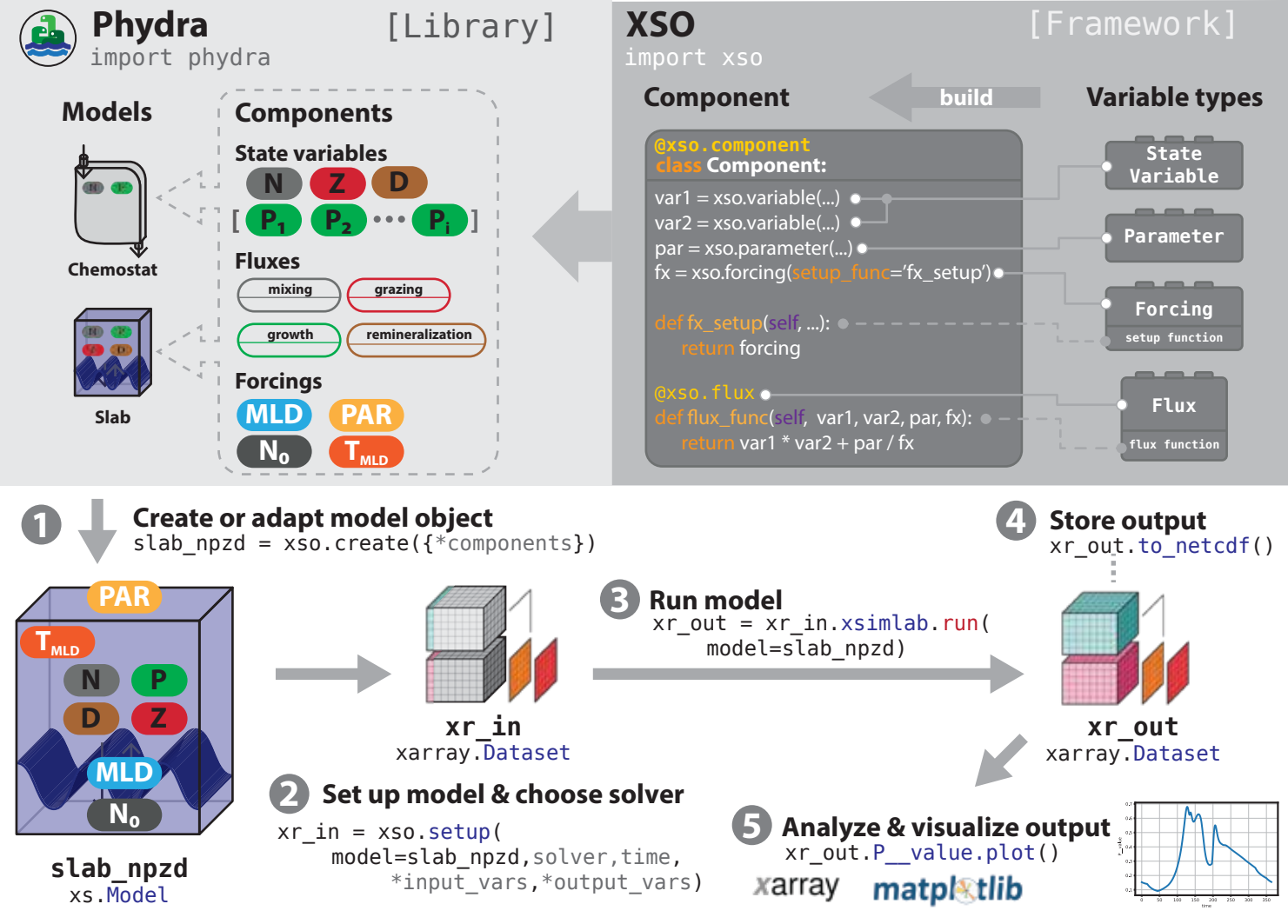
Phydra
import phydra
[Library]
XSO
import xso
[Framework]
Models
Components
State variables
N
Z
D
[P1 P2 ••• Pi]
Fluxes
mixing
grazing
growth
remineralization
Forcings
MLD
PAR
N0
TMLD
Chemostat
Slab
Component
build
Variable types
@xso.component
class Component:
var1 = xso.variable(...)
var2 = xso.variable(...)
par = xso.parameter(...)
fx = xso.forcing(setup_func='fx_setup')
def fx_setup(self, ...):
return forcing
@xso.flux
def flux_func(self, var1, var2, par, fx):
return var1 * var2 + par / fx
State Variable
Parameter
Forcing
setup function
Flux
flux function
1
Create or adapt model object
slab_npzd = xso.create({*components})
PAR
TMLD
N
P
D
Z
MLD
N0
slab_npzd
xs.Model
xr_in
xarray.Dataset
2
Set up model & choose solver
xr_in = xso.setup(
model=slab_npzd,solver,time,
*input_vars,*output_vars)
3
Run model
xr_out = xr_in.xsimlab.run(
model=slab_npzd)
xr_out
xarray.Dataset
4
Store output
xr_out.to_netcdf()
5
Analyze & visualize output
xr_out.P__value.plot()
xarray
matplotlib