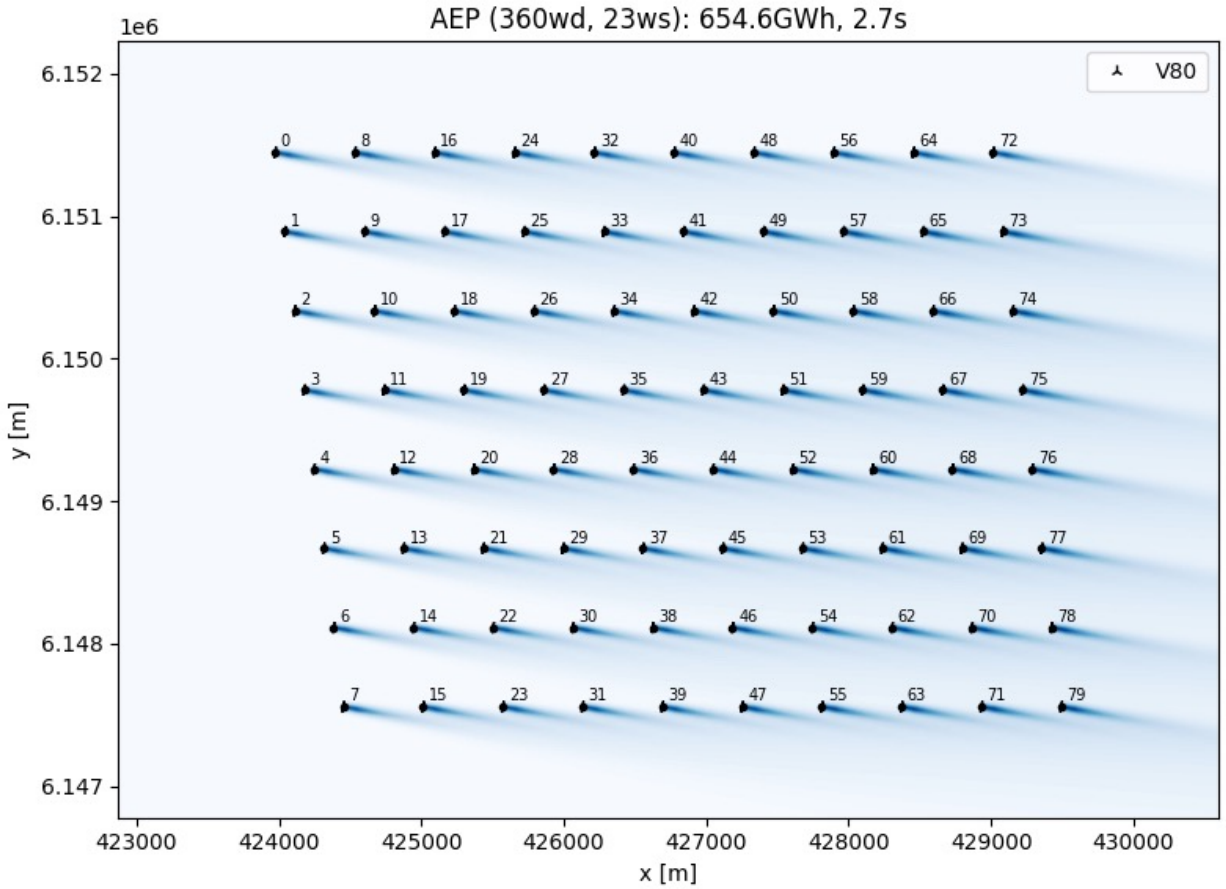

DTU

Mads M. Pedersen

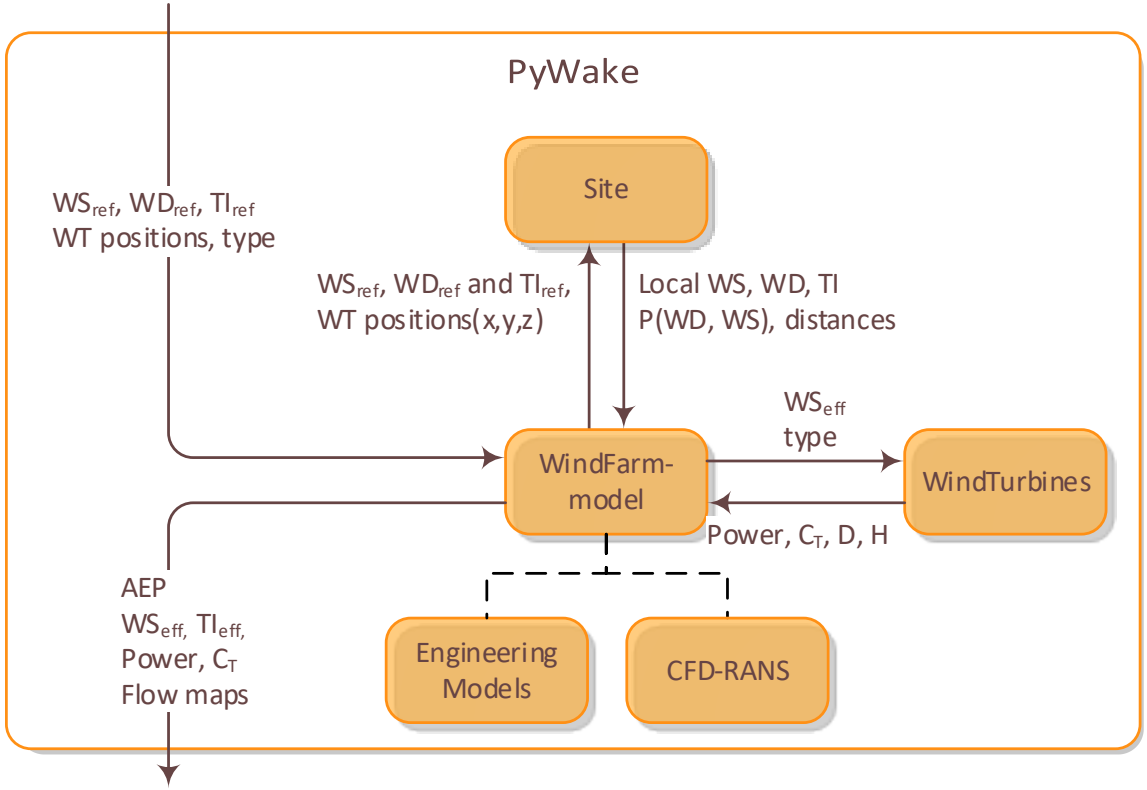
From PyWake to Dynamiks

PyWake

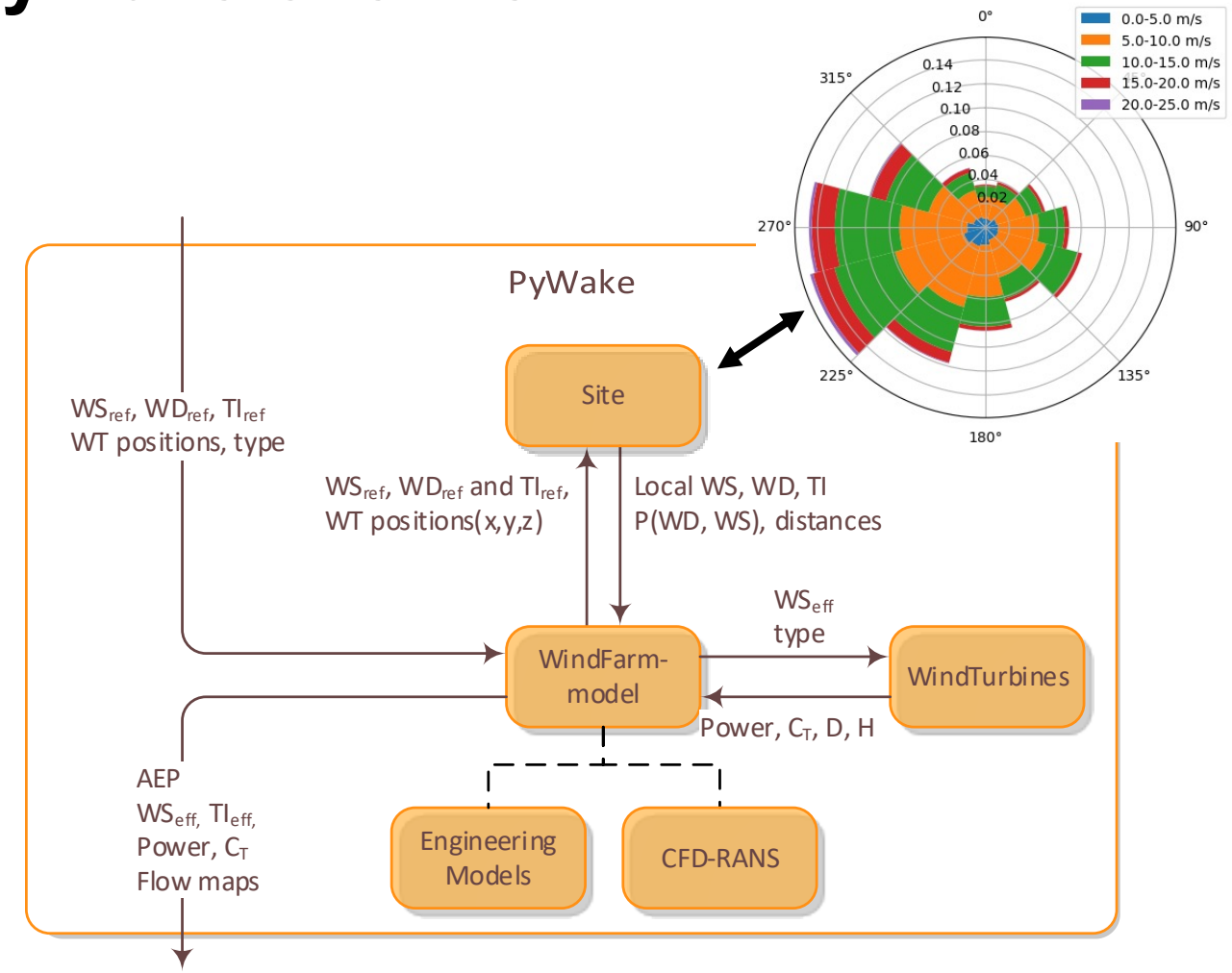
- Open source python package from DTU
- Static wind farm simulator
- Computes:
  - Effective wind speed (including wakes)
  - Power, AEP
  - Flow maps
  - Analytical gradients via automatic differentiation
- Very fast
- Suitable for optimization



