Landscape Change

Geoprocessing Tools

# Introduction

The purpose of the Landscape Change geoprocessing tools is to automate portions of the protocol workflow. The toolbox containing the geoprocessing tools, associated Python scripts, and documentation are on GitHub at <https://github.com/NPS-NCCN/Landscape_GeoTools>.

The *docs* folder contains the README.md file for the repository and the Microsoft Word and PDF versions of the documentation. The *toolbox* folder contains the toolbox, *LandscapeChange.atbx*, and all Python scripts.

* **Add Attributes to Patches**

Adds attributes/predictors to the LandTrendr output patches for data loading into the SQL Server Database and Google Earth Engine (GEE).

* **Add Attributes to Select Patches & Export CSV**

**Used to “recalculate” attributes for select patches (split or merged patches), and exports the patches as a CSV for database data loading.**

* **Join Labels to Patches**

Joins labels exported from the database to the patches feature class, or if the event fields already exist, updates the event fields with the labels.

* **Export Patches**

Performs validation checks on patch attributes and event attributes. When exporting patches to a CSV, the data is also formatted to use with the applicable SSIS packages for database data loading. When exporting for GEE, a shapefile is exported.

## Setting Up the Toolbox

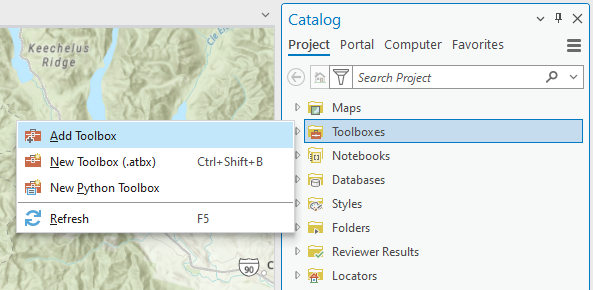
The GitHub repository containing the toolbox and Python scripts can either be downloaded or cloned. It is recommended to save these files locally on your computer and not on a network drive.

If you have Git installed on your computer you can clone the repository by opening the Command Prompt, changing the directory to the desired folder, and running the following command:

git clone https://github.com/NPS-NCCN/Landscape\_GeoTools.git

If you do not have Git installed on your computer or prefer to download the repository instead, go to <https://github.com/NPS-NCCN/Landscape_GeoTools>, click the green Code button, and select Download Zip at the bottom. Extract the files from the zip folder.

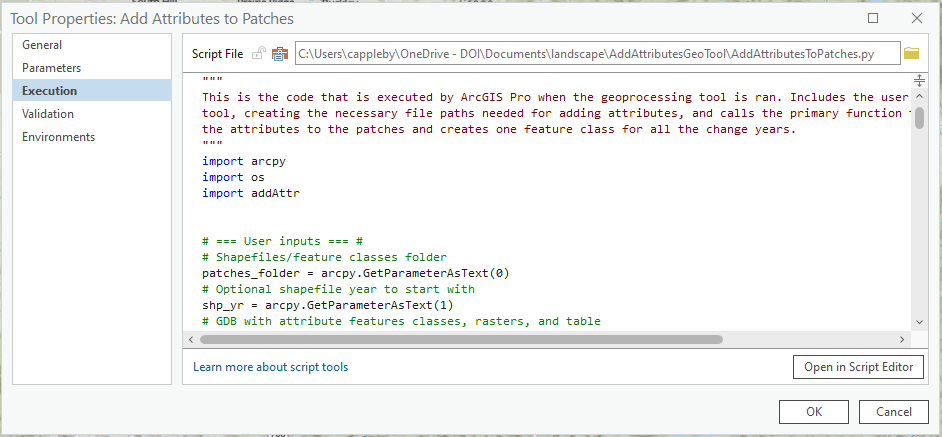
The *.idea* and *docs* folders are not needed to set up the toolbox and can be deleted, if desired. To set up the toolbox, open ArcGIS Pro. In the Catalog pane, right-click on Toolboxes and select Add Toolbox. Navigate to where you saved the toolbox (inside the toolbox folder), select it, and click OK.



