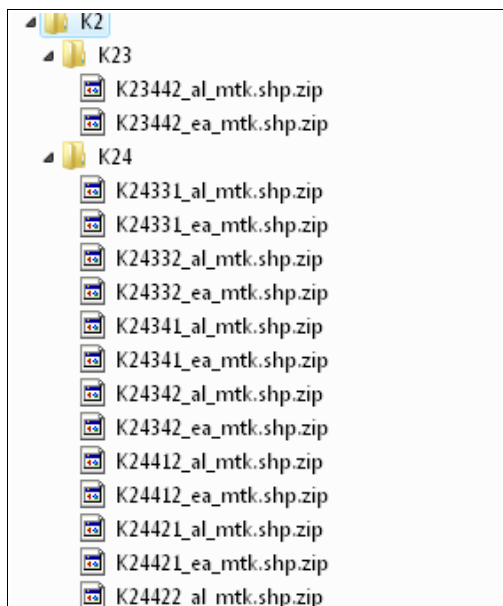


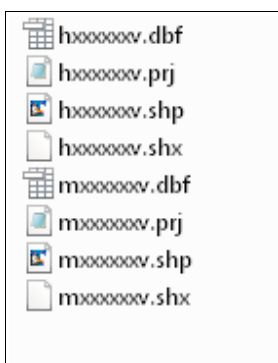
Maastotietokannan torrent-jakelun shapefile-tiedostojen purkaminen zip-arkistoista Windows-komentojoilla

Viimeksi muokattu 5. toukokuuta 2012

Maastotietokannan torrent-jakeluun sisältyy yli 5000 zip-arkistoa, jotka on sijoitettu karttalehtiäon eri mittakaavatasojen tunnusten mukaisesti nimettyihin hakemistoihin. Arkistotiedostojen nimet ovat yksilöllisiä ja nimiin sisältyy sen karttalehden tunnus, jonka alueelta arkiston sisältämät shapefile-tiedostot ovat, esimerkiksi ”K23442”.



Shapefile-tiedostojen jatkokäsittely yhtenä tietojoukkona on kuitenkin hankalaa, koska arkiston sisältämien tiedostojen nimet eivät yksilöllisiä muuten kuin arkiston sisällä. Joka ikinen arkistotiedosto sisältää täsmälleen saman nimisiä tiedostoja, joiden keskellä on kuusi äksää peräkkäin, eikä tiedoston nimen perusteella voi tietää minkä karttalehden shapefile on kyseessä. Tämän tiedon saa selville ainoastaan ottamalla tavalla tai toisella selvää minkä nimisestä zip-tiedostosta shapefilet on purettu, tai avaamalla shapefile karttaohjelmassa yhdessä karttalehtijakokartan kanssa. Lisäksi, koska eri zip-tiedostojen sisällä on saman nimisiä tiedostoja, niin niitä ei voi purkaa yhteiseen hakemistoon.



Haaste: Tuhansien, kymmeneen eri hakemistoihin sijoitettujen zip-arkistojen automaattinen purkaminen, niistä löytyvien samoja nimiä käyttävien shapefile-tiedostojen yksilöivä nimeäminen ja purettujen tiedostojen vieminen yhteiseen hakemistoon

Seuraavassa esitetään ehkä kömpelö, mutta toimivaksi testattu menetelmä Windows-käyttöjärjestelmää varten.

- Windowsin komentoikkuna
- TextPad-tekstinkäsittelyohjelma. Muita vastaavia ja varmasti yhtä hyviä on olemassa.
- 7za-ohjelma, eli 7-zip-arkistointiohjelman komentoriviversio. Muutkin komentoriviltä suoritettavat arkistointiohjelmat luultavasti ovat käyttökelpoisia.