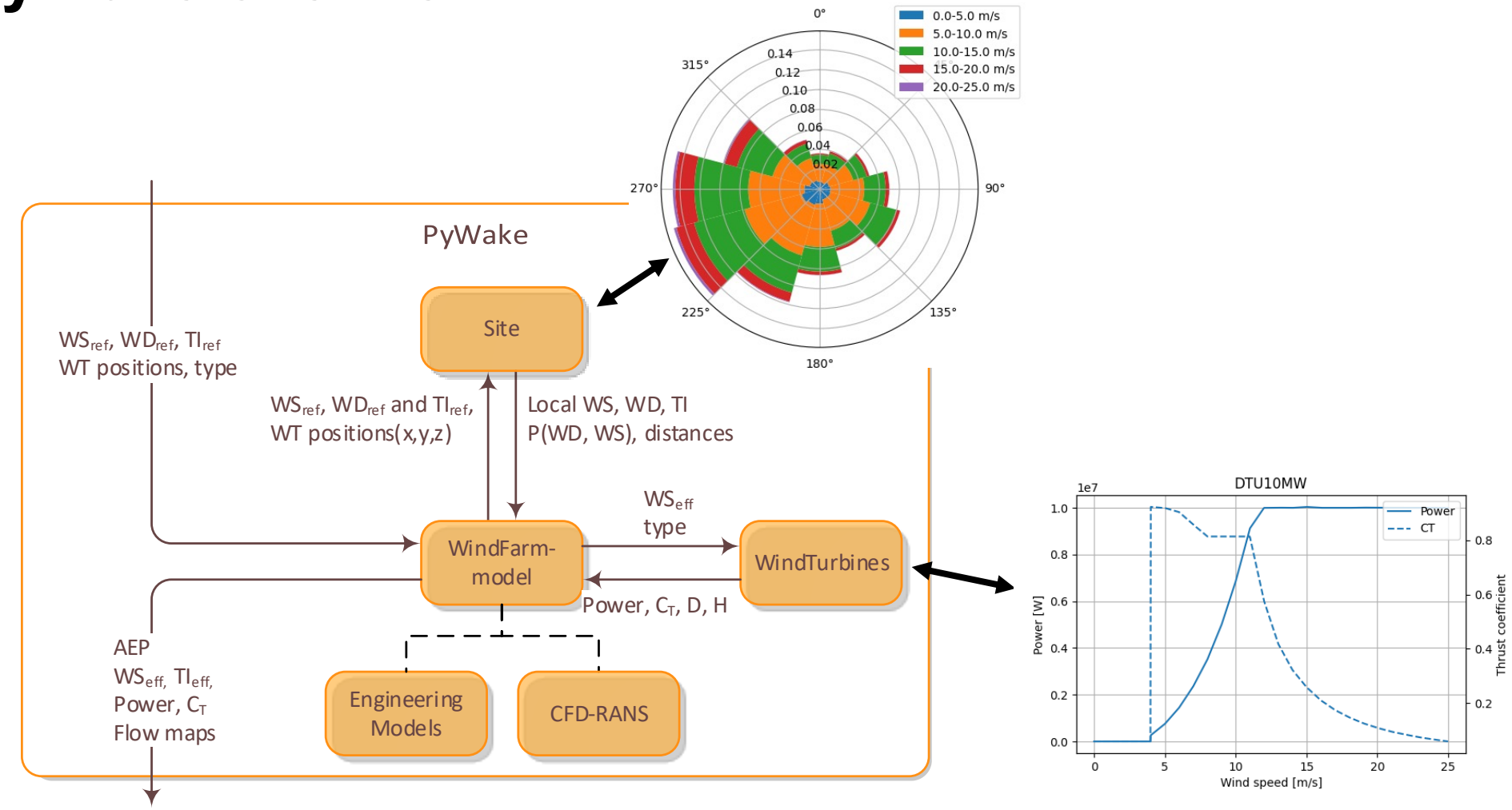
PyWake overview



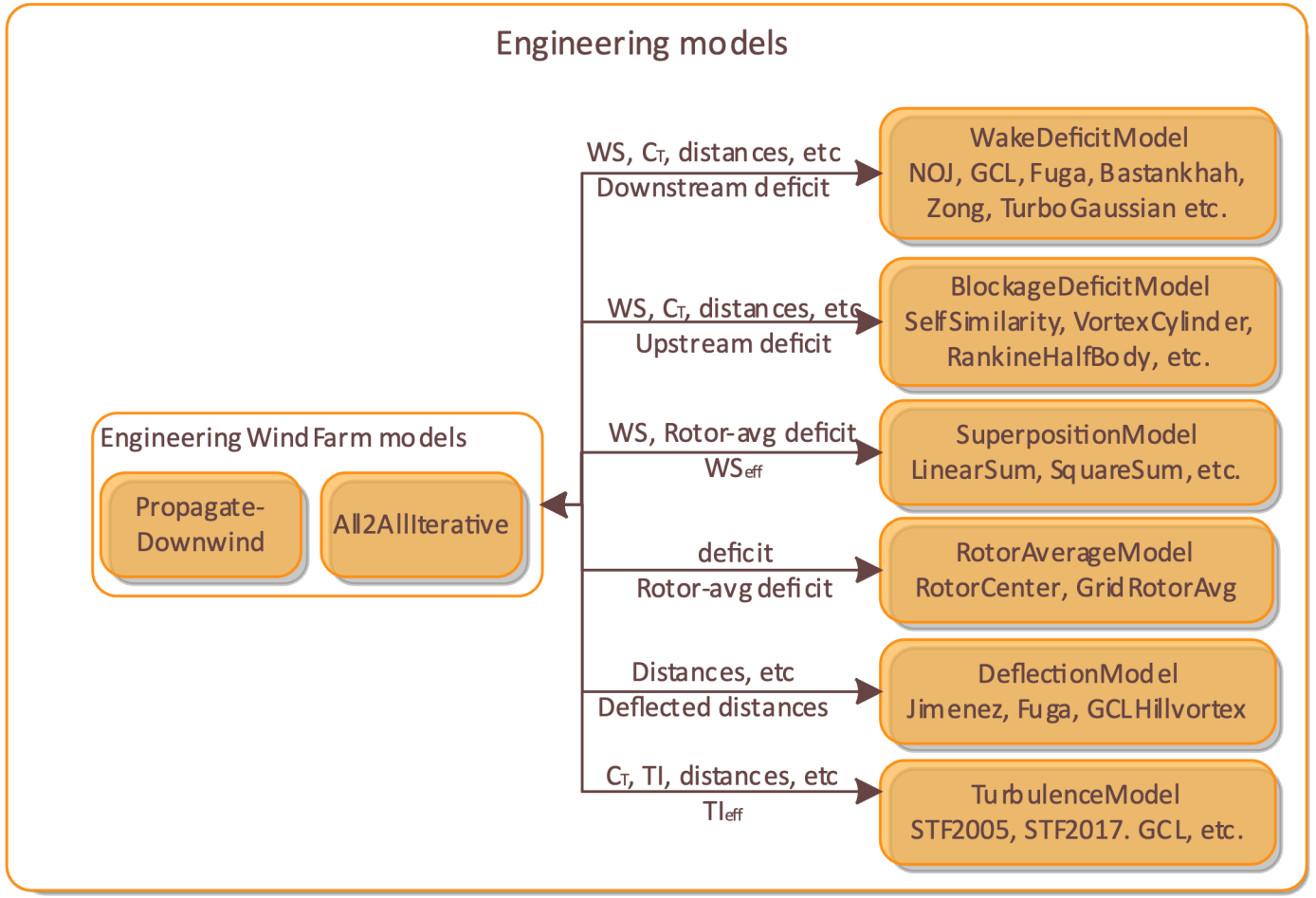
PyWake overview



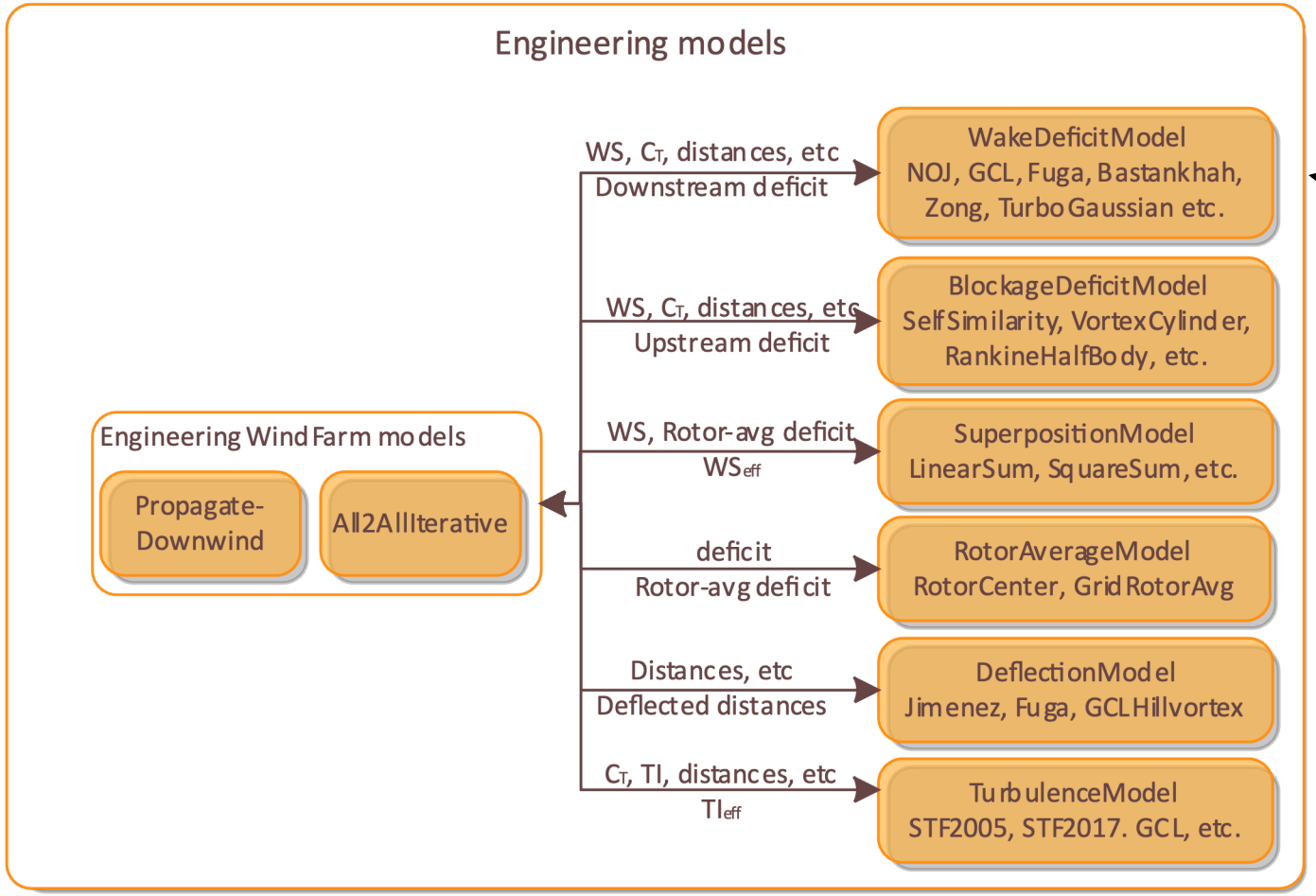
PyWake overview

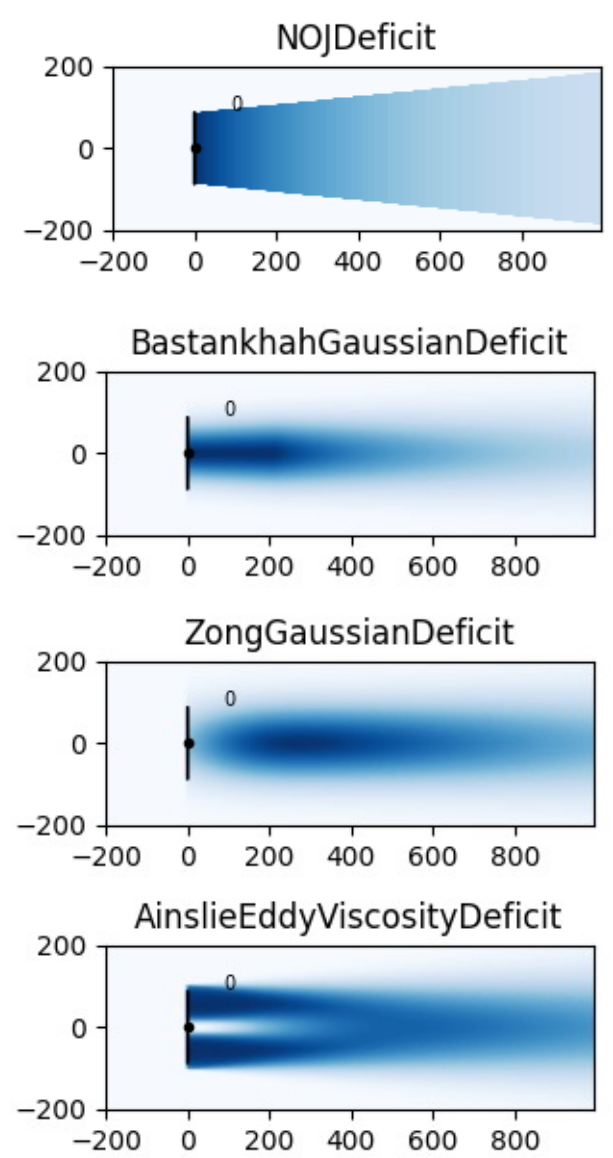


PyWake engineering models

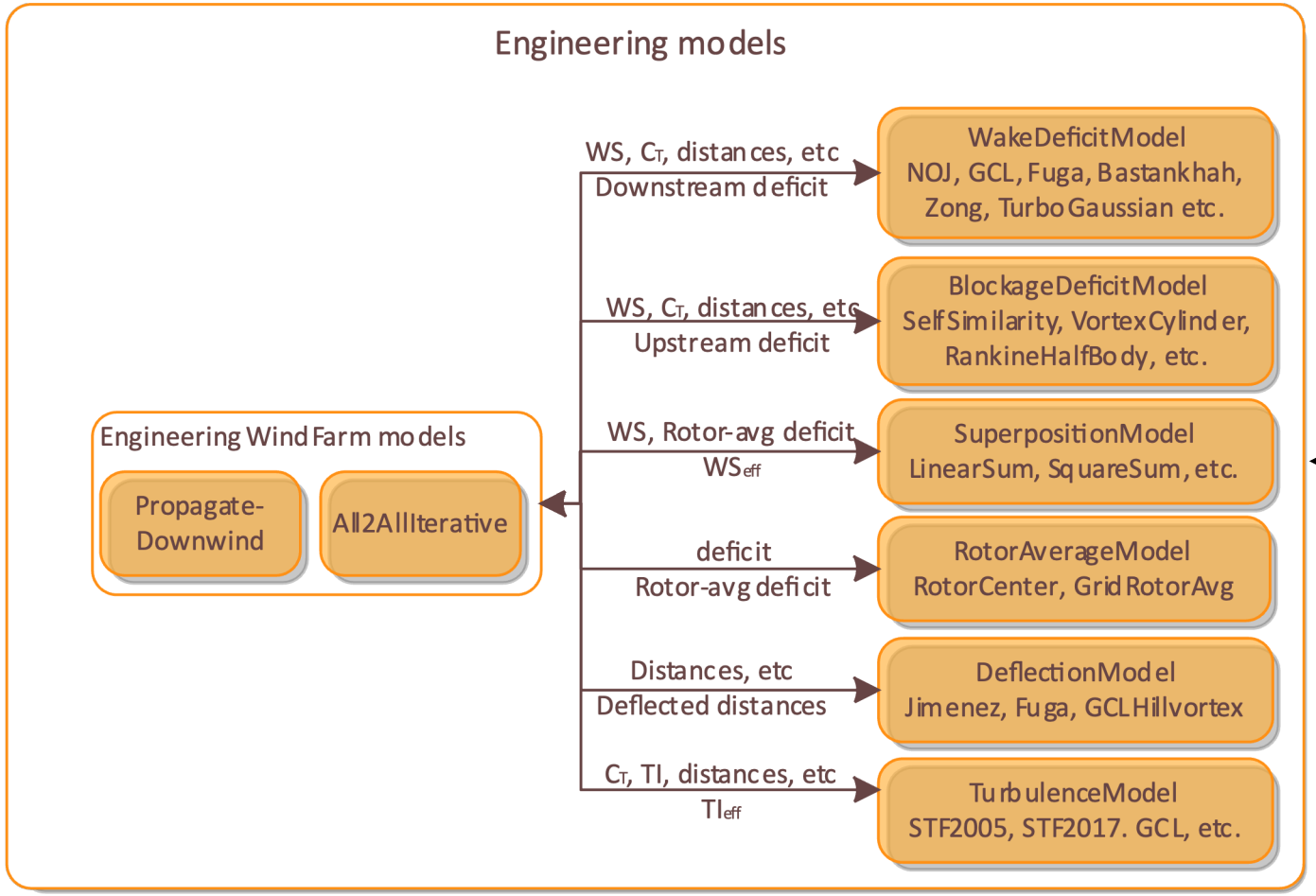


