p>Values: true   false</p>
<p><b>returnCountOnly</b></p>	<p>If true, the response only includes the count (number of features/records) that would be returned by a query. Otherwise, the response is a feature set. The default is false. This option supersedes the <code>returnIdsOnly</code> parameter. If <code>returnCountOnly = true</code>, the response will return both the count and the extent. This parameter can be used with <code>returnDistinctValues</code> to return the count of distinct values of subfields.</p> <p>Values: true   false</p>



## POST Response

GET Response Example (Snippet) – application/json

```
{
  "objectIdFieldName": "OBJECTID",
  "globalIdFieldName": "",
  "geometryType": "esriGeometryPolygon",
  "spatialReference": {
    "wkid": 4326,
    "latestWkid": 4326
  },
  "fields": [
    {
      "name": "OBJECTID",
      "alias": "OBJECTID",
      "type": "esriFieldTypeOID"
    },
    {
      "name": "CALCACRES",
      "alias": "CALCACRES",
      "type": "esriFieldTypeDouble"
    },
    {
      "name": "CALCACRES2",
      "alias": "CALCACRES2",
      "type": "esriFieldTypeDouble"
    }
  ],
  "features": [
    {
      "attributes": {
        "OBJECTID": 5024814,
        "CALCACRES": 0.62,
        "CALCACRES2": null
      },
      "geometry": {
        "rings": [
          [
            [-89.972683, 39.093529],
            [-89.972834, 39.093529],
            [-89.974718, 39.094745],
            [-89.972683, 39.093529]
          ]
        ]
      }
    }
  ]
}
```

## Response Parameters

Parameter	Data Type	Description
<b>objectIdFieldName</b>	String	Gets the name of the object ID field.
<b>globalIdFieldName</b>	String	Gets or sets the global ID field name.
<b>geometryType</b>	String	The <a href="#">esriGeometryType Constant</a>
<b>spatialReference</b>	List	Defines the spatial reference of a map, layer, or task parameters.
<b>fields</b>	List	Each field stores a specific type of data, such as a number, date, or piece of text.
<b>features</b>	List	<p><a href="#">Features</a> are used to associate tabular data with geographic information. Features have two important top-level properties:</p> <p><b>geometry</b> - A point, polyline, polygon or extent object.</p> <p><b>attributes</b> - An object of key/value pairs in JSON format to associate with the geometry.</p>



## GET Request

GET Request Example – application/json

```
{
  "geometry" : {"xmin": -88, "ymin": 40.2, "xmax": 88.126, "ymax": 40.22,
  "spatialReference": {"wkid": 4326}},
  "f" : "geojsonjson"
}
```

## Header Parameters

Ocp-Apim-Subscription-Key: Subscription keys are given upon purchase - [Purchase APIs](#) 

## Request Parameters

Parameter	Data Type	Required?	Default	Options	Description
<b>geometry</b>	string	Yes	--	Spatial Coordinates (xmin, ymin, xmax, ymax, lat/long coordinates & wkid spatialReference)	The geometry to apply as the spatial filter. The structure of the geometry is the same as the structure of the json/geojson
<b>f</b>	string	Yes	--	geojson/json format	Format of returned data



## GET Response

GET Response Example (Snippet) – application/json

```
{
  "type": "FeatureCollection",
  "crs": {"type": "name", "properties": {"name": "EPSG:4326"}},
  "features": [{"type": "Feature", "id": 4166436, "geometry": {"type": "Polygon", "coordinates": [[[-88.00395, 40.197919], [-88.003955, 40.200814], [-88.007578, 40.20089], [-88.008708, 40.200302], [-88.009195, 40.199943], [-88.009672, 40.199352], [-88.009672, 40.197926], [-88.00395, 40.197919]]]}, "properties": {"OBJECTID": 4166436, "CALCACRES": 36.27999878, "CALCACRES2": null}}, {"type": "Feature", "id": 4167422, "geometry": {"type": "Polygon", "coordinates": [[[-88.01045, 40.201013], [-88.010456, 40.201441], [-88.010784, 40.201438], [-88.010745, 40.201711], [-88.011059, 40.201708], [-88.011033, 40.202017], [-88.012063, 40.20202], [-88.012058, 40.201654], [-88.012084, 40.201425], [-88.012353, 40.201422], [-88.01235, 40.201216], [-88.012337, 40.201028], [-88.010554, 40.201017], [-88.01045, 40.201013]]]}, "properties": {"OBJECTID": 4167422, "CALCACRES": 3.3900001, "CALCACRES2": null}}, {"type": "Feature", "id": 4168358, "geometry": {"type": "Polygon", "coordinates": [[[-88.076322, 40.215273], [-88.078181, 40.215265], [-88.078091, 40.208126], [-88.076313, 40.208129], [-88.076286, 40.213926], [-88.076303, 40.214416], [-88.076322, 40.215273]]]}, "properties": {"OBJECTID": 4168358, "CALCACRES": 30.600000380000001, "CALCACRES2": null}}
}
```

## Response Parameters

Parameter	Data Type	Description
<b>type</b>	String (GeoJSON Object)	Describes the geojson object type (Point, Feature, FeatureCollection)
<b>crs</b>	List	Coordinate reference system (EPSG codes)
<b>features</b>	List	Container for all the features of the raster
<b>features.id</b>	number	Feature object common identifier
<b>features.geometry</b>	GeoJSON feature object	Represents spatially bounded points, curves, and surfaces in coordinate space (Polygon, Point, etc)
<b>features.properties</b>	List	List of properties assigned to the spatially bounded feature (Here OBJECTID is the common identifier, CALCACRES is the acreage)



## Citations:

- [USDA Common Land Unit Boundaries](#)
- [GeoJSON Specification](#)
- [ArcGIS Features and Geometries](#)
- [ArcGIS REST API Query Information](#)
- [ArcGIS Esri Geometry Types](#)
- Spatial Reference Information: World Geodetic System (WGS 84) - National Geospatial-Intelligence Agency – 1984



Please contact [support@analytics.ag](mailto:support@analytics.ag) or [josh@ag-analytics.org](mailto:josh@ag-analytics.org) with any comments or questions.

[Terms of Use and Privacy](#)