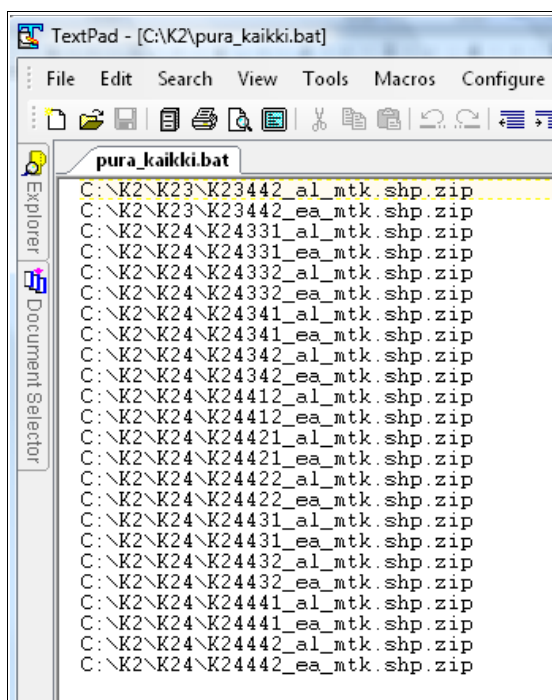
Vaihe 1: Kaikkiin alihakemistoihin sisältyvien zip-tiedostojen listaaminen tekstitiedostoon

Avataan Windowsin komentoikkuna, siirrytään maastotietokannan juurihakemistoon, ja annetaan siellä seuraava komento:

```
dir /s /b *.zip >pura_kaikki.bat
```

Komento tekee vain pelkät tiedostonimet (/b) sisältävän hakemistolistauksen lähtöhakemistosta ja kaikista sen alihakemistoista (/s) ja ohjaa tuloksen (>) tiedostoon jonka nimi on ”pura_kaikki.bat”.