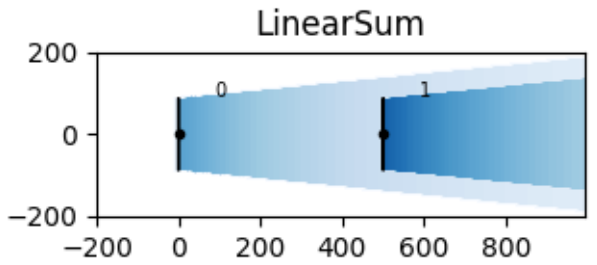
PyWake wake deficit models

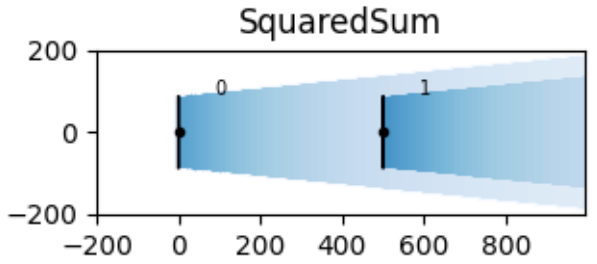


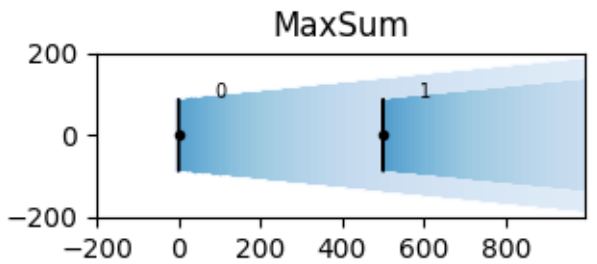


PyWake wake deficit models

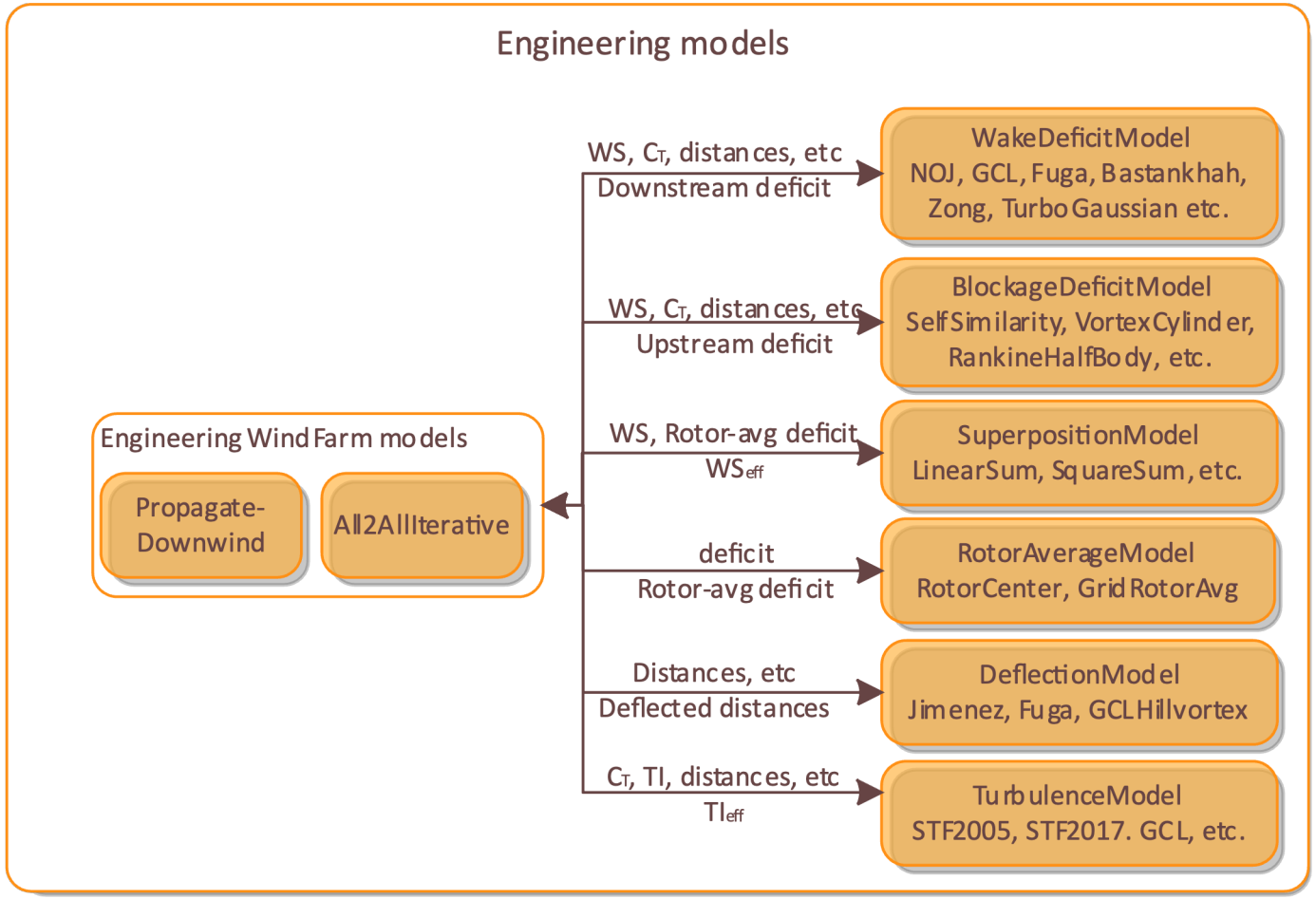


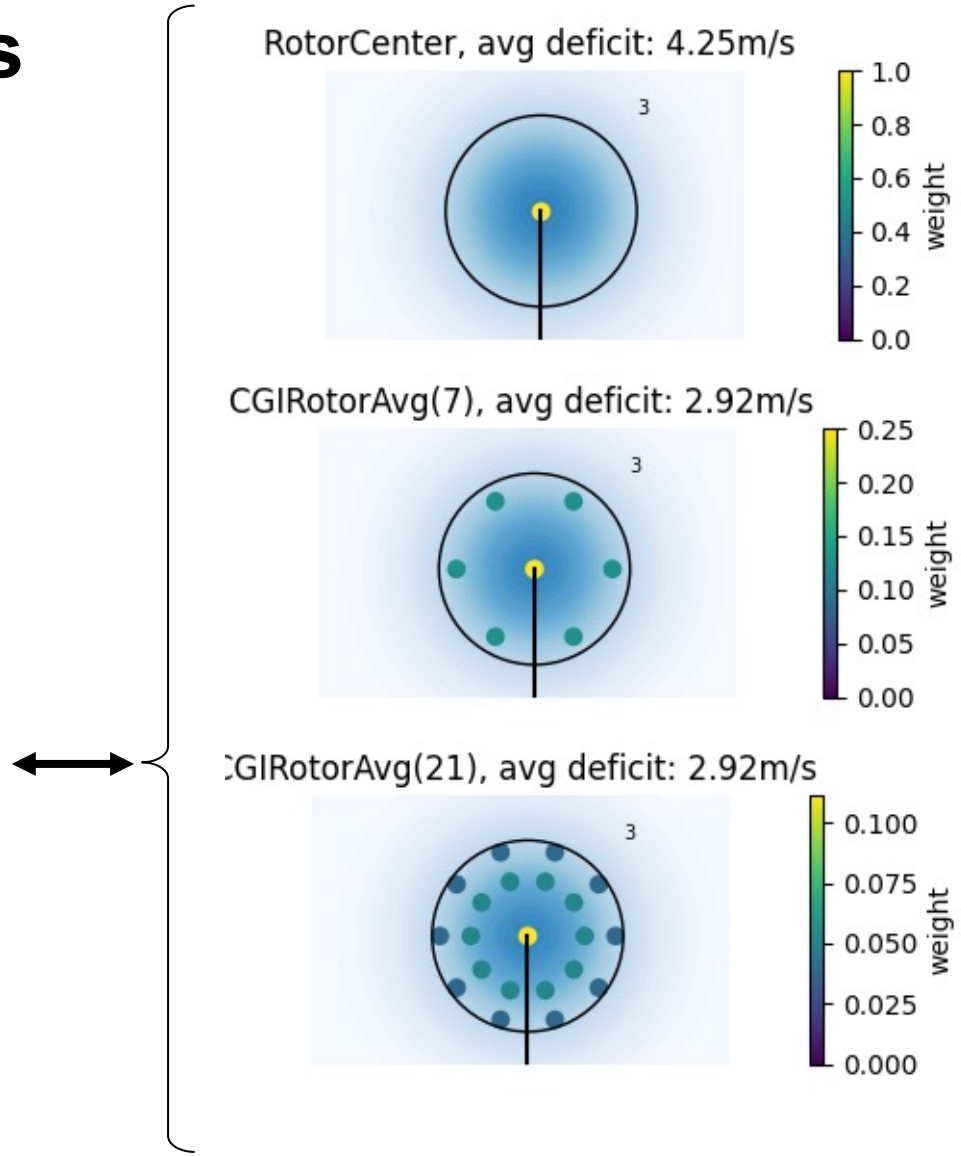




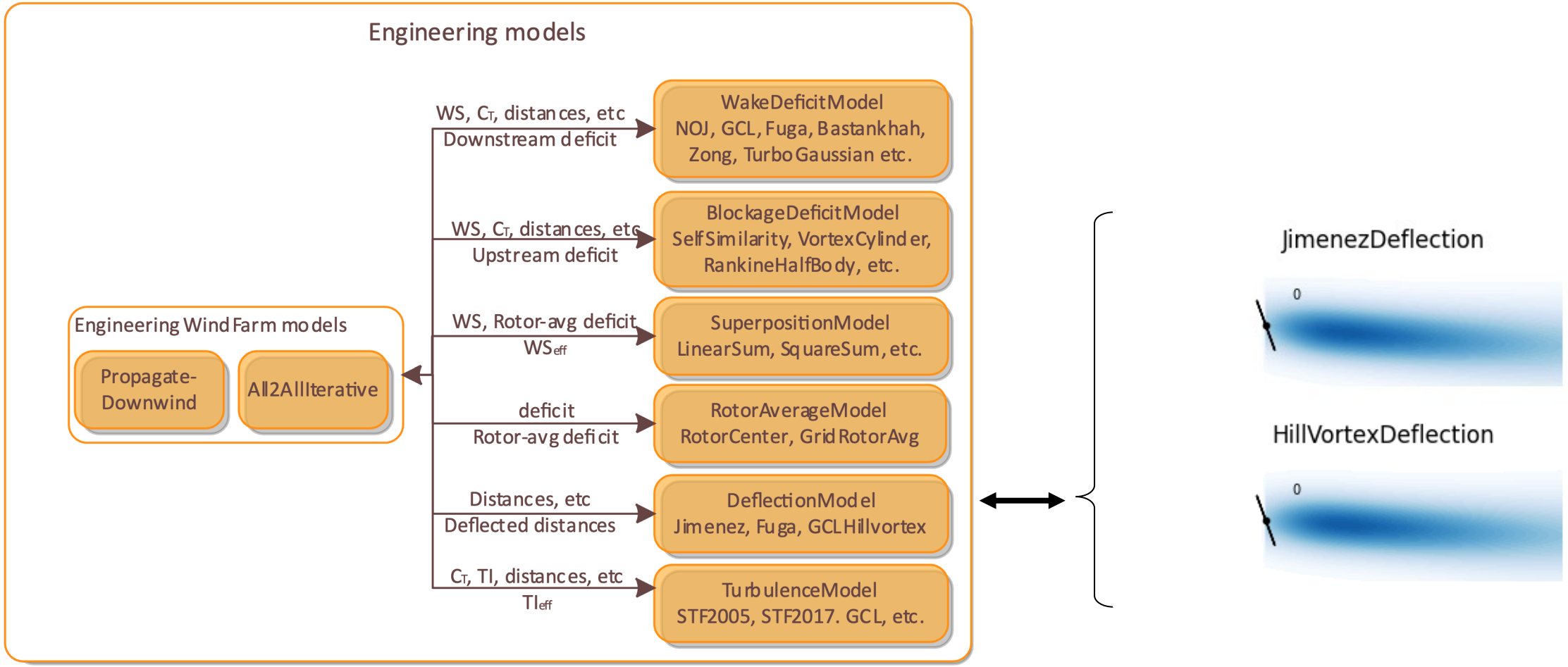