When you expand the toolbox in the Catalog pane, you should see the five geoprocessing tools. Right-click on one of the tools and select Properties. Select Execution and click on the folder icon for Script File. Navigate to the folder containing the Python scripts and select the script with the same name as the geoprocessing tool (in parentheses below) and click OK. This points the tool to the code that it needs to run.

The additional Python scripts contain additional code necessary to run the tool and must remain in the same folder as the selected script. Double-click the geoprocessing tool to open it.

* **Add Attributes to Patches (*AddAttributesToPatches.py)***
  + *addAttr.py*
  + *addAttrFunctions.py*
  + *addAttrUtils.py*
  + *eventsFunctions.py*
* **Add Attributes to Select Patches & Export CSV (*AddAttrSelectPatchesExportCSV.py)***
  + *addAttr.py*
  + *addAttrFunctions.py*
  + *addAttrUtils.py*
  + *expPatchesFunctions.py*
  + *eventsFunctions.py*
* **Join Labels to Patches (*JoinLabelsToPatches.py)***
  + *eventsFunctions.py*
* **Export Patches (*ExportPatches.py)***
  + *expPatchesFunctions.py*



# Add Attributes to Patches

Adds attributes/predictors to the LandTrendr output patches for each individual year and creates an output feature class containing all years. The original shapefiles are not modified.

The rasters, feature classes, and vegetation type table needed to add the attributes must be in a park-specific geodatabase. The following attributes are added: coordinates for the patch centroid (Albers, UTM, DD), wilderness name, land management, watershed, whether the patch centroid is in the park, buffer, mask, and protected area, vegetation value and code, mean slope and elevation, aspect category, zonal geometry (thickness, major and minor axis, orientation), and whether the patch overlaps a patch from the previous year.

Additionally, the fields Park and PatchName are generated. There is also the option to add the label "Annual Variability" to patches that fall completely within the elevation or water mask. The field names for the attributes match those needed to use the Landscape Change geoprocessing tool Export Patches to CSV.

## Parameters

In addition to the parameter descriptions below, similar information is available by hovering over the info icon next to each parameter in the geoprocessing tool window.

