

# Aplikace pro automatizované zpracování dat DPZ a vyhodnocení hydrologických poměrů

## Manuál

V tomto dokumentu je na konkrétních datech ilustrováno použití aplikace krok za krokem. Všechny dostupné možnosti při volání skriptů jsou pak popsány v Uživatelské dokumentaci.

1. Stáhneme data z družice Sentinel-2 z oblasti Krkonoš ze srpna 2018 a ze srpna 2024 s pokrytím oblačnosti do 20 %. Spustíme z příkazového řádku následující dva příkazy:

```
01_downloader.py -area "15.6247100,50.6852986,15.7927664,50.7487722" -start 20180801 -end 20180831 -max_cloud 20 -out_dir S:\images -work_dir S:\temp -user myuser -password mypass
```

a

```
01_downloader.py -area "15.6247100,50.6852986,15.7927664,50.7487722" -start 20240801 -end 20240831 -max_cloud 20 -out_dir S:\images -work_dir S:\temp -user myuser -password mypass
```

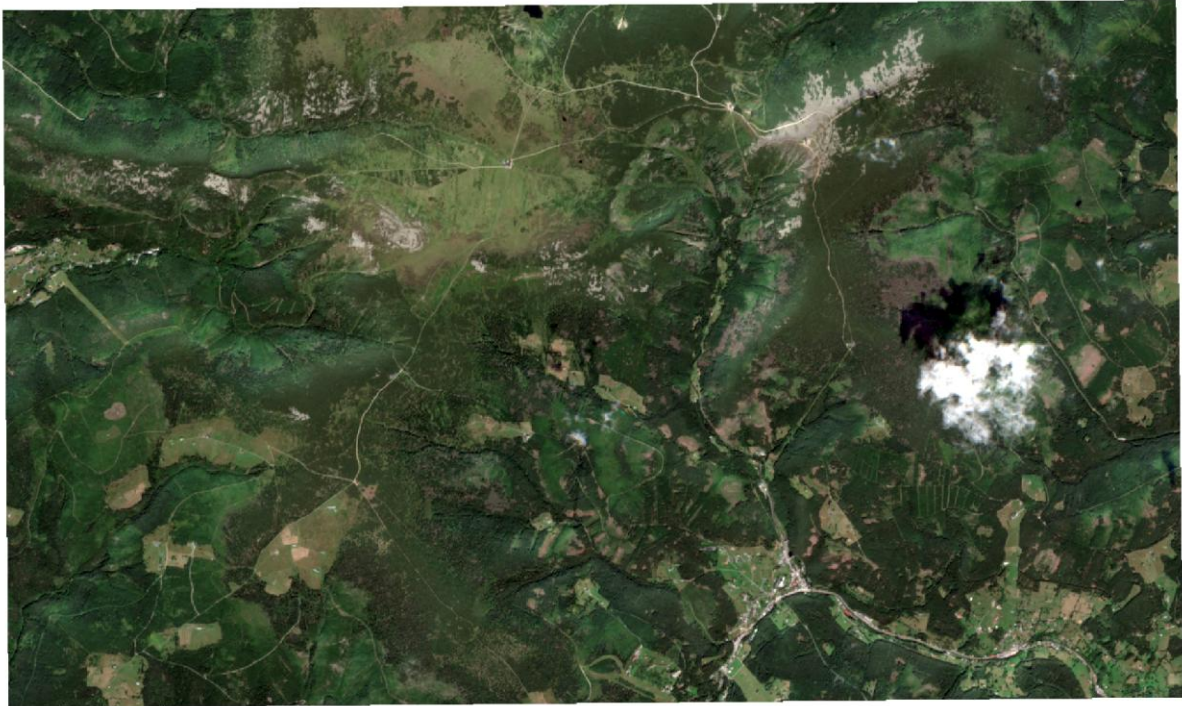
Před spuštěním skriptu je potřeba vytvořit příslušné složky pro uložení výsledků (v uvedeném příkladu S:\images a pro práci s dočasnými soubory S:\temp. Dále je nutné mít účet nebo se zaregistrovat a aktivovat si účet pro Copernicus Data Space Ecosystem na <https://documentation.dataspace.copernicus.eu/Registration.html>. Uživatelské jméno a heslo se zadává jako parametr skriptu místo řetězců myuser a mypass.