PyWake wake deficit models

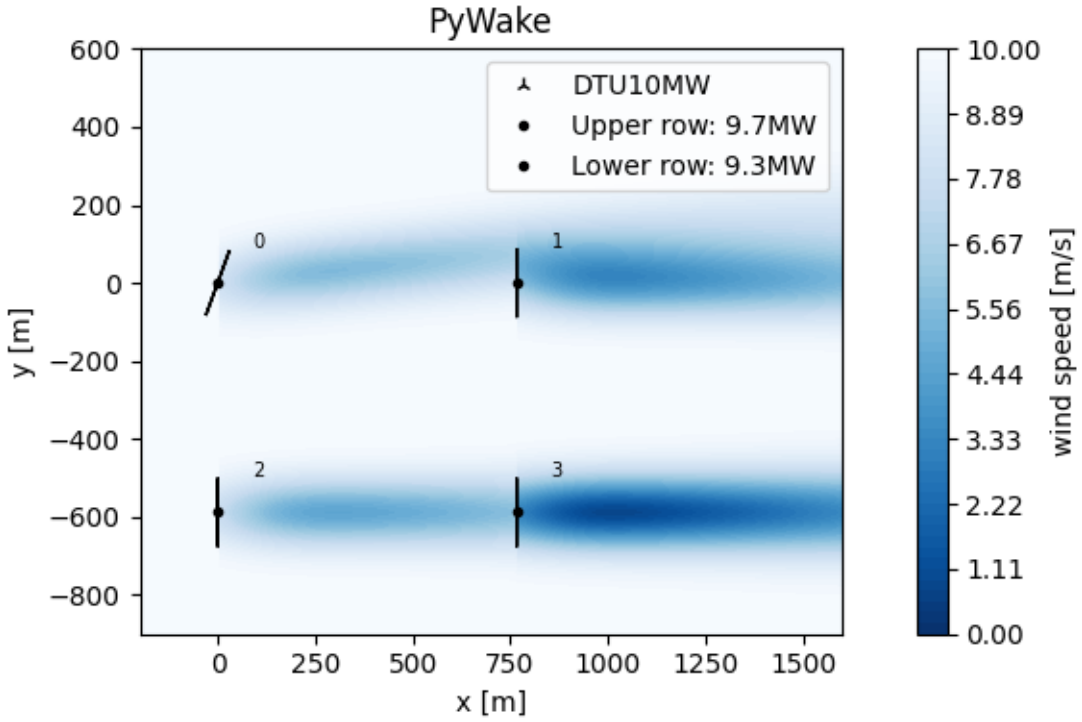




PyWake wake deficit models

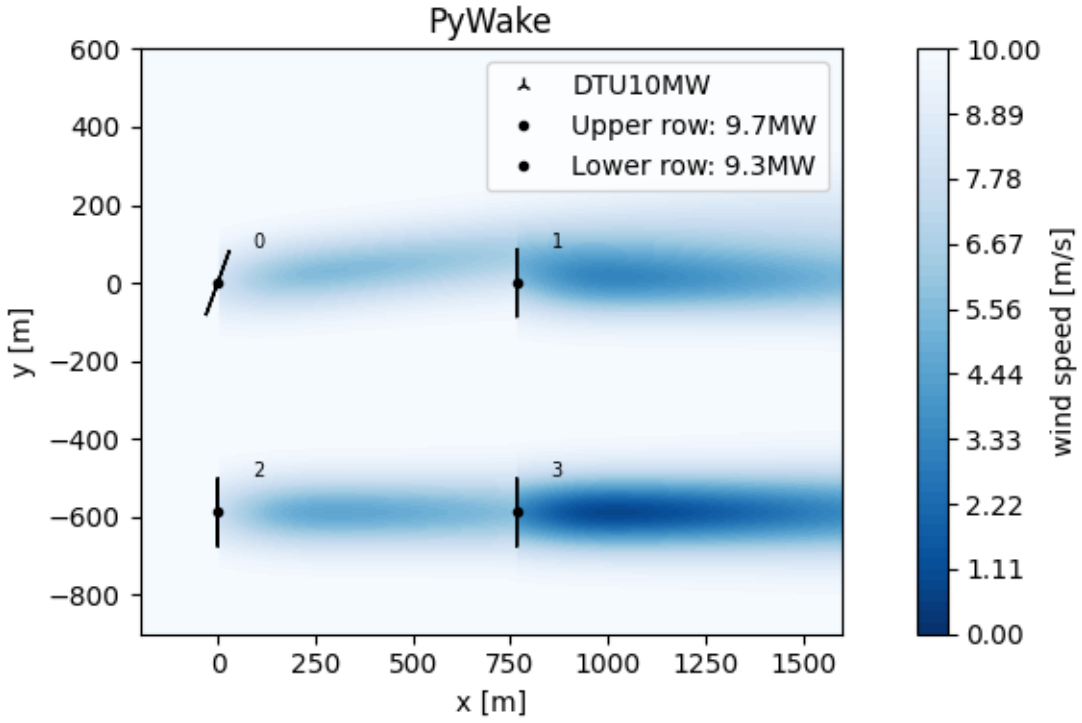


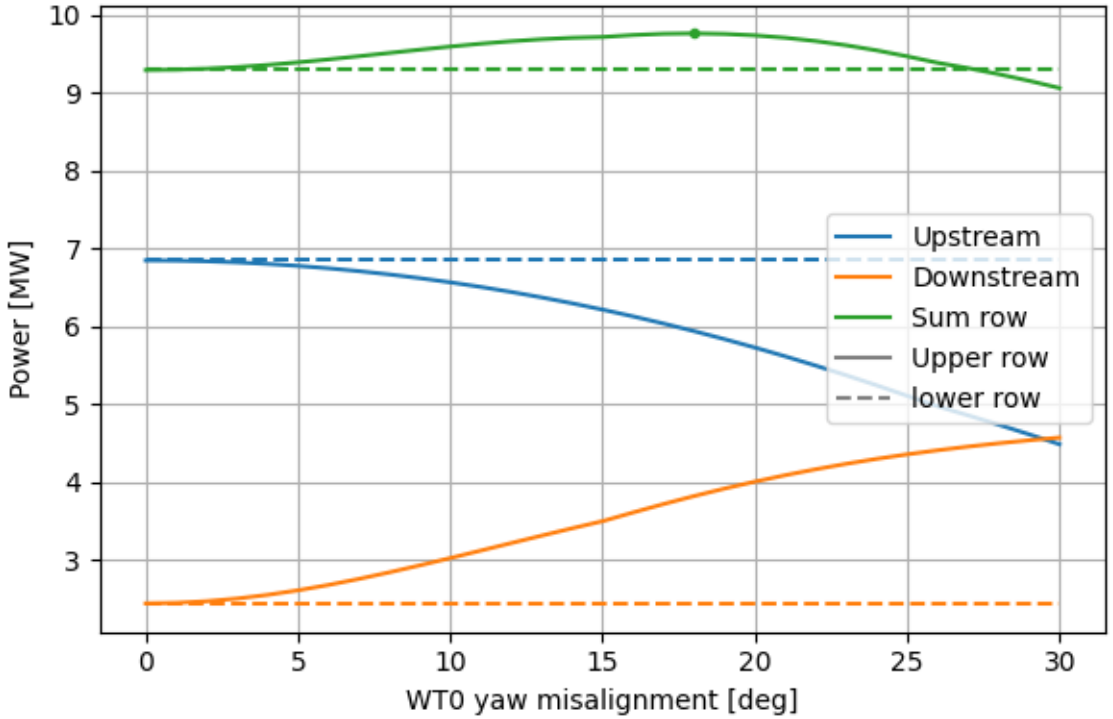
Case study



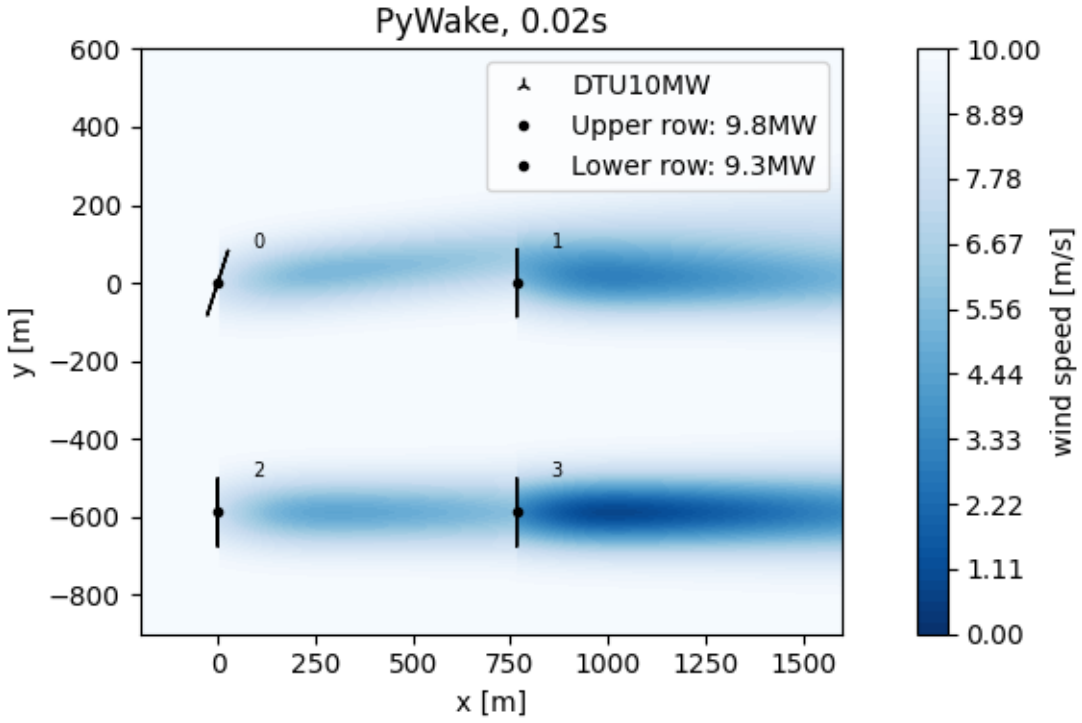
– 2x2 DTU 10 MW wind turbines
  - $Power = power\left(\cos\theta_{yaw} \cdot ws\right)$
  - $CT = ct\left(\cos\theta_{yaw} \cdot ws\right)\cos^2\ \theta_{yaw}$
– 10m/s, 6% turbulence intensity
– Zong gaussian wake deficit
– Linear superposition
– 7 points rotor average
– Jimenez wake deflection($\beta = 0.0268$ ~ 6% TI)