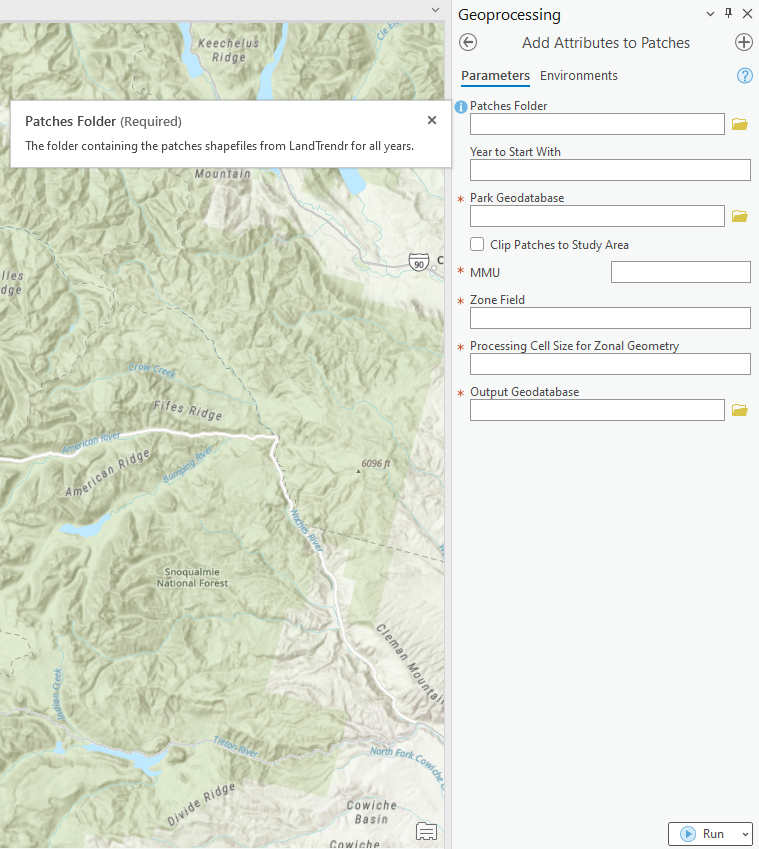
**Patches Folder**

**This is the folder containing the patches shapefiles from LandTrendr for all the years of the particular run. The original shapefiles are not modified.**

**Resume Previous Run (optional)**

Occasionally, the geoprocessing tool will unexpectedly fail on an error that is not persistent. If this happens, the tool can be run again without reprocessing the previous years’ shapefiles by checking this box.

The tool will automatically look for the patches feature classes in the default geodatabase and the output feature class with patches from all years in the output geodatabase. Processing will begin on the year the error occurred during the previous tool run. If the patches feature classes are not in the default geodatabase or the name has been changed for the output feature class with patches from all years, the tool will fail. If this parameter is checked, the Year to Resume With parameter cannot be used.



**Year to Resume With (optional)**

Occasionally, the geoprocessing tool will unexpectedly fail on an error that is not persistent. If this happens, the tool can be run again starting with the year where the error occurred without having to reprocess previous years’ shapefiles. For example, if the tool fails on the shapefile for 2012, enter “2012” for this parameter, and the processing would begin with the shapefile for that year.

This method can be used instead of the Resume Previous Run parameter if the in-process shapefiles from the failed run have been deleted from the default geodatabase. If the tool cannot find the shapefile for the year specified or the name has been changed for the output feature class with patches from all years, the tool will fail. If a value is entered for this parameter, the Resume Previous Run parameter cannot be used.

Park Geodatabase

This is the park-specific geodatabase containing the data required for the geoprocessing tool to add certain attributes. The name of the entities must be the same as the list below with the park code and an underscore as a prefix (*MORA\_buff\_fc*). Entities with an asterisk (\*) are not required for LEWI. The geodatabase name needs to be the park code as the park code is used by the geoprocessing for various functions (*MORA.gdb*).

* *aspect\_rst* - raster with aspect classes, 1 to 9, representing the aspect designation, used to calculate *Aspect*
* *buff\_fc* - feature class of the 300-meter buffer (2-kilometer for LEWI), used to calculate *InBuff*
* *dem\_rst* - raster of a seamless digital elevation model, used to calculate *ElevMean*
* *east\_west\_fc\** - feature class of study area split into two polygons by the crest
* *land\_mgmt\_wild\_fc* - feature class with land management and wilderness names, used to calculate *LandMgmt* and *Wildname*
* *mask\_fc\** - feature class of the elevation mask, used to calculate *InMask*
* *park\_bndry\_fc* - feature class of the park boundary, used to calculate *InPark*
* *protected\_fc\** - feature class of the protected areas, used to calculate *Protected*
* *slope\_rst* - raster of the slope in degrees, used to calculate *SlopeMean*
* *study\_area\_fc* - feature class of the study area, used to clip patches, if applicable
* *veg\_rst* - raster of the vegetation types with the MCIDs specific to the park, used to find the majority vegetation type for *VegValue*
* *veg\_type\_tbl­\** - table containing the MCIDs and associated vegetation codes specific to the park, used to populate *VegCode* using the *VegValue*
* *water\_fc* - feature class of the water mask, used to add “Annual Variability” label
* *watershed\_fc* - feature class of the HUC12 watersheds with names, used to calculate *Watershed*

Clip Patches to Study Area (optional)

When checked, the patches will be clipped to the study area, and patches with an area less than the Minimum Mapping Unit will be deleted. Additionally, the *area* and *perim* fields are update with the new values for the clipped patches using the rounded values in *Shape\_Area* and *Shape\_Length*. When unchecked, the patches will be processed as-is.

