

## Requirements for the biological response model PhytoBasinRisk

U. Mischke, A. Gericke, M. Venohr 2018. PhytoBasinRisk v1.100816b, <http://www.moneris.igb-berlin.de/index.php/phytobasinrisk.html>

- (1) Relevance of phytoplankton in your river system: A concentration of 20 µg/L chlorophyll a should be surpassed at least at one monitoring station.
- (2) Agree the license – no commercial use! PhytoBasinRisk is released under the Creative Commons license CC BY-NC-SA 4.0 without any warranties.  
<https://creativecommons.org/licenses/by-nc-sa/4.0/>
- (3) Topology of sub-catchments\* must be a directed acyclic graph in downstream direction
- (4) Areas of sub-catchments should ideally be small enough so that water retention time is below one day; lakes are handled with specific assumptions.
- (5) Complete input data sets including discharge and nutrient concentration (TP, DIN, SI)
  - a. The application for the Middle Elbe used the output of the model MONERIS
  - b. River and lake segments marked as “main” or “tributary” (e.g. “MR” or “TRIB”)
  - c. River attributes averaged for sub-catchment (e.g. depth, length, width, slope, altitude etc.)
  - d. Number of lakes, lake mixing type and total lake length in each sub-catchment
  - e. Information to estimate the shading factor by riparian shading based on land cover along the river network (within 10 m distance)
- (6) Basic IT skills: a) working with databases, b) running programs from the commandline interface (shell), and c) installing of required 3<sup>rd</sup>-party Python libraries and ODBC drivers
- (7) See model description and doku\*-Excel-file for more details

\*e.g. taken from a water quality model like MONERIS with which PhytoBasinRisk can be coupled

### Technical requirements:

- PhytoBasinRisk is written in Python 2.7 and depends on these required libraries (tested version): pandas (0.17.1), numpy (1.10.4), networkx (1.11)  
optionally for database access: sqlalchemy (1.0.13), pycopg2 (2.6.1), pypyodbc (1.3.3), pyodbc (3.0.10)
- PhytoBasinRisk is OS-independent; please check the availability of ODBC drivers for specific OS versions
- SQLAlchemy allows using any DB supported by that library (see <http://docs.sqlalchemy.org/en/latest/dialects/index.html> for more details)
- Pypyodbc and pyodbc allow using MS Access databases as input database only under MS Windows, note: the 32bit version of MS Office requires the 32bit version of Python