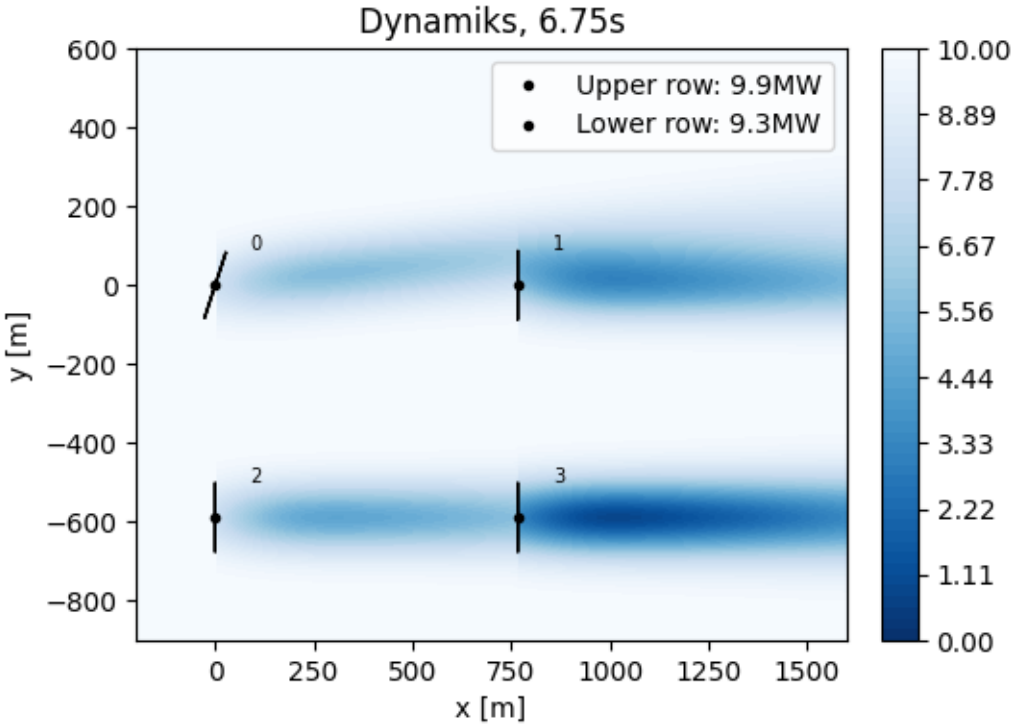
PyWake to Dynamiks



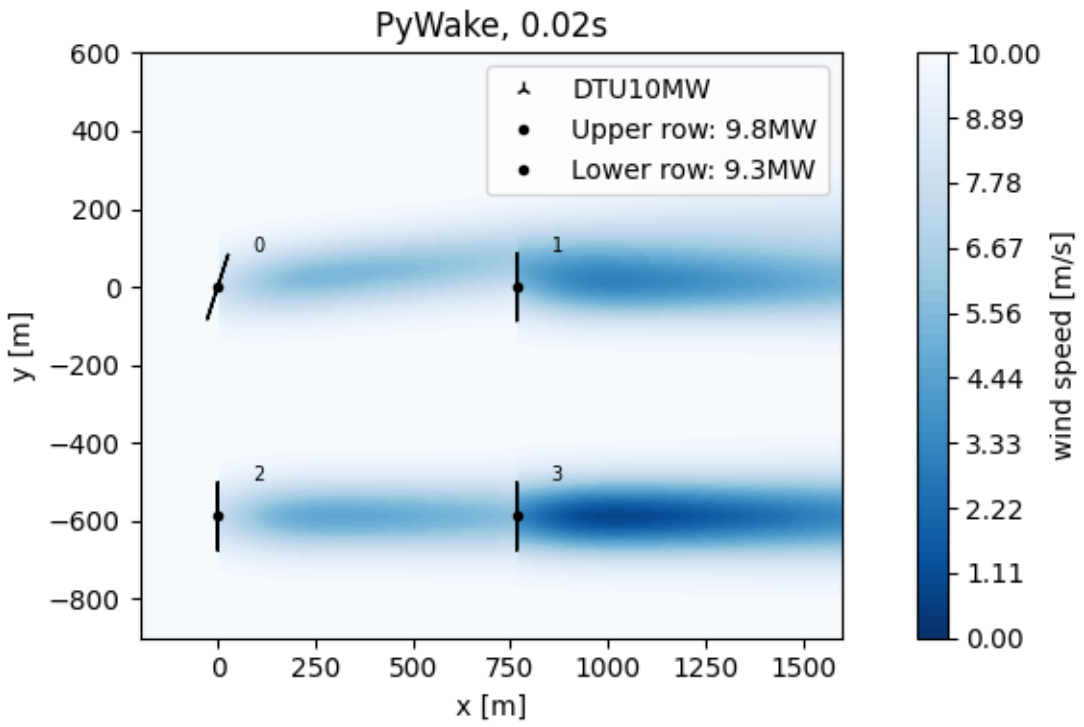


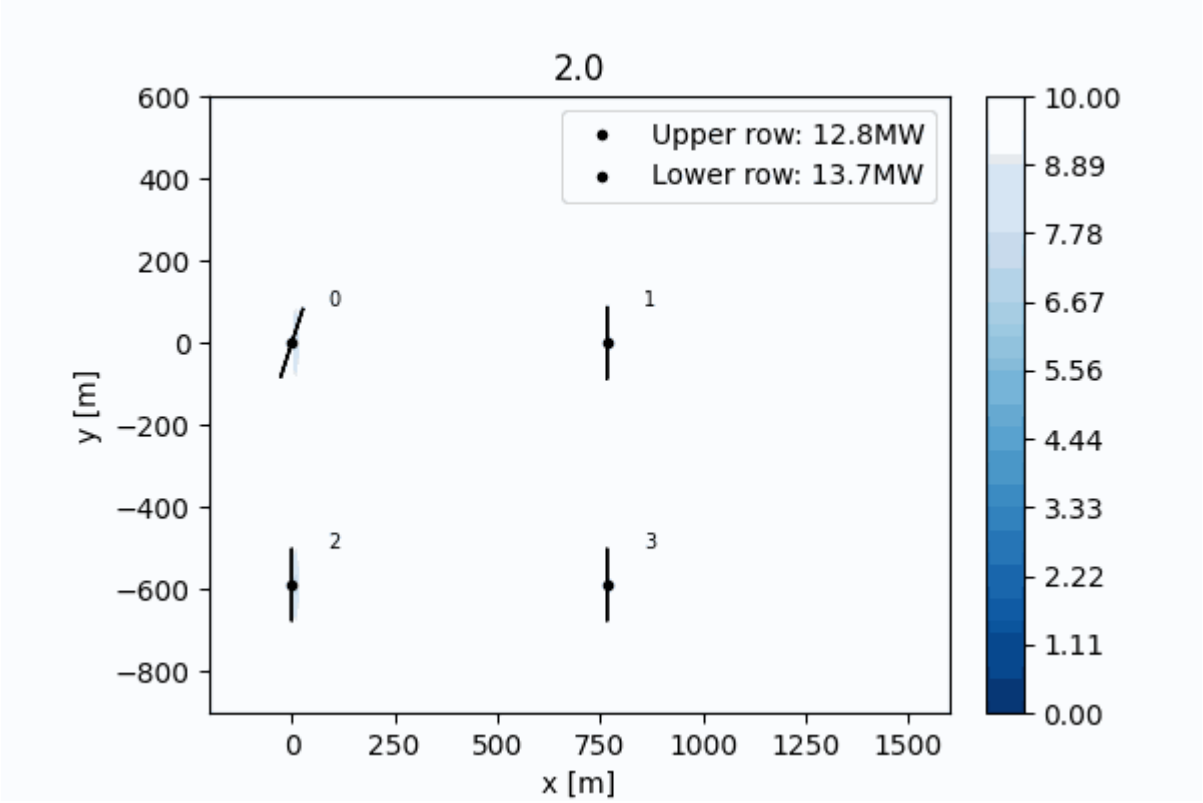
PyWake to Dynamiks





PyWake to Dynamiks





Dynamiks

Dynamiks: Dynamic wind system simulator

Time series	Single blade to multiple wind farms	Inflow, wind turbines, wakes

- Open source python framework from DTU
  – Interfaces HAWC2, EllipSys3D (commercial software, requires licenses)
- User friendly and easy to install
- Modular, flexible and multi fidelity

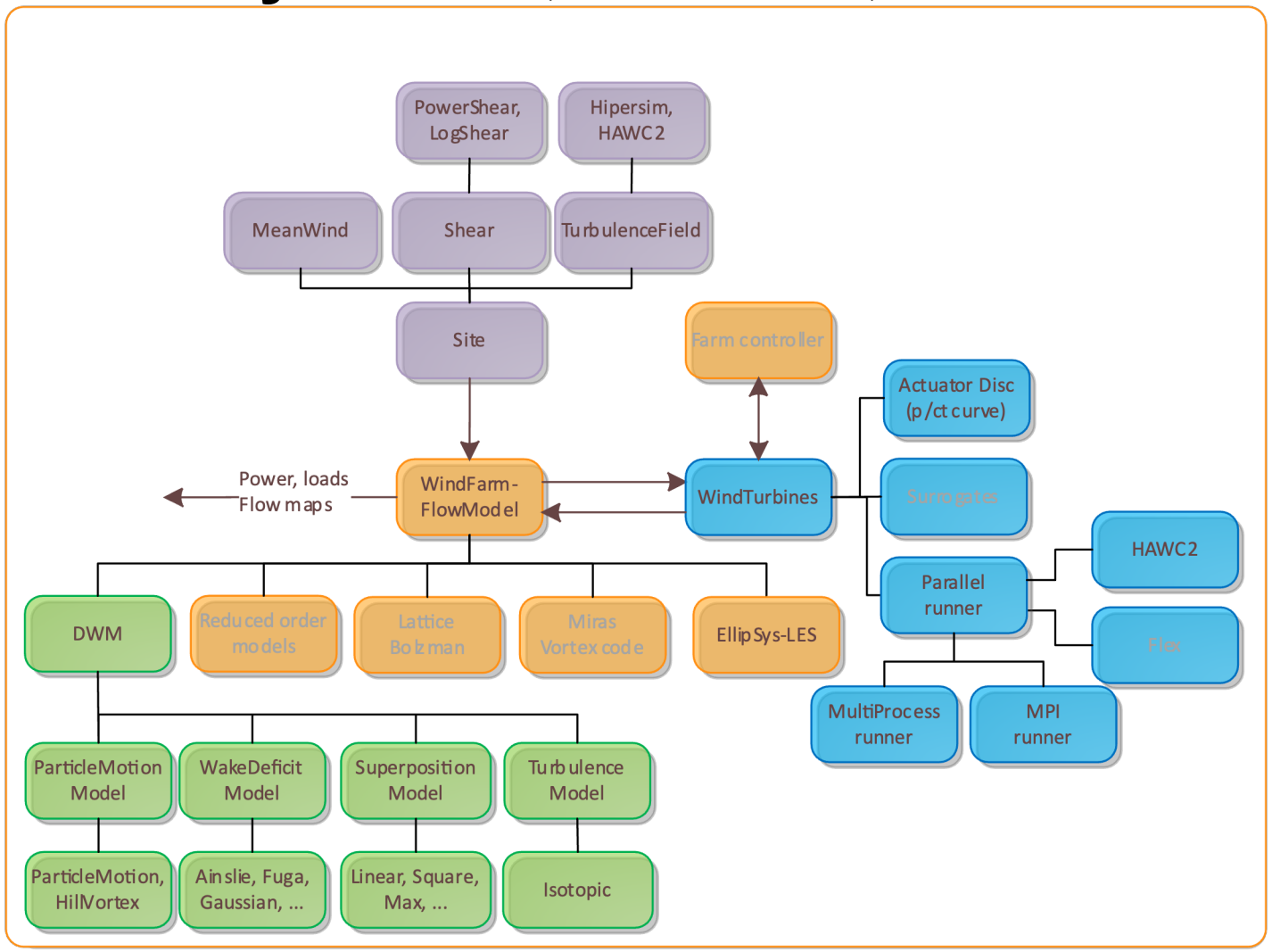
Dynamiks

Dynamiks: Dynamic wind system simulator

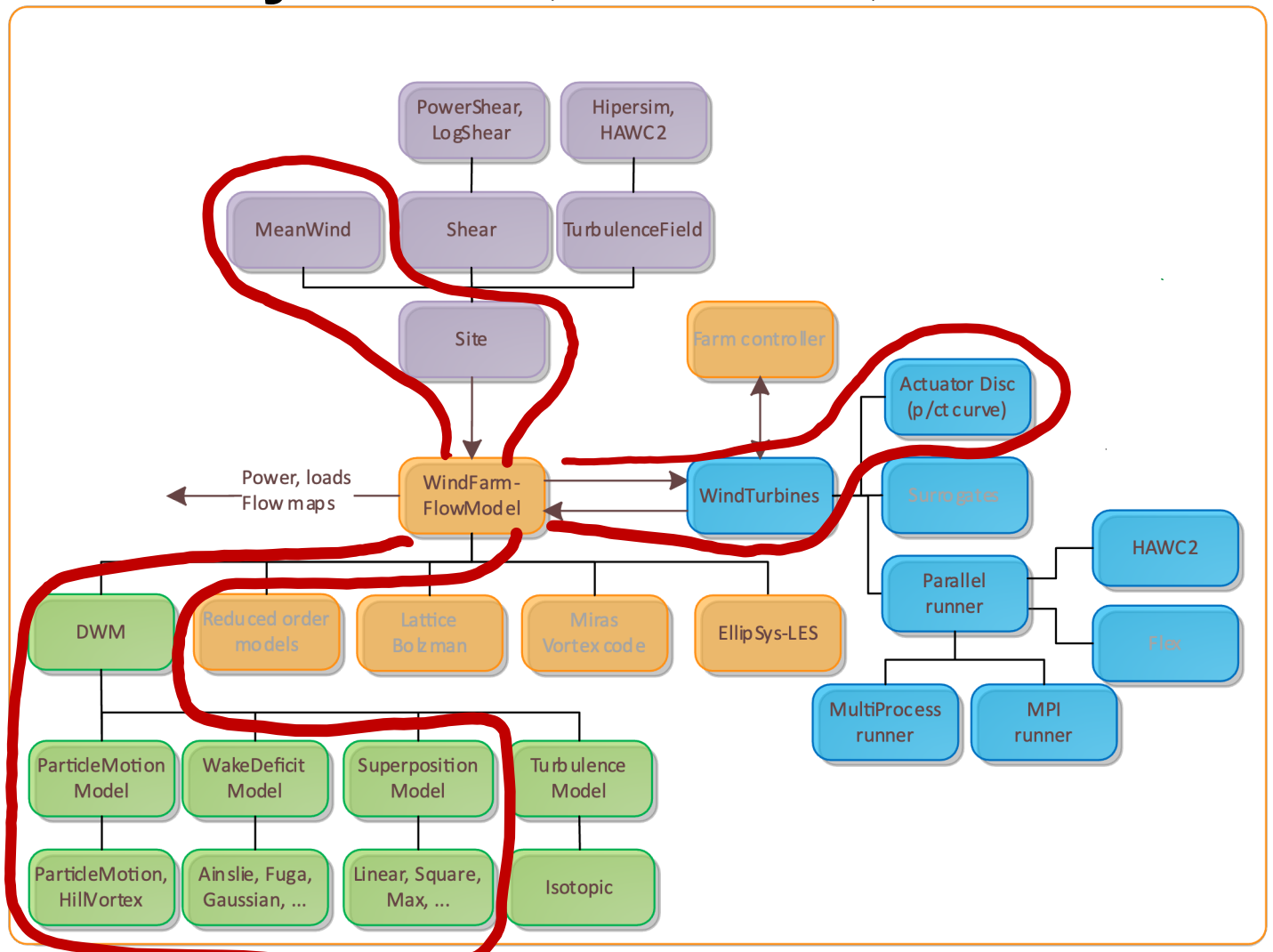
Time series	Single blade to multiple wind farms	Inflow, wind turbines, wakes