MMU

The minimum mapping unit is the minimum number of pixels required for a patch to be analyzed and should be the same MMU used for the LandTrendr run. This input is used if the patches are clipped, and to create the *PatchName*.

Zone Field

The patches field containing the unique identifier. This is used by the tool for various joins and as an input for Zonal Geometry as Table.

Processing Cell Size

This is the processing cell size that will be used as an input for Zonal Geometry as Table.

Output Geodatabase

This is where the output feature class with patches from all years will be saved. A geodatabase should have been created for each LandTrendr run, and this geodatabase should be selected as the output.

Add Event Fields and Mask Labels

When checked, all event fields (EventType, ChangeType, Confidence, AltType, ChangeDesc, EventDate, LabeledBy, PriorRun, PostDist, DistName, DistYear, and Split) will be added to the patches feature class. Additionally, labels will be assigned to patches that are completely within the elevation or water/lakes masks.

## Processing Steps

The geoprocessing tool loops over the same process for each shapefile (one shapefile per year) in the patches folder, using data from the applicable park geodatabase. The patches shapefile is converted to a feature class in the default geodatabase. If any of the fields that are to be added already exist, those fields are deleted. If the Clip Patches to Study Area parameter is checked, the processes described for that parameter will be executed.

Albers, UTM Zone 10N, and NAD 83 decimal degrees coordinates for the patch central point are added. If the central point cannot be calculated, the centroid is calculated instead. The field *CoordType* records which calculation was used. The Albers coordinates are then used to create a point feature class in the default geodatabase, and several attributes are added to the patches using the points.

The *MANAGER* and *WildName* fields from the *land\_mgmt\_wild\_fc* feature class are added to the points using a spatial join. The *MANAGER* field is renamed *LandMgmt*. The *NAME* field from *watershed\_fc* is also added to the points using a spatial join, and the *NAME* field is renamed *Watershed*.

The fields *InPark, InBuffer,* and *Protected* are calculated by selecting the points by locations using the appropriate feature class (*park\_bndry\_fc, buff\_fc,* and *protected\_fc*, respectively), calculating 1 (True) for those points, inverting the selection, and calculating 0 (False) for those points. The field *EastWest* is added to the points using a spatial join with *east\_west\_fc*. *Protected* and *EastWest* are all null for LEWI. The fields *WildName, LandMgmt, Watershed, InPark, InBuffer,* *Protected*, and *EastWest* fields from the points are joined to the patches polygon feature class on the Zone Field parameter field. Then the point feature class is deleted. The remaining attributes are calculated using the patches feature class.

*InMask* is calculated by selecting patches that completely within *mask\_fc*, calculating 1 (True) for those patches, inverting the selection, and calculating 0 (False) for those patches.

After ensuring the *veg\_rst­ ­*is the same projection as the patches, Zonal Statistics as Table is used to find majority vegetation value in each patch using the Zone Field parameter as the Zone Field for the tool. The value field from the table is joined to the patches, the field is renamed *Veg\_value\_text*, and the zonal statistics table is deleted, as well as the projected raster, if applicable. The *VegValue* field is calculated by converting *Veg\_value\_text* from a Text data type to a Double data type, and *Veg\_value\_text* is deleted. The *CODE* from *veg\_type\_tbl* is joined to the patches on the *VegValue* (*MCID* in the table), the *CODE* field is renamed *VegCode*. For LEWI, *veg\_type\_tbl* is not used; *VegCode* is calculated by adding “L” to the *VegValue* (*VegValue* of 1 becomes L01, *VegValue* of 23 becomes L23, etc.).

Zonal Statistics as Table is further used to calculate three additional fields. *ElevMean* and *SlopeMean* use *dem\_rst* and *slope­\_rst*, respectively, to find the mean elevation and slope for each patch. For *Aspect*, the majority value from *aspect\_rst* is used.