V závislosti na rychlosti připojení skript po několika minutách až desítkách minut stáhne snímky z dané oblasti, uloží je do zadané složky a ořízne je na zájmovou oblast definovanou zeměpisnými souřadnicemi.

Po doběhnutí skriptu nalezneme ve složce S:\images celkem šest stažených snímků, čtyři ze srpna 2018 a dva ze srpna 2024. Výsledné soubory, se kterými budeme dále pracovat, mají konec názvu zakončený na \_clip. Začátek názvu odpovídá konvencím pro pojmenování dat na Copernicus Data Space Ecosystem. Příklad názvu výsledného souboru (snímek byl pořízen 30. 8. 2024 10:05:59):

```
S2B_MSIL2A_20240830T100559_N0511_R022_T33UWS_20240830T134009.SAFE_clip.tif
```

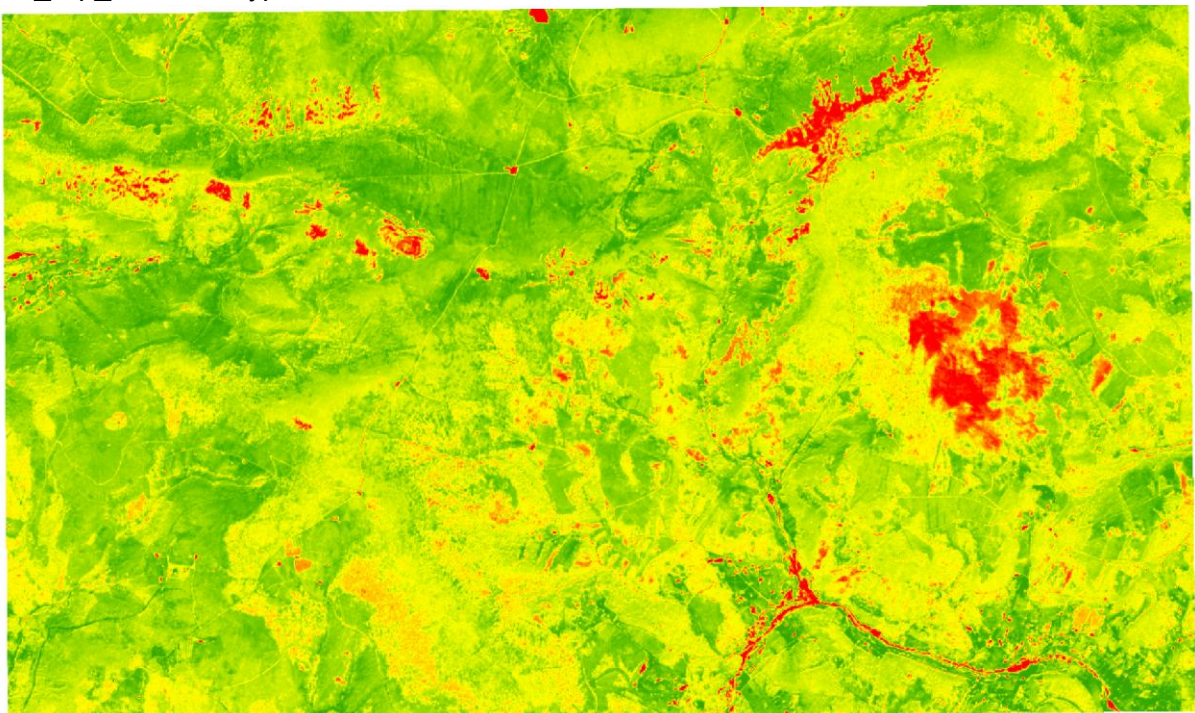
Ostatní soubory se jmenují analogicky. Po zobrazení v libovolném GIS software (např. ArcGIS Pro) při nastavení RGB kompozitu po řadě B04, B03, B02 dostaneme následující výsledek:



2. Vypočítáme vegetační indexy, které mohou být užitečné z hlediska klasifikace. Pro studium vegetace je často používaný index NDVI. Pro snímek z 30. 8. 2024 vypočítáme hodnotu NDVI:

```
02_indexer.py -i  
s:\images\S2A_MSIL2A_20240812T100031_N0511_R122_T33UWS_20240812T140649.SAFE  
_clip.tif -idx_def veg_index_sentinel2.txt -idx NDVI -out_dir S:\indices
```

Výsledek nalezneme ve složce S:\indices jako soubor s:\indices\S2A\_MSIL2A\_20240812T100031\_N0511\_R122\_T33UWS\_20240812T140649.SAFE\_clip\_NDVI.tif. Vypadá následovně:



Pro výpočet je důležitá existence souboru s definicí indexů `veg_index_sentinel2.txt`, ke kterému se zadává cesta při volání skriptu. Vybrané indexy jsou předdefinované a uloženy spolu se zdrojovým kódem skriptu. Seznam všech dostupných indexů využívaných pro Sentinel-2 je zde: [https://www.indexdatabase.de/db/is.php?sensor\\_id=96](https://www.indexdatabase.de/db/is.php?sensor_id=96). Pokud by byla potřeba definici nějakého indexu přidat, otevřeme soubor v libovolném textovém editoru a doplníme do souboru vzorec následovně:

```
# NDVI (Normalized Difference Vegetation Index)
ndvi;(b08 - b04) / (b08 + b04)
```

Na řádku začínajícím znakem `#` je jen komentář, můžeme ho tedy vynechat, ale pro přehlednost je vhodné ho uvést. Důležitější je druhý řádek, kde je název indexu (na který se pak odkazuje při volání skriptu v parametru `idx`), středník, a vzorec pro jeho výpočet. Vzorec je libovolný výraz v Pythonu. Může obsahovat libovolné výrazy používané v ArcGIS v rastrové algebře (viz <https://pro.arcgis.com/en/pro-app/latest/arcpy/spatial-analyst/an-overview-of-the-map-algebra-operators.htm>), s tím, že proměnná reprezentující každé pásmo je označena `b01`, `b02`, ... `b12` a navíc je k dispozici `b8a`.

