Handbok för att använda skript i tavastland.py

Metoder körs generellt i nämnda ordning då de i regel bygger på att tidigare steg är utförda.

**Skapa objekt av klass FileHandler**

* kwargs:
  + mit\_directory: sökväg till mappen där mit-filerna ligger.
  + co2\_directory: sökväg till mappen där co2-filerna ligger.
  + export\_directory: sökväg till mapp där du vill ha exporterna. Alla filnamn på exporterna sätts automatiskt och sparas i den här mappen. Du kan även sätta exportmapp som argument i ”save\_”-metoderna.
  + log\_info: information om hur man vill logga. Se modul ”log” i sharkpylib.

Objektet från den här klassen används sedan för att utföra olika operationer och manipulationer av data. Exemplet utgår från att vi skapat objektet ”handler” från klassen FileHandler, dvs:   
handler = tavastland.FileHandler(\*\*kwargs).   
Här mappas filer med sökvägar. Även start och stopptid för varje fil plockas ut och sparas.

**handler.set\_time\_delta**

* kwargs
  + argument som kan ges till pd.Timedelta (t.ex. seconds=30)

Sätter den tidstollerans som är acceptabel för sammanslagning (merge) av data. Vid beräkningar tas utökas scoopet med den här toleransen. Det betyder att en den första tid i primära filtypen (mit eller co2) kan matchas mot en tidigare tid så länge den är innanför time\_delta. Samma gäller för sista tid.

**handler.set\_time\_range**

* kwargs
  + time\_start: datetime.datetime objekt som beskriver starten på den period du vill jobba med.
  + time\_end: datetime.datetime objekt som beskriver slutet på den period du vill jobba med.
  + time + file\_type: ange datetime.datetime objekt för tiden du är intresserad av och filtypen som det gäller (mit, co2). Start- och stopptid från den fil som matchar kommer att användas.
  + file\_id: Id på den fil du vill jobba med (filnamn). Start- och stopptid från den fil som matchar kommer att användas.

Här bestäms/sätts den period du vill jobba med. Senare exporter kommer att ha dessa tider som start och stop.

**handler.load\_data**

Inga input

Laddar data inom angivet tidsspann (se föregående steg).

**handler.merge\_data**

* Arg 1: primär filtyp (’mit’ eller ’co2’) som bas för ”merge”
* Arg 2: sekundär filtyp (’mit’ eller ’co2’) att ”merga” med.

Mergar data med arg 1 som utgångspunkt. Mergar på tid med tolerans angivet i handler.set\_timedelta. Närmast tid matchas.

**handler.calculate\_pCO2**

* Kwargs:
  + std\_time\_acceptance: Ej implementerad än. Är det bättre att ta kalibrerade värden om tiden från standardgaser är få lång?

Beräknar pCO2 samt underlag som behövs. Standardgaser beräknas löpande. För data som inte har någon standardgaser inom tidsspannet (typiskt i början av fil) plockas de senaste standardgaserna som hittas innan time\_start.

**handler.save\_merge\_data**

* directory: (valfri) Mapp för export. Måste anges om detta inte gjorts vin initieringen av FileHandler.

Exporterar mergad data till exportmappen. Filnamnet sätts automatisk.