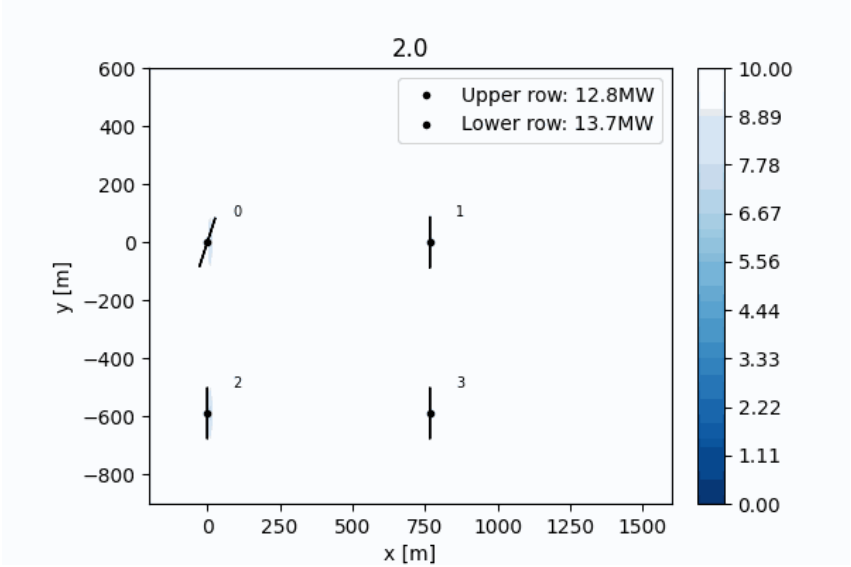
- Simulate
  - Extreme and fatigue loads of wind farms
  - Changing inflow (wind speed and direction)
  - Turbine control (start-up/shut-down, etc)
  - Wind farm control (derating, yaw to optimize total farm production)
- Validation of low-fidelity models against high-fidelity models

Dynamiks, modular, flexible and multifidelity



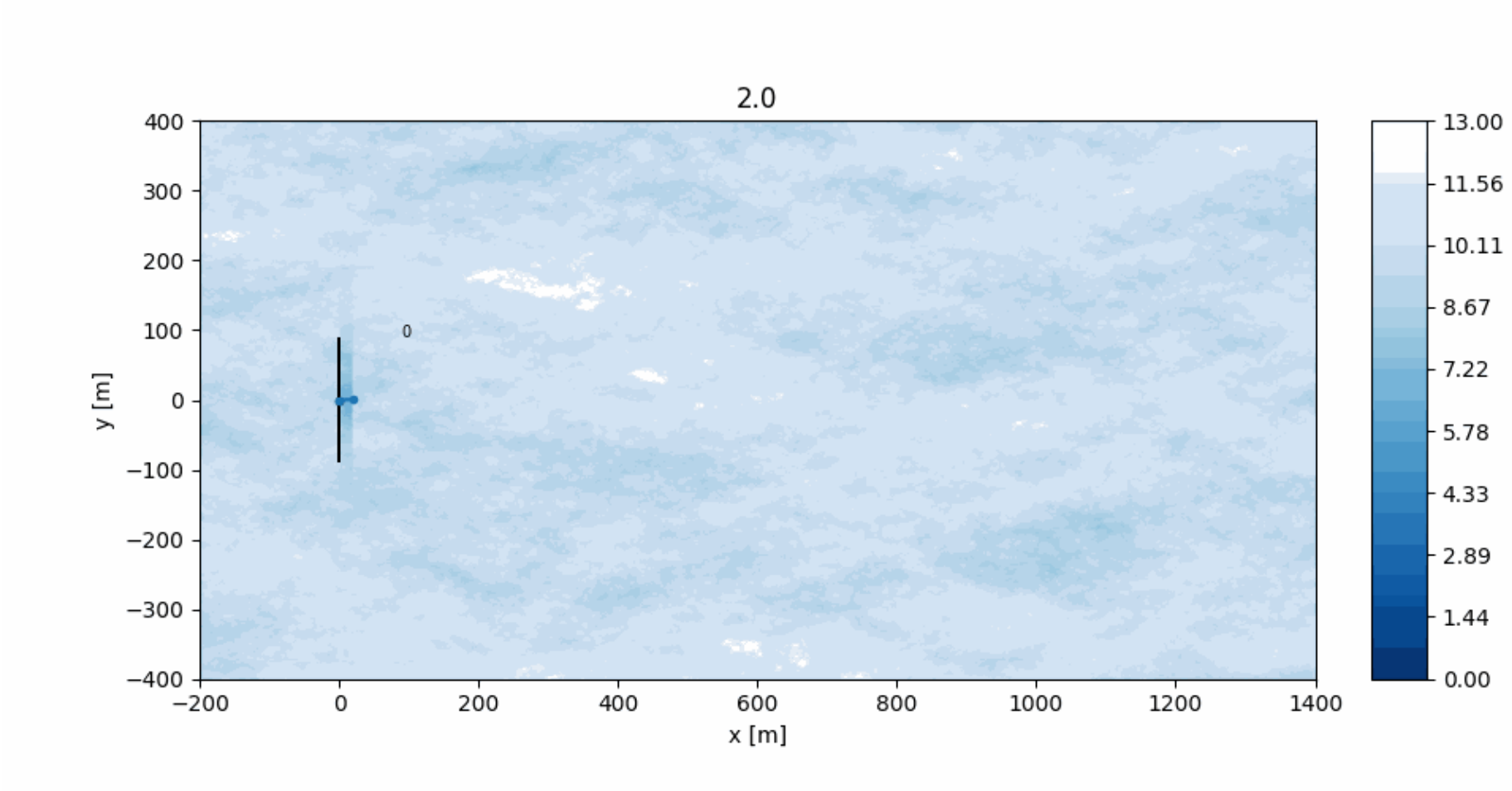
Dynamiks, modular, flexible and multifidelity



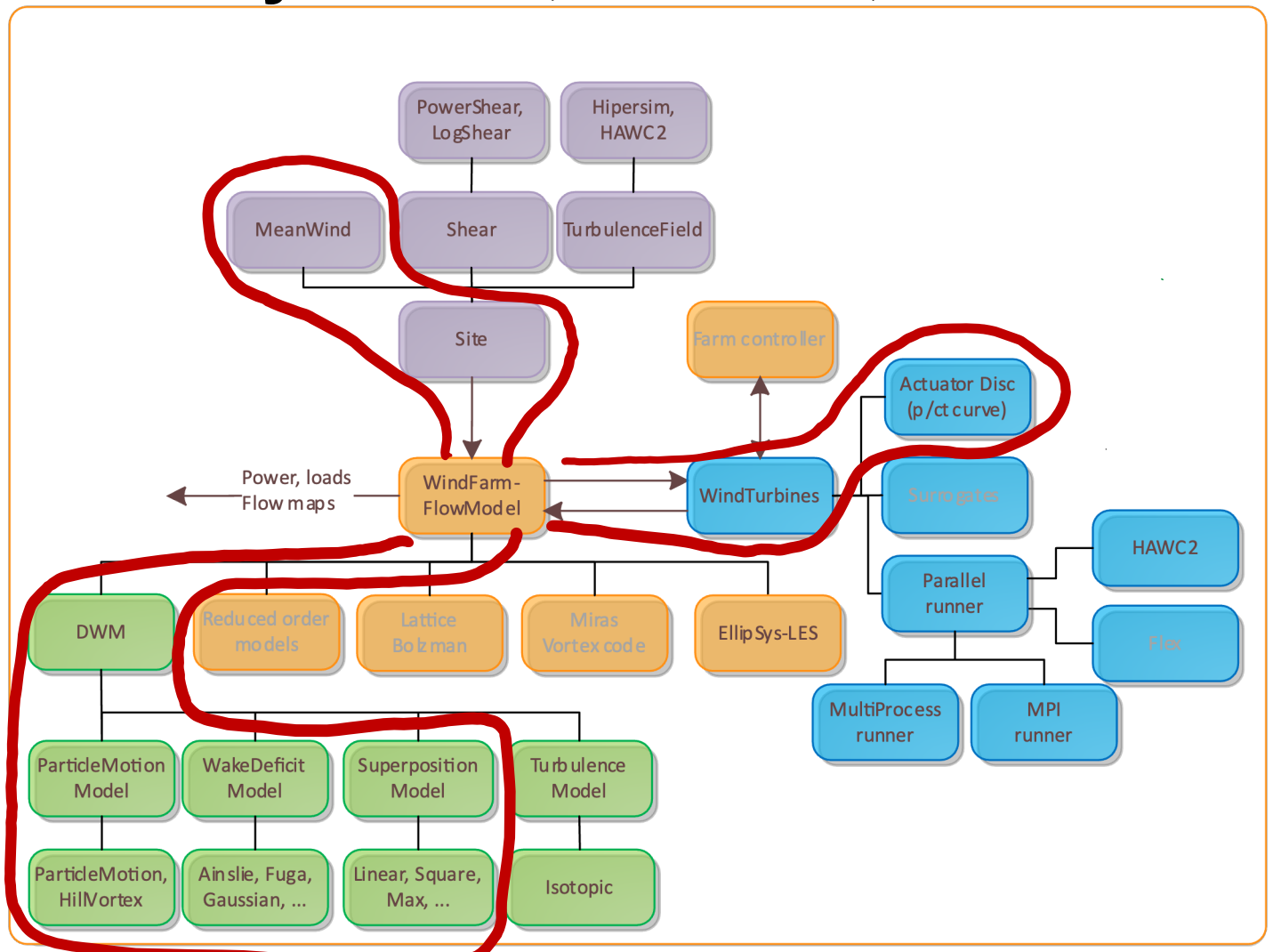


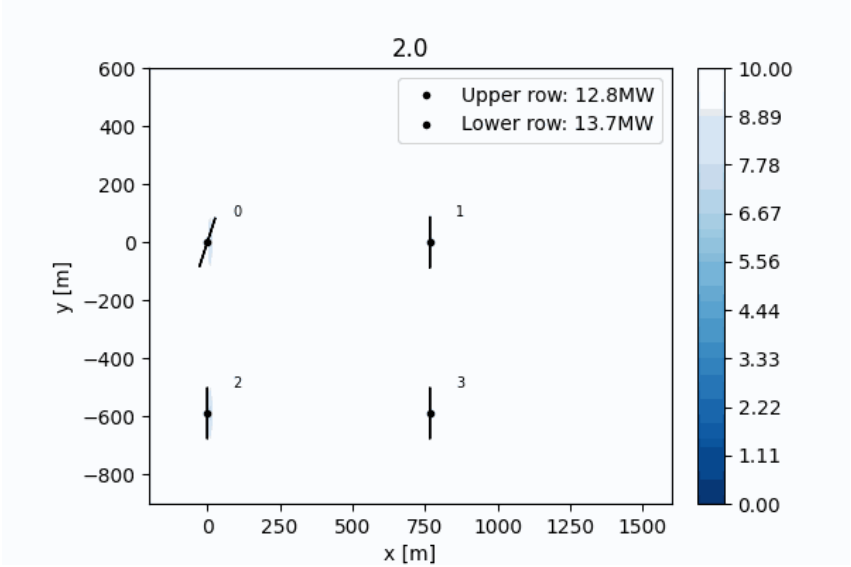

```
site = UniformSite(ws=10, ti=.06)
wts = PyWakeWindTurbines(x=wt_x, y=wt_y,
                         windTurbine=DTU10MW())
wts.yaw = [-18, 0, 0, 0]

fs = DWMFlowSimulation(
    site=site,
    windTurbines=wts,
    ...)
fs.animate(250, view=...)
```

Dynamic Wake Meandering model (DWM)