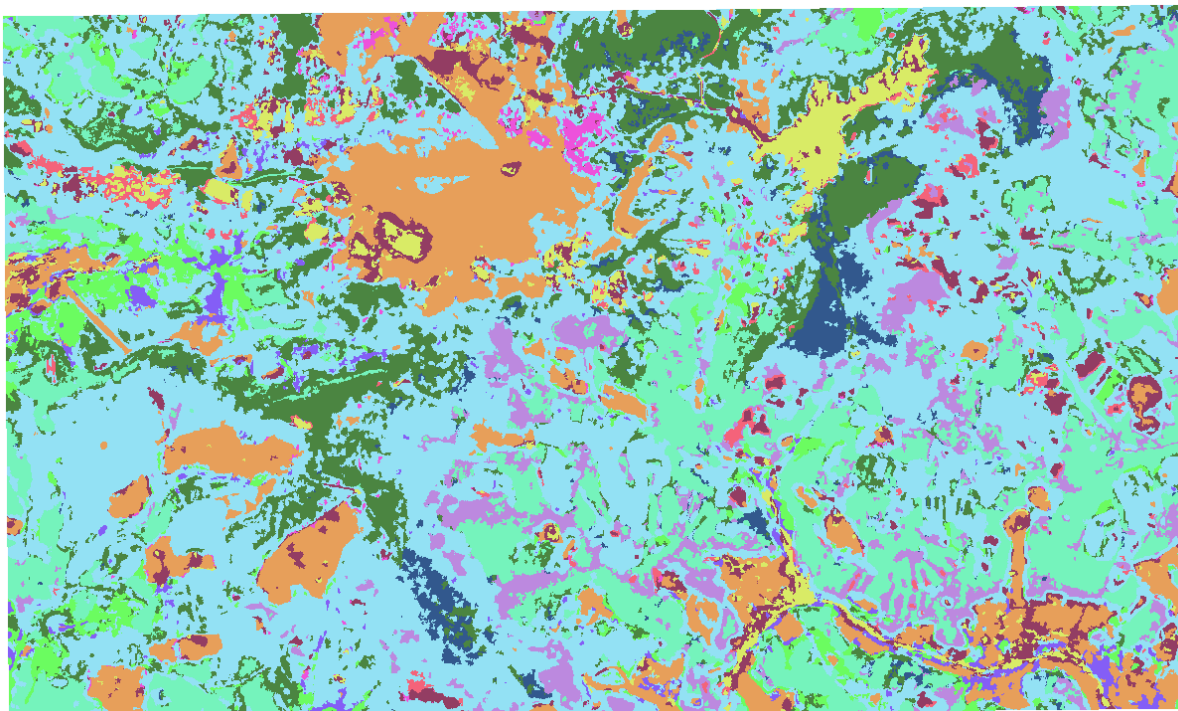
Při výpočtu můžeme v parametru `-idx` zadat více indexů najednou. Oddělujeme je čárkami bez mezer (tedy např. `-idx NDVI,EVI,GNDVI,NDRE`). Skript pak vytvoří více souborů, jejichž název na konci vždy obsahuje text `_` a název daného indexu.

3. Provedeme klasifikaci snímku z 27. 8. 2018. Spustíme následující příkaz:

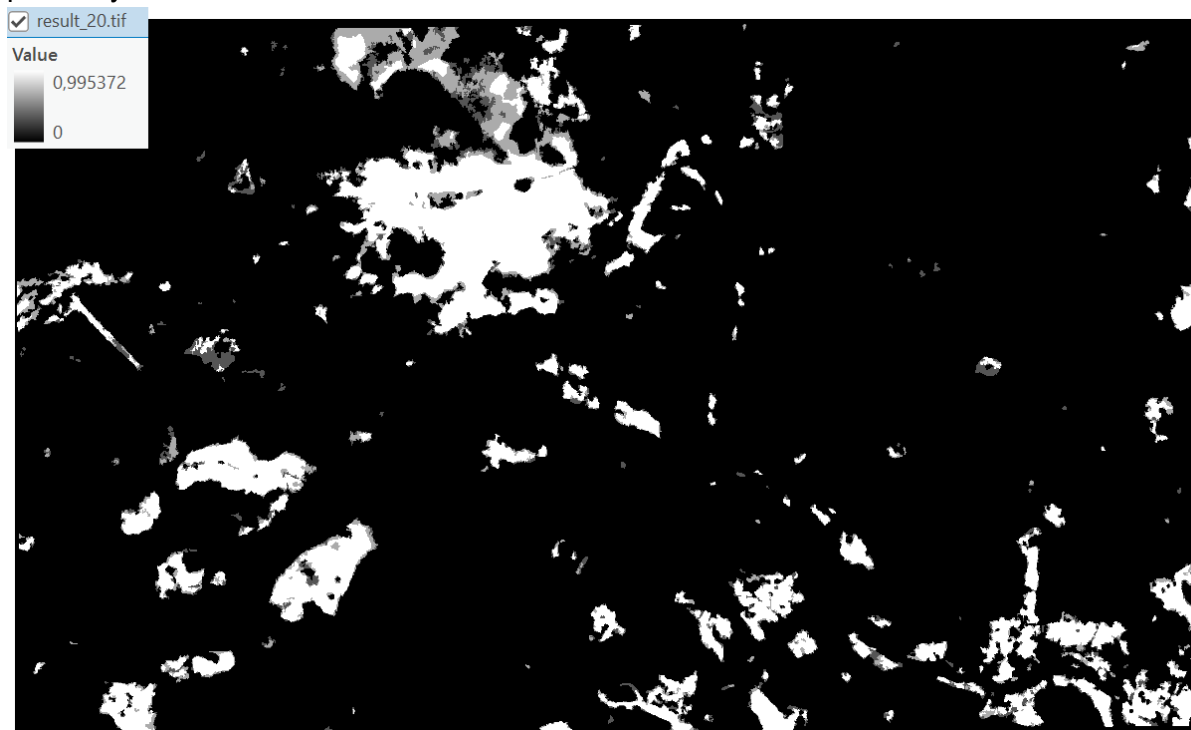
```
03_classifier.py -i
S:\classification2\S2A_MSIL2A_20180807T101021_N0500_R022_T33UWS_20230803T01
0219.SAFE_clip.tif -out
S:\classification2\S2A_MSIL2A_20180807T101021_N0500_R022_T33UWS_20230803T01
0219.SAFE_clip_classified.tif -work S:\classification2 -tv
S:\classification2\tv_data.shp -ratio 0.33 -repeat 10 -min_patch 10 -simpl
4 -select 5
```