Tulostiedosto näyttää tämän tapaiselta

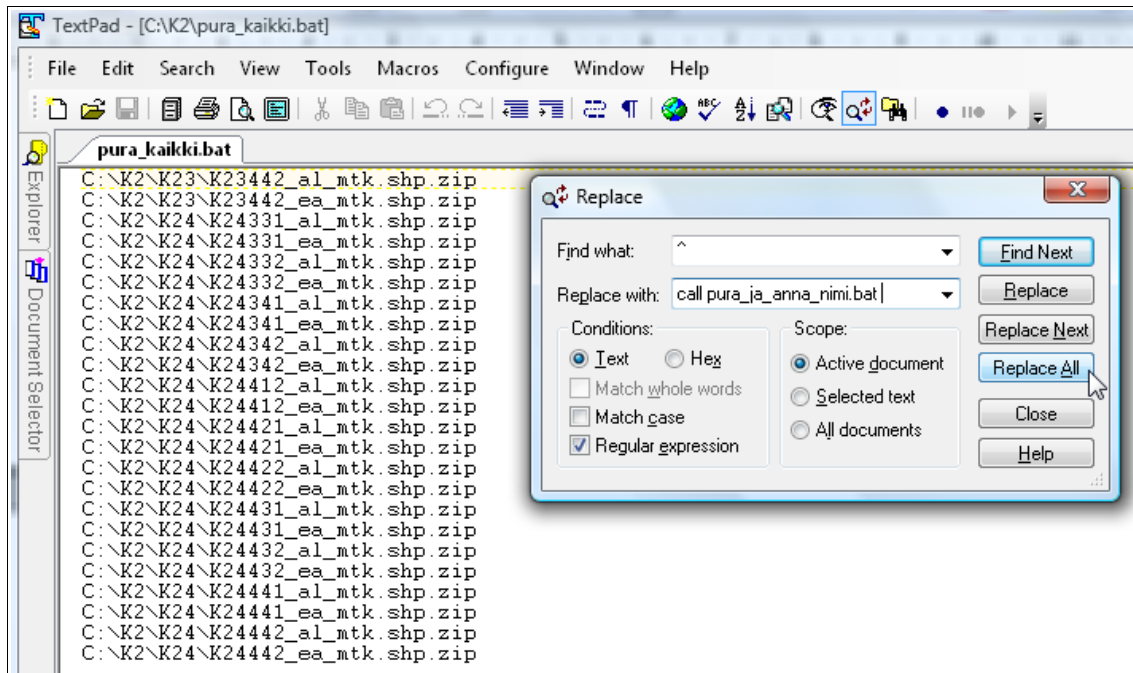


Vaihe 2: Tiedostolistan muokkaaminen purkamisen käynnistäväksi komentojonoksi

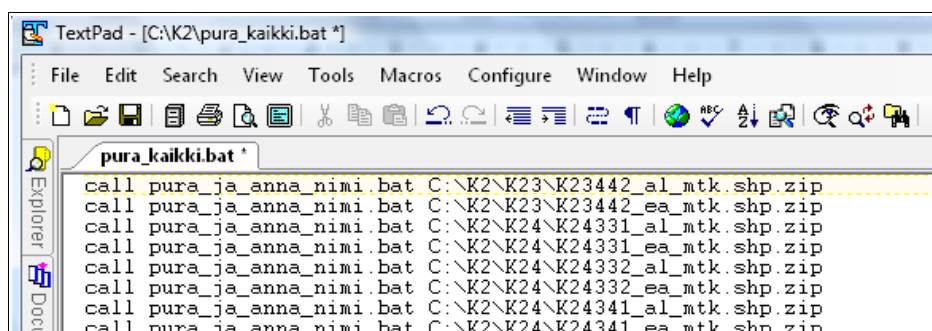
Edellä luotuun komentojonopohjaan tehdään jokaiselle riville kaksi muutosta:

- lisätään rivin alkuun kutsu toiseen komentojonoon
- kopioidaan karttalehtitunnus zip-tiedoston nimestä rivin loppuun

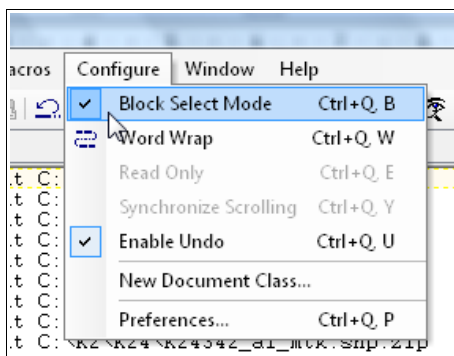
Muutosten teko onnistuu millä tahansa tekstinkäsittelyohjelmalla, mutta joillakin niistä helpommin kuin joillakin muilla. Helppoudella on merkitystä, koska tiedostossa on rivejä yli 5000. Seuraavassa esitetään, kuinka nämä muokkaukset voidaan tehdä TextPad-ohjelmalla.



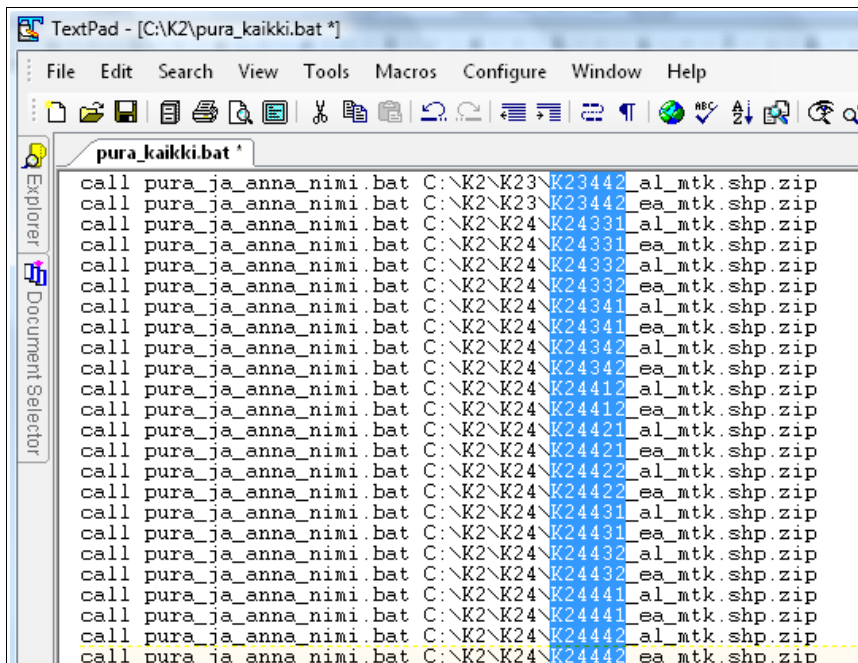
Tiedosto korvaustoiminnon suorittamisen jälkeen. Kannattaa tarkistaa tiedoston loppu ja poistaa sinne mahdollisesti syntynyt turha rivi.