Zonal Geometry as Table is used to calculate *THICKNESS, MAJORAXIS, MINORAXIS,* and *ORIENTATION* using the user inputs Zone Field and Processing Cell Size as inputs to the tool. The fields are then joined to the patches feature class, and the zonal geometry table is deleted. The field *paratio* is then calculated using the following equation (the double asterisks \*\* is used for exponents in Python):

Starting with the second patches feature class, *OverlapPrv* is calculated, initially, calculating 0 (False) for all patches. The patches feature class from the previous year is saved in the default geodatabase and used to check if any patches for the current year overlap. This is done by selecting current year patches by location that intersect with the previous year patches and calculating 1 (True) for those patches. Then the patches feature class from the previous year is deleted. The current year is saved to be used as the previous year for the next year’s patches.

If the Add Event Fields and Mask Labels parameter is checked, the event fields listed in the parameter description are added the patches feature class. To assign labels to patches in the elevation mask, patches whose *InMask = 1* are selected (not applicable to LEWI); to assign labels to patches in the lakes/water mask, all patches that are completely within *lakes\_water\_fc* are selected. The following event fields are populated for the selected patches:

* EventType: Mask
* ChangeType: Annual Variability
* Confidence: 2
* EventDate: date the tool is ran (today)
* LabeledBy: Geoprocessing
* PriorRun: 0
* PostDist: 0

For the first patches feature class processed, a new feature class is created named *PARK\_changeDB* (*MORA\_changeDB*) in the geodatabase specified in the Output Geodatabase parameter. The park code is derived from the park-specific geodatabase name. For subsequent patches feature classes, their data are appended to *PARK\_changeDB*. Additionally, since *OverlapPrv* cannot be calculated for the first year of patches, the field populated with 0 (False) for all patches.

Once all shapefiles/years have been processed, the remaining patches feature class is deleted. Next, the *Park* field is created and populated with the park code. Additionally, the *PatchName* field is created and calculated using the park code, Minimum Mapping Unit parameter, *index* field value, minimum and maximum *yod*, *yod* field value, and *annualID* field value, with each separated by an underscore (*MORA\_5\_NBR\_1987\_2023\_2012\_835*). The minimum and maximum year values in the *yod* field are found, and the final patches feature class is renamed *PARK\_changeDB\_MinYr\_MaxYr* (*MORA\_changeDB\_1987\_2023*).

# Add Attributes to Select Patches & Export CSV

Updates attributes/predictors for selected patches that have split or merged by copying the patches, deleting the attributes, recalculating the attributes, deleting the selected patches from the feature class, and copying the updated patches to the feature class. The rasters, feature classes, and vegetation type table needed to add the attributes must be in a park-specific geodatabase.

The following attributes are added: coordinates for the patch centroid (Albers, UTM, DD), wilderness name, land management, watershed, whether the patch centroid is in the park, buffer, and protected area, whether the patch is completely within the elevation mask (if applicable), vegetation value and code, mean slope and elevation, aspect category, and zonal geometry (thickness, major and minor axis, orientation), and paratio. The patches are then exported to a CSV for loading into the Landscape Change database.

## Parameters

In addition to the parameter descriptions below, similar information is available by hovering over the info icon next to each parameter in the geoprocessing tool window.

Patches

The feature class containing the selected patches/features. If no patches/features are selected, all patches in the feature class will be processed.

Park Geodatabase

See parameter description for Add Attributes to Patches.

Zone Field

See parameter description for Add Attributes to Patches.

Processing Cell Size for Zonal Geometry

See parameter description for Add Attributes to Patches.

Output Folder (for CSV)

The folder where the CSV for the selected patches will be saved.

Output CSV File Name

The filename for the CSV for the selected patches.

## Processing Steps

The selected patches are copied to a new feature class in the default geodatabase, then the selected patches in the original feature class are deleted. The remaining steps are the same as Add Attributes to Patches with exceptions. *OverlapPrv*, *Park*, and *PatchName* are not calculated because it is assumed that the patches have already been processed with Add Attributes to Patches. Additionally, since the selected patches are likely split or merged patches, *PatchName* should have already been updated by the user.

