

Appendix: NetCDF header

De PROTIDE API voorziet in de mogelijkheid om hydrometeo data (waterstand, stroming, wind, 2D golfspectra) aan te leveren in het NetCDF v3 (classic) formaat. Zie ook:

https://www.unidata.ucar.edu/software/netcdf/docs/file_format_specifications.html

Onderstaand wordt per type hydrometeo data de NetCDF header beschreven volgens de huidige stand van zaken. Deze beschrijving kan nog niet als definitief worden beschouwd. Echter, wij voorzien geen fundamentele wijzigingen tussen nu en de definitieve specificatie van de PROTIDE API.

Waterstand

```
dimensions:
  time = t;                // t = aantal tijden variables:
  int    time(time);       // Seconden sinds 1-1-1970
  float  water_level__m(time); // Waterstand (in m) t.o.v. overeengekomen
                                // referentievlak
```

Stroming

```
dimensions:
  time = t;                // t = aantal tijden variables:
  int    time(time);       // Seconden sinds 1-1-1970
  float  speed__m_s(time); // Absolute stromingssnelheid in m/s
  float  direction__deg(time); // Stromingsrichting t.o.v. noorden in graden
                                // 0 = Noord, 90 = Oost, 180 = Zuid, 270 =
West
```

Wind

```
dimensions:
  time = t;                // t = aantal tijden variables:
  int    time(time);       // Seconden sinds 1-1-1970
  float  speed__m_s(time); // Absolute windsnelheid in m/s
  float  direction__deg(time); // Windrichting t.o.v. noorden in graden
                                // 0 = Noord, 90 = Oost, 180 = Zuid, 270 =
West
```

2D golfspectra

Een 2D golfspectrum wordt beschreven door een energiedichtheid per frequentieband en richtingsband. Er wordt aangenomen dat alle richtingsbanden gelijke grootte hebben, en dat de opgegeven frequentie/richting het midden van een band aangeeft.

```
dimensions:
    time = t;           // t = aantal tijden
    frequency_hz = f;  // f = aantal frequentiebanden
    direction_deg = d; // d = aantal richtingsbanden
variables:
    float density_m2s_deg(time,           // Energiedichtheid in m^2*s/deg
                          frequency_hz,
                          direction_deg);
    int    time(time);           // Seconden sinds 1-1-1970
    float  frequency_hz(frequency_hz);   // Frequentie in Hz
    float  direction_deg(direction_deg);  // Richting in graden
```