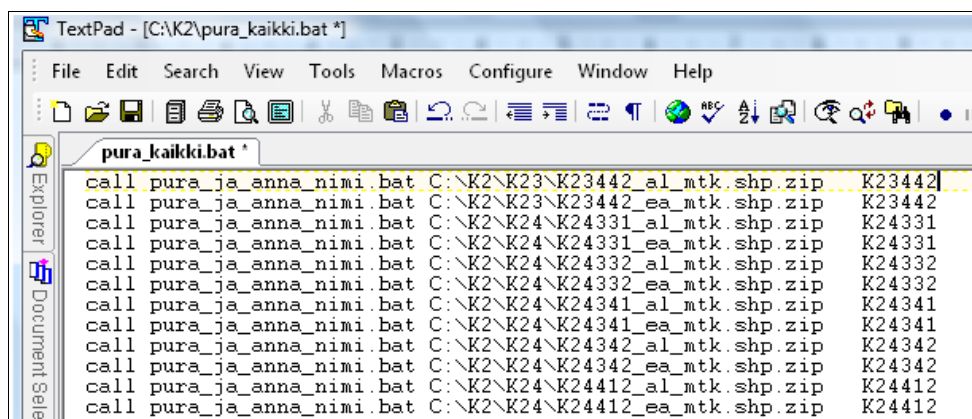
Seuraavaksi laitetaan TextPad laatikkovalintatilaan.



Valitaan zip-tiedostojen nimestä se osa, joka kertoo karttalehden tunnuksen. Koska kaikki zip-tiedostot on – toivottavasti – nimetty samalla mallilla, niin ne voidaan valita kaikki melko helposti laatikkovalinnalla.



Kopioidaan laatikon sisällä oleva teksti, liitetään se rivien jatkeeksi ja tallennetaan tiedosto.



Vaihe 3: Varsinaisen työn tekävän komentojonon kirjoittaminen

Kirjoitetaan komentojono ”pura_ja_anna_nimi.bat” ja tallennetaan se samaan hakemistoon kuin edellä luotu ”pura_kaikki.bat”. Komentojono on seuraavan mallin mukainen, mutta se on muokattava vastaamaan omalla koneella olevia hakemistopolkuja.

```
c:\ohjelmat\7za\7za e %1 -oc:\zipout  
ren c:\zipout\?xxxxxx?.* ?%2?.*
```

Vaihe 4: Pura kaikki maastotietokannan shapefilet arkistoista, anna niille yksilöivät nimet ja kirjoita ne yhteen yhteen hakemistoon yhdellä komennolla

Kun edellä luodut kaksi komentojonoa ovat valmiina niin kaikki onkin valmista yhdellä komennolla tapahtuvaan automaattiseen kaikkien zip-tiedostojen purkamiseen ja arkistoista löytyneiden shapefile-tiedostojen nimeämiseen niitä vastaavien karttalehtitunnusten mukaan. Komento on

```
pura_kaikki
```

Mitä oikein tapahtuu?

Windows alkaa suorittaa käynnistävää komentojonoa ”pura_kaikki.bat” rivi kerrallaan. Esimerkin ensimmäinen rivi on

```
call pura_ja_anna_nimi.bat C:\K2\K23\K23442_al_mtk.shp.zip K23442
```

Kun rivi on luettu, niin Windows kutsuu komentojonoa ”pura_ja_anna_nimi.bat” ja välittää sille kaksi parametria, joista ensimmäinen on C:\K2\K23\K23442_al_mtk.shp.zip ja toinen K23442.