An additional step is performed when adding attributes to select patches - the *area* and *perim* fields are updated with the *Shape\_Area* and *Shape\_Length* values, rounded to the nearest integer. Then the *paratio­* field is calculated using the equation shown in Add Attributes to Patches. Once the attributes have been added to the copied patches, the patches are then appended to the original feature class. The copied patches are exported to CSV using the same steps as Export Patches to CSV and copied patches feature class is deleted from the default geodatabase.

# Export Patches

Depending on the user selections, this tool either exports a CSV and/or shapefile. Exports only required fields from the patches feature class to a CSV to be used for importing patches and prior events data into the Landscape Change database. Exports only required fields from the patches feature class to a shapefile to be used in Google Earth Engine. Validation checks are performed, and the patches will only be exported if the validation checks pass.

## Parameters

In addition to the parameter descriptions below, similar information is available by hovering over the info icon next to each parameter in the geoprocessing tool window.

Run Geodatabase

This is where the output feature class with patches from all years were saved for all parks, which should have been created for the particular LandTrendr run.

Feature Classes to Export (optional)

If the user does not want to export patches for all the feature classes in the run geodatabase, the desired feature classes to export can be selected.

Export a CSV (optional)

When checked, the patches will be exported as a CSV. All patches are exported.

CSV Output Folder

The folder where the CSVs for all the parks (or the selected feature classes) will be saved. If Export a CSV parameter is checked, a folder must be specified. The default is folder is R:\Projects\LPa01\_Landscape\_Change\Data\Raw. If the run geodatabase ends in the years for the runs, the tool will automatically look for a folder with that name. Ex: If the run geodatabase name is Patches\_1987\_2023, the tool will automatically look for a folder in R:\Projects\LPa01\_Landscape\_Change\Data\Raw named 1987\_2023 and make that the default.

Also, the tool will automatically look for a folder with the same name as the park code at the beginning of the feature class it is processing and save the CSV to that folder if it exists. Otherwise, the CSV is saved to the output folder. Ex: If the feature class is MORA\_changeDB\_1987\_2023, the tool will automatically look for a folder named MORA in the output folder.

Export for GEE (optional)

When checked, the patches will be exported as a shapefile. Only patches with Events that do not have an EventType of Mask or Model are exported.

GEE Output Folder

The folder where the shapefiles for all the parks (or the selected feature classes) will be saved. If Export for GEE is checked, a folder must be specified. The default is folder is R:\Projects\LPa01\_Landscape\_Change\Spatial\_info\GIS\_data. If the run geodatabase ends in the years for the runs, the tool will automatically look for a folder with that name. Ex: If the run geodatabase name is Patches\_1987\_2023, the tool will automatically look for a folder in R:\Projects\LPa01\_Landscape\_Change\Spatial\_info\GIS\_data named 1987\_2023 and make that the default. Also, each shapefile is exported into its own folder inside the output folder.

## Processing Steps

The geoprocessing tool loops over each feature class in the run geodatabase or the selected feature classes. If exporting a CSV, the tool extracts the park code from the feature class name and looks for a folder with that name in the CSV output folder. If a folder with that park code does exist, this is where the CSV will be saved. The CSV file is name *PARK\_patches\_YYYY\_YYYY.csv* (ex: *MORA\_patches\_1987\_2023.csv*)*.* If that filename already exists, “*\_1*” is added to the filename, and this check continues until the filename is unique (ex:  *MORA\_patches\_1987\_2023\_1.csv*, and if this filename exists, the tool tries *MORA\_patches\_1987\_2023\_1\_1.csv*, etc.).