Je potřeba mít vytvořenou složku pro výpočet (v uvedeném příkladu `S:\classification2`) a připravená trénovací/validační data. V ukázce byla použity plochy použité pro trénování a validaci při tvorbě výsledku V7. Použili jsme třetinu dat pro trénování a dvě třetiny pro validaci (`-ratio 0.33`), klasifikaci chceme opakovat 10x s náhodnými trénovacími a validačními daty (`-repeat 10`) a pro výsledek použít kombinaci pěti nejlepších klasifikací (`-select 5`), generalizovat výsledek v max. 4 iteracích (`-simpl 4`) na minimální mapovací jednotku 1000 m<sup>2</sup> (`-min_patch 10`; s tím, že víme, že pixel rastru má 10x10 m<sup>2</sup>). Pro provedení klasifikace výsledný soubor vypadá přibližně takto:





Součástí výstupu jsou i soubory result\_<název\_třídy>, obsahující rastr pravděpodobnosti výskytu dané dané třídy na daném místě. Například pro třídu 20 (Louky a pastviny) vypadá příslušný rastr takto:



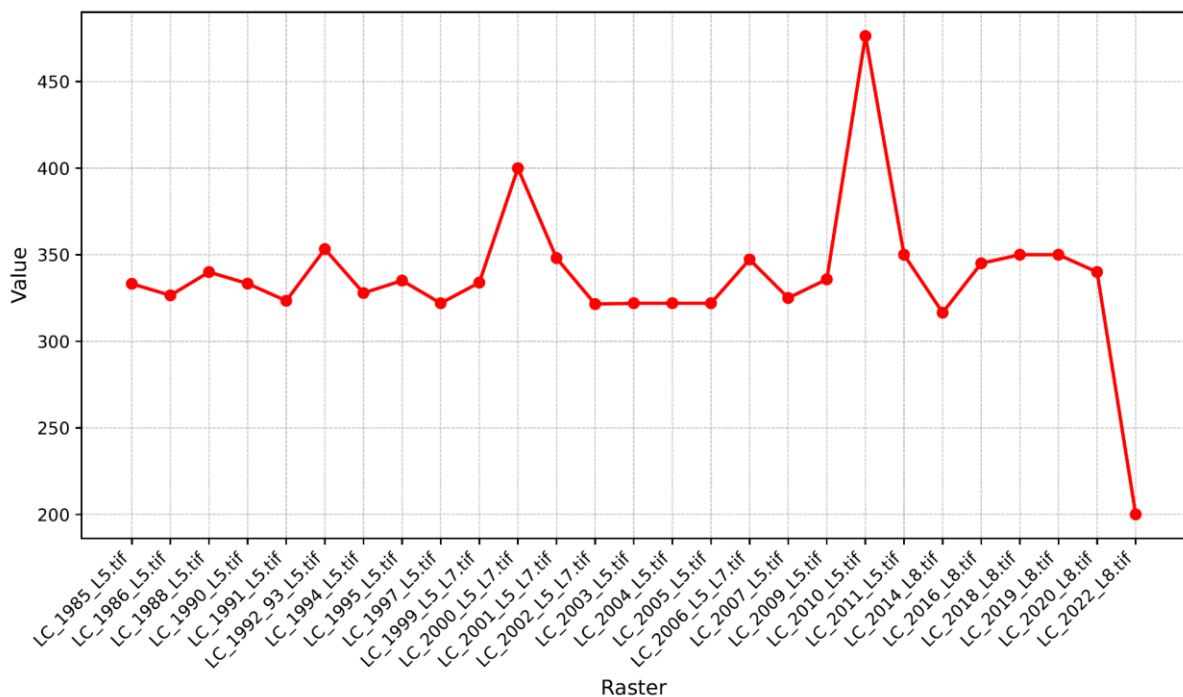
Tyto rastry nahrazují obvyklé statistiky (uživatelskou a zpracovatelskou přesnost), které jsou výsledkem provedení jedné klasifikace. Vznikají kombinací f1-skóre pro konkrétní třídu a výsledek klasifikace. Průměrná hodnota příslušného rastru pak představuje odhad f1-skóre pro příslušnou třídu na základě kombinace více metod klasifikace. Užitečnější je ale informace o hodnotě v konkrétním pixelu, kterou lze interpretovat jako spolehlivost klasifikace (resp. míru nejistoty) v daném místě pro každou třídu zvlášť.