”Pura_ja_anna_nimi.bat” -komentojonon ensimmäinen rivi on

```
c:\ohjelmat\7za\7za e %1 -oc:\zipout
```

Komento tarkoittaa:

- suorita ohjelma c:\ohjelmat\7za\7za
- määrää ohjelma purkamaan arkisto antamalla parametri ”e” eli extract
- ota arkistotiedoston nimi kutsuvan komentojonon ensimmäisestä muuttujasta, johon viitataan termillä ”%1”
- kirjoita puretut tiedostot hakemistoon c:\zipout

Komennon suorittamisen jälkeen zipout-hakemistossa on zip-arkistosta puretut tiedostot.

```
Järjestelmänvalvoja: GDAL 19 (MSVC 2010) Command Prompt - pura_kaikki
C:\K2>pura_kaikki
C:\K2>call pura_ja_anna_nimi.bat C:\K2\K23\K23442_al_mtk.shp.zip K23442
C:\K2>c:\ohjelmat\7za\7za e C:\K2\K23\K23442_al_mtk.shp.zip -oc:\zipout
7-Zip (A) 9.20 Copyright (c) 1999-2010 Igor Pavlov 2010-11-18
Processing archive: C:\K2\K23\K23442_al_mtk.shp.zip
Extracting hxxxxxxp.dbf
Extracting hxxxxxxp.prj
Extracting hxxxxxxp.shp
Extracting hxxxxxxp.shx
Extracting mxxxxxxp.dbf
Extracting mxxxxxxp.prj
Extracting mxxxxxxp.shp
Extracting mxxxxxxp.shx
Everything is Ok
Files: 8
Size: 3384
Compressed: 2259
```

Tilanne on tässä vaiheessa periaatteessa hyvä, mutta muuttuu onnettomaksi jos samaan hakemistoon puretaan seuraava maastotietokannan zip-arkisto, koska siihen sisältyy täsmälleen saman nimisiä tiedostoja. Tämä johtaisi joko virheilmoitukseen, joka kertoo että tiedosto on jo olemassa, tai vanhojen tiedostojen päälle kirjoittamiseen. Jotta päästäisiin järkevään lopputulokseen, niin ennen kuin aloitetaan seuraavan arkiston purkaminen suoritetaan toinen komento, joka antaa jo puretuille tiedostoille yksilölliset nimet. Komento on tämä:

```
ren c:\zipout\?xxxxxx?.* ?%2?.*
```

Komento tarkoittaa

- nimeä uudelleen (ren)
- hakemistosta c:\zipout
- tiedostot, joiden keskellä on kuusi x-kirjainta, niiden edessä ja jäljessä yksi merkki (jokeri ”?”), ja tiedostotarkenteena mikä tahansa, kuinka pitkä tahansa merkkijono (jokeri ”*”)
- siten, että ensimmäinen ja tiedostotarkenteen edessä oleva merkki sekä tarkenne säilyvät entisinä, mutta kuuden x-kirjaimen sijaan kirjoitetaan kutsuvan komentojonon toinen parametri, johon viitataan termillä ”%2”.

Uudelleen nimeämisen jälkeen purkuhakemistoon voidaan taas kirjoittaa uusia xxxxxx-nimettyjä tiedostoja. ”Pura_ja_anna_nimi.bat” on suoritettu loppuun ja järjestelmä palaa takaisin käynnistävään komentojonoon ”pura_kaikki.bat” ja lukee siitä seuraavan rivin. Tällä tavalla käydään läpi kaikki listalta löytyvät zip-arkistot.

```
Järjestelmänvalvoja: C:\Windows\system32\cmd.exe
C:\zipout>dir
Asemalla C ei ole nimeä.
Aseman sarjanumero on 466E-C444

Kansio C:\zipout

05.05.2012 01:29 <KANSIO> .
05.05.2012 01:29 <KANSIO> ..
30.04.2012 11:16 954 hK23442p.dbf
30.04.2012 11:16 410 hK23442p.prj
30.04.2012 11:16 220 hK23442p.shp
30.04.2012 11:16 108 hK23442p.shx
30.04.2012 11:16 954 mK23442p.dbf
30.04.2012 11:16 410 mK23442p.prj
30.04.2012 11:16 220 mK23442p.shp
30.04.2012 11:16 108 mK23442p.shx
8 tiedosto(a) 3 384 tavua
```

Jälkisanat: Miksi näin mutkikkaasti?