There are three primary validation checks that are performed on the patches whether they are being exported as a CSV or shapefile. To perform the validation checks, the patches data are read into a pandas data frame. Only the fields required for exported are extracted, and any fields not found in the feature class are displayed as a warning to inform the user. If the patches are being exported to a CSV, the data frame is also cleaned to ensure the values will import into the database using the SSIS package. If the patches are being exported to a shapefile, only patches with events that do not have an EventType of “Mask” or “Model” are validated.

The first validation check ensures all ChangeTypes match the ChangeTypes in the database lookup table. For any mismatches, the PatchName and ChangeType are displayed as a message. The second validation checks that Confidence values are either 1, 2, or 3 for patches with events. Additionally, if the patches are being exported to a shapefile, the tool also checks that all events with a Confidence of 1 or 2 also have an AltType. The PatchName of any patches with invalid confidence values are displayed as a message. The third validation check ensures there are no duplicate PatchNames, and any duplicate PatchNames are displayed as a message. When exporting a shapefile, the tool also checks that all applicable fields have values. Any fields with missing values are displayed as a message. If any of the validation checks are not passed, the patches will not be exported.

If patches are being exported to a CSV and passed the validation checks, the pandas data frame is saved as a CSV. When exporting patches to a shapefile, the pandas frame is no longer used, and the feature class is used for the remaining steps.

A feature layer is created containing only patches with events that do not have an EventType of “Mask” or “Model”. The feature layer is then copied to a temporary feature class in preparation for shapefile export.

Using the Longitude and Latitude fields, a point feature class is created. The Longitude and Latitude in WGS 84 are then added to the points using Calculate Geometry Attributes, and the WGS 84 values are transposed into the existing Longitude and Latitude fields. Next, the Longitude and Latitude fields are deleted from the temporary feature class, and the Longitude and Latitude fields from the point feature class are joined. Lastly, a datum field, *LLDatum*, is added and populated, and the point feature class is deleted.

A folder is created for each shapefile export, and the name of the folder and shapefile is the patches feature class name with “*\_GEE”* added to the end (ex: *MORA\_chanageDB\_1987\_2023\_GEE*). As with the CSV filename, if the folder already exists in the GEE output folder, the name is appended with “*\_1*”. The temporary feature class is exported as a shapefile to created folder, and the temporary feature class is deleted.

# Join Labels to Patches

Joins Event labels (EventType, ChangeType, Confidence, AltType, ChangeDesc, DistYear, DistName) exported from the SQL Server database to the patches feature class if the Event fields do not exist. Otherwise, the existing Event fields are updated with labels. This tool is useful when the Event fields already exist in the feature class.

## Parameters

Patches Feature Class

The feature class containing the patches you wish to join the labels to. This should be a feature class from the run geodatabase.

Create Backup of Run Geodatabase (optional)

When checked, a backup of the run geodatabase will be made and saved to the folder specified in the Folder for Backup Geodatabase parameter.

Folder for Backup Geodatabase

The folder path to save the backup of the run geodatabase. If the Create Backup of Run Geodatabase parameter is checked, a folder must be specified.

Events Table with Labels

The table containing the labels (EventType, ChangeType, Confidence, AltType, ChangeDesc, DistYear, DistName) to be joined to the patches or to update existing Event fields.

## Processing Steps

If the Create Backup of Run Geodatabase parameter is checked, the tool extracts the geodatabase name from the feature class path. It then creates a name for the backup geodatabase with today’s date (ex: if the name of the run geodatabase is *Landscape\_Change\_1987\_2023*, the backup geodatabase name would be *Landscape\_Change\_1987\_2023\_backup\_20241113*). Then the backup geodatabase is saved to the folder specified in the Folder for Backup Geodatabase parameter.

The tool checks the patches feature class for a field named *EventType*. If this field does not exist, it is assumed that none of the event fields exist. The event fields from the Events Table with Labels are joined to the patches feature class regardless of whether the patches had existing events fields. If the patches did have existing event fields, the tool loops through the existing and joined event fields and updates the existing fields with the values from the joined fields. The joined fields are then deleted.