4. porovnáme vývoj v čase a to jak z jednotlivých rastrů (typicky indexů, výsledků klasifikace, případně i zdrojových dat), tak z vektorových vrstev odvozených z výsledků klasifikace.

Nejprve vyzkoušíme data pro vývoj na časové řadě klasifikovaných snímků z Landsatu 5 a 7 a Sentinelu 2, ze kterých vznikl výsledek 7 tohoto projektu. Z pohledu zpracování jde o výstupy kroku 3, spuštěné nad několika snímky z různých časových období. Spustíme příkaz:

```
04_ts_analyzer_pnt.py -i S:\change\seznam_klasifikaci.txt -c  
15.7590942,50.7245099 -o S:\change\output_data.csv -g  
S:\change\output_chart.pdf
```

a krom dat v souboru csv dostaneme také graf zobrazující vývoj tříd v daném místě v čase:



V souboru seznam\_klasifikaci.txt je seznam jednotlivých snímků a pásmo, z něž chceme extrahovat výslednou hodnotu do datového souboru, resp. grafu. Soubor vypadá následovně:

```

S:\change\LC_1985_L5.tif*1
S:\change\LC_1986_L5.tif*1
S:\change\LC_1988_L5.tif*1
S:\change\LC_1990_L5.tif*1
S:\change\LC_1991_L5.tif*1
S:\change\LC_1992_93_L5.tif*1
S:\change\LC_1994_L5.tif*1
S:\change\LC_1995_L5.tif*1
S:\change\LC_1997_L5.tif*1
S:\change\LC_1999_L5_L7.tif*1
S:\change\LC_2000_L5_L7.tif*1
S:\change\LC_2001_L5_L7.tif*1
S:\change\LC_2002_L5_L7.tif*1
S:\change\LC_2003_L5.tif*1
S:\change\LC_2004_L5.tif*1
S:\change\LC_2005_L5.tif*1
S:\change\LC_2006_L5_L7.tif*1
S:\change\LC_2007_L5.tif*1
S:\change\LC_2009_L5.tif*1
S:\change\LC_2010_L5.tif*1
S:\change\LC_2011_L5.tif*1
S:\change\LC_2014_L8.tif*1
S:\change\LC_2016_L8.tif*1
S:\change\LC_2018_L8.tif*1
S:\change\LC_2019_L8.tif*1
S:\change\LC_2020_L8.tif*1
S:\change\LC_2022_L8.tif*1

```

Pásmo je všude 1, protože výsledek klasifikace je jednopásmový soubor.