Komentojonomestareilla ja muuten vain ohjelmointitaitoisilla on tapana kirjoittaa nämä call-kutsuilla yhdistetyt erilliset komentojonot yhdeksi ainoaksi komentojonoksi. Tiedostoon ohjatun `dir /b /s`-listauksen sijasta tätä porukkaa viehättää kirjoittaa jotain `for %%i in ... do ...`-juttuja. Tuloksena on komentojonotiedosto, jossa on vähemmän rivejä ja enemmän outoja merkkejä. Siihen, mitä komentojonolla saadaan aikaan ei tyylillä ole vaikutusta.

Minulla on kolme syytä käyttää call-kutsuja

1. Osaaminen ei muuhun riitä.
2. Testaaminen on helpompaa, kun on selkeitä paikkoja pysäyttää suoritus lisäämällä komentojonoon `pause`-komentoja. Esimerkiksi voi testata komentojonon toiminnan ensin yhdellä zipillä, tarkistaa lopputuloksen ja antaa vasta sitten ajon jatkua ja lähteä itse kahville.

```
pura_ja_anna_nimi.bat  pura_kaikki.bat *
call pura_ja_anna_nimi.bat C:\K2\K23\K23442_al_mtk.shp.zip K23442
pause
call pura_ja_anna_nimi.bat C:\K2\K23\K23442_ea_mtk.shp.zip K23442
call pura_ja_anna_nimi.bat C:\K2\K24\K24331_al_mtk.shp.zip K24331
```

Tai sitten voi testata suorittavaa komentojonoa vaihe kerrallaan

```
pura_ja_anna_nimi.bat *  pura_kaikki.bat *
c:\ohjelmat\7za\7za e %1 -oc:\zipout
pause
ren c:\zipout\?xxxxxx?.* ?%2?.*
```

3. Call-menetelmä listaa näytölle tietoja, joista näkee missä vaiheessa komentojonot suoritus on menossa ilman että tarvitsee kirjoittaa erikseen koodia sitä varten. Tästä voi olla paljonkin hyötyä jos esimerkiksi levytila on yllättäen loppumassa. Komentoikkunan listauksen perusteella on helppo katsoa missä asti käynnistävän komentojonon suoritus on menossa. Tarpeen vaatiessa komentojonoa voidaan muokata lisäämällä suoritettavana olevan kohdan jälkeen uusi `pause`-komento. Tällä tavalla ajo voidaan joko keskeyttää tai lopettaa kokonaan hallitusti ja haluttuun paikkaan. Kokonaan pysäytettyäkin ajoa voidaan myöhemmin jatkaa poistamalla jo suoritettavat rivit käynnistävästä komentojonosta. Tätä ei ole `for..in..do`-systemillä helppo tehdä.

```
ca Järjestelmänvalvoja: GDAL 19 (MSVC 2010) Command Prompt
Extracting sxxxxxxp.dbf
Extracting sxxxxxxp.prj
Extracting sxxxxxxp.shp
Extracting sxxxxxxp.shx

Everything is Ok

Files: 20
Size: 338120
Compressed: 140391

C:\K2>ren c:\zipout\?xxxxxx?.* ?K24442?.*
C:\K2>call pura_ja_anna_nimi.bat C:\K2\K24\K24442_ea_mtk.shp.zip K24442
```

Huom! Jos muokkaa komentojonoa kesken ajon, niin `pause` kannattaa sijoittaa vasta pari riviä suoritettavana olevan rivin jälkeen. Heti seuraavalle riville lisättyä `pause`-komentoa ei välttämättä lueta ollenkaan.