Dále vyzkoušíme výpočet změn na vektorových datech, odvozených z časové řady výše uvedených snímků z let 1985, 1997, 2010 a 2022. Spustíme příkaz

```

04_ts_analyzer_area.py -i S:\change_v\seznam_lc_vektoru.txt -og
S:\change_v\result.gdb -of result -size 3600 -iters 4 -sep :

```

Minimální požadovaná velikost objektu pro účely detekce změn je 3600 m<sup>2</sup> (-size 3600; plocha přibližně 4 pixelů dat Landsat) a mají se provést čtyři iterace generalizace (-iters 4). Výsledkem zpracování je databáze result.gdb, ve které nalezneme jedinou feature class result. Její atributová tabulka vypadá následovně:

OBJECTID *	Shape *	T_0	T_1	T_2	T_3	Shape_Length	Shape_Area	CHANGE
188	Polygon	200	200	200	200	720	18000	200:200:200:200
189	Polygon	200	200	200	200	960	23400	200:200:200:200
190	Polygon	200	200	200	200	7860	351900	200:200:200:200
191	Polygon	200	200	200	200	240	3600	200:200:200:200
192	Polygon	200	200	200	200	2220	84600	200:200:200:200
193	Polygon	200	200	200	200	2100	76500	200:200:200:200
194	Polygon	200	200	200	200	8460	349200	200:200:200:200
195	Polygon	200	200	200	200	1260	46800	200:200:200:200
196	Polygon	200	200	200	310	780	14400	200:200:200:310
197	Polygon	200	200	200	310	300	3600	200:200:200:310
198	Polygon	200	200	200	310	600	9000	200:200:200:310
199	Polygon	200	200	200	310	300	3600	200:200:200:310
200	Polygon	200	200	200	310	300	5400	200:200:200:310

Pro daný polygon jsou ve sloupcích T\_0 až T\_3 (obecně je sloupců T\_x tolik, kolik je časových řezů v datech) hodnota z daného období, a ve sloupci CHANGE jsou pak tyto hodnoty

agregované. Pomocí funkce Select By Attributes můžeme vybírat území podle charakteru změny, například (uveden vždy výraz, který se zadává do pole s dotazem):

- zjistí území, kde byly celou dobu louky a pastviny (mají kód 200):  
T\_0=200 AND T\_1=200 AND T\_2=200 AND T\_3=200
- zjistí území, kde nedošlo k žádné změně:  
T\_0=T\_1 AND T\_1=T\_2 AND T\_2=T\_3 AND T\_3=T\_0
- zjistí území, kde došlo k nějaké změně v posledním období:  
T\_0=T\_1 AND T\_1=T\_2 AND T\_2=T\_0 AND T\_3=T\_0
- zjistí území, kde byla v prvních dvou obdobích louka a pastvina:  
T\_0=200 AND T\_1=200  
případně  
CHANGE LIKE '200:200:%'
- ...

Tímto způsobem lze položit v zásadě libovolná dotaz týkající se změny krajinného pokryvu. Výsledkem dotazu je výběr polygonů, se kterým lze dále pracovat. Typicky se výběr uloží a dále se mohou počítat věci jako rozloha či počet takových ploch, případně se plochy mohou pomocí Dissolve sloučit do výsledného polygonu například pro účely kartografického zpracování.

Soubor seznam\_lc\_vektoru.txt vypadá následovně:

```
S:\change_v\LC_1985.shp;gridcode  
S:\change_v\LC_1997.shp;gridcode  
S:\change_v\LC_2010.shp;gridcode  
S:\change_v\LC_2022.shp;gridcode
```

Na každém řádku je uveden název souboru a atribut, ve kterém je kód třídy. Pro fungování celého procesu se předpokládá, že atributy jsou napříč jednotlivými obdobími shodné.