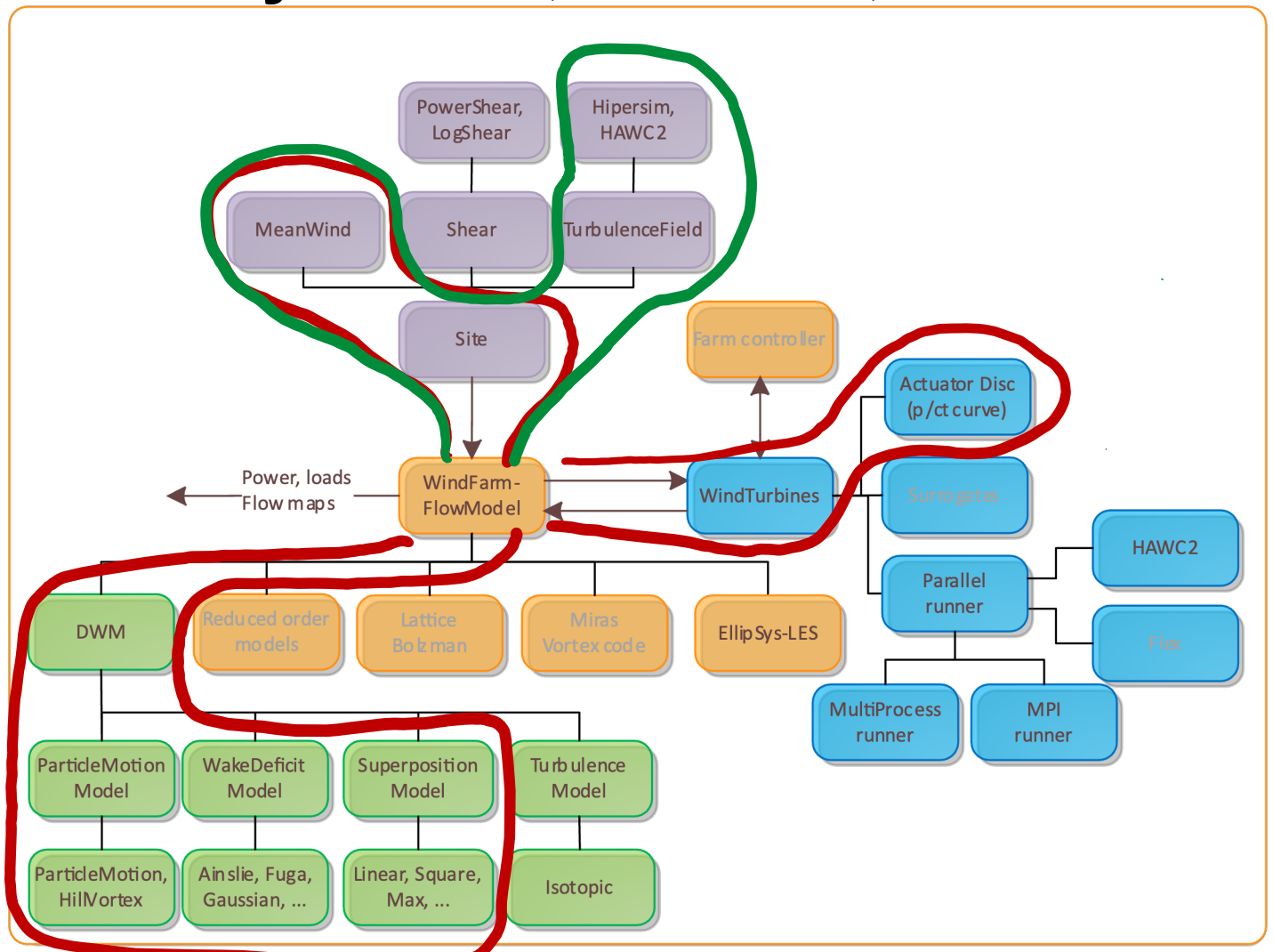
Dynamiks, modular, flexible and multifidelity

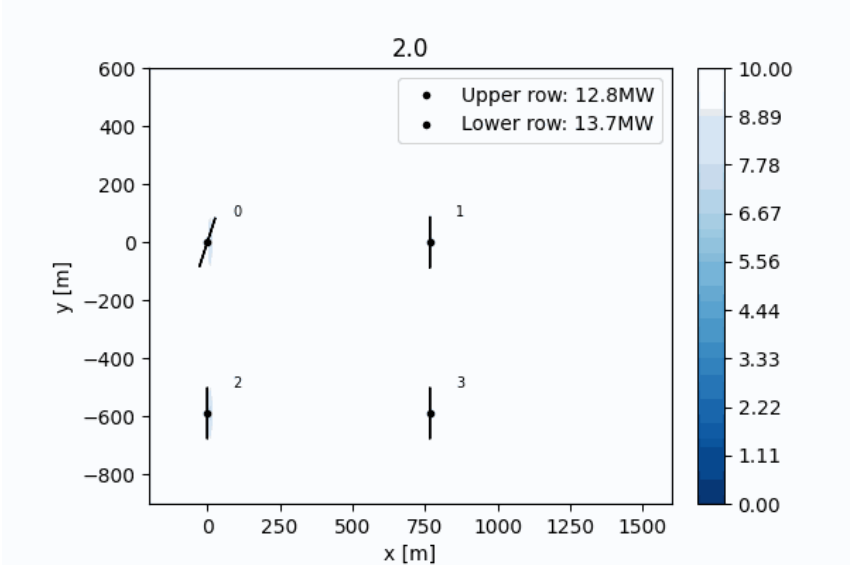





```
site = UniformSite(ws=10, ti=.06)
wts = PyWakeWindTurbines(x=wt_x, y=wt_y,
                         windTurbine=DTU10MW())
wts.yaw = [-18, 0, 0, 0]

fs = DWMFlowSimulation(
    site=site,
    windTurbines=wts,
    ...)
fs.animate(250, view=...)
```

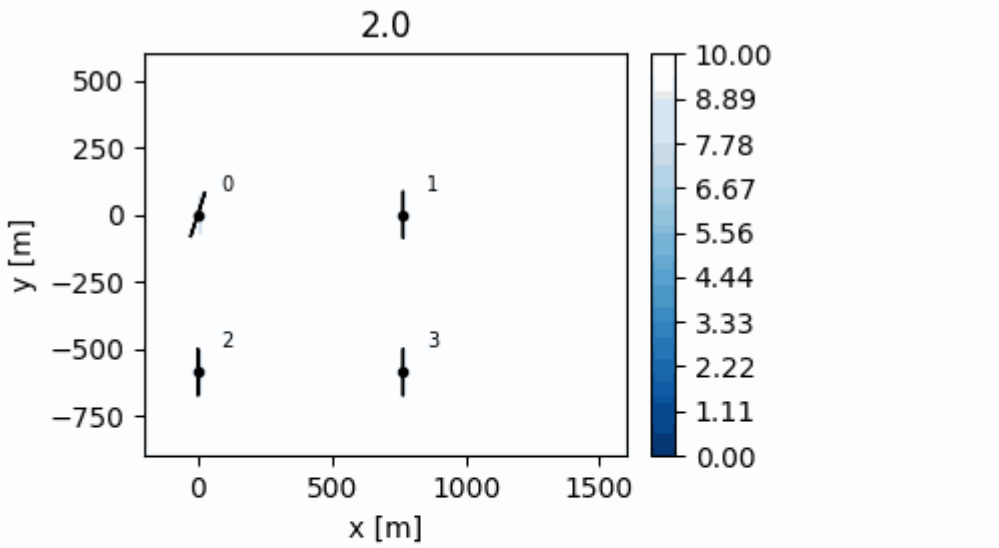
Dynamiks, modular, flexible and multifidelity

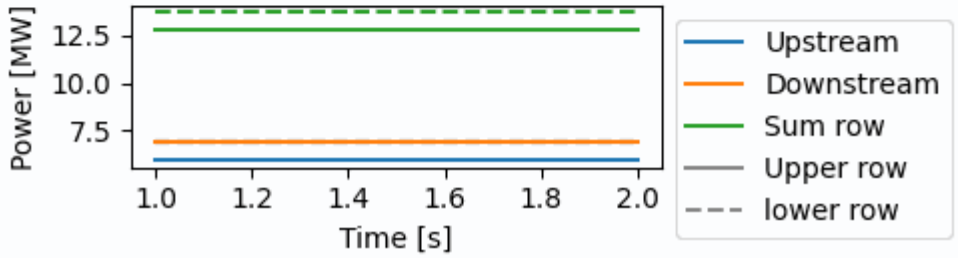




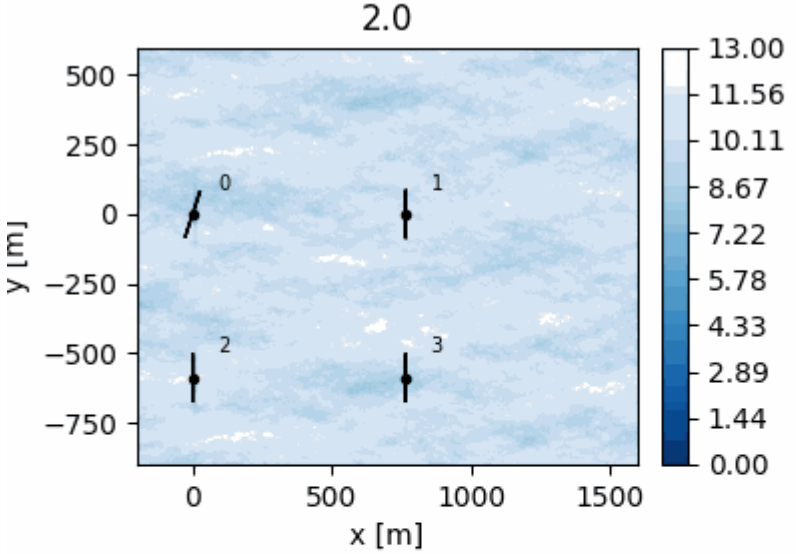

```
site = UniformSite(ws=10, ti=.06)
wts = PyWakeWindTurbines(x=wt_x, y=wt_y,
                         windTurbine=DTU10MW())
wts.yaw = [-18, 0, 0, 0]

fs = DWMFlowSimulation(
    site=site,
    windTurbines=wts,
    ...)
fs.animate(250, view=...)
```

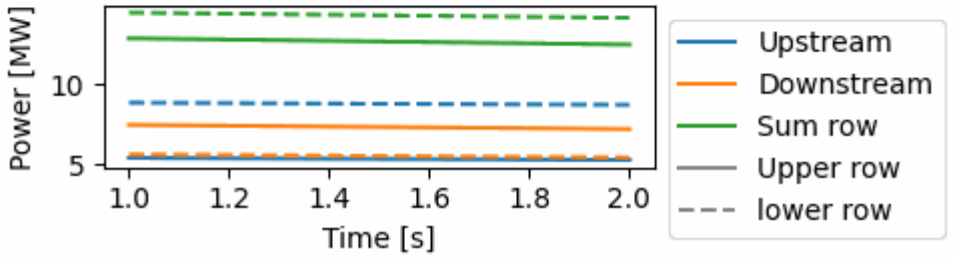
```
site = UniformSite(ws=10, ti=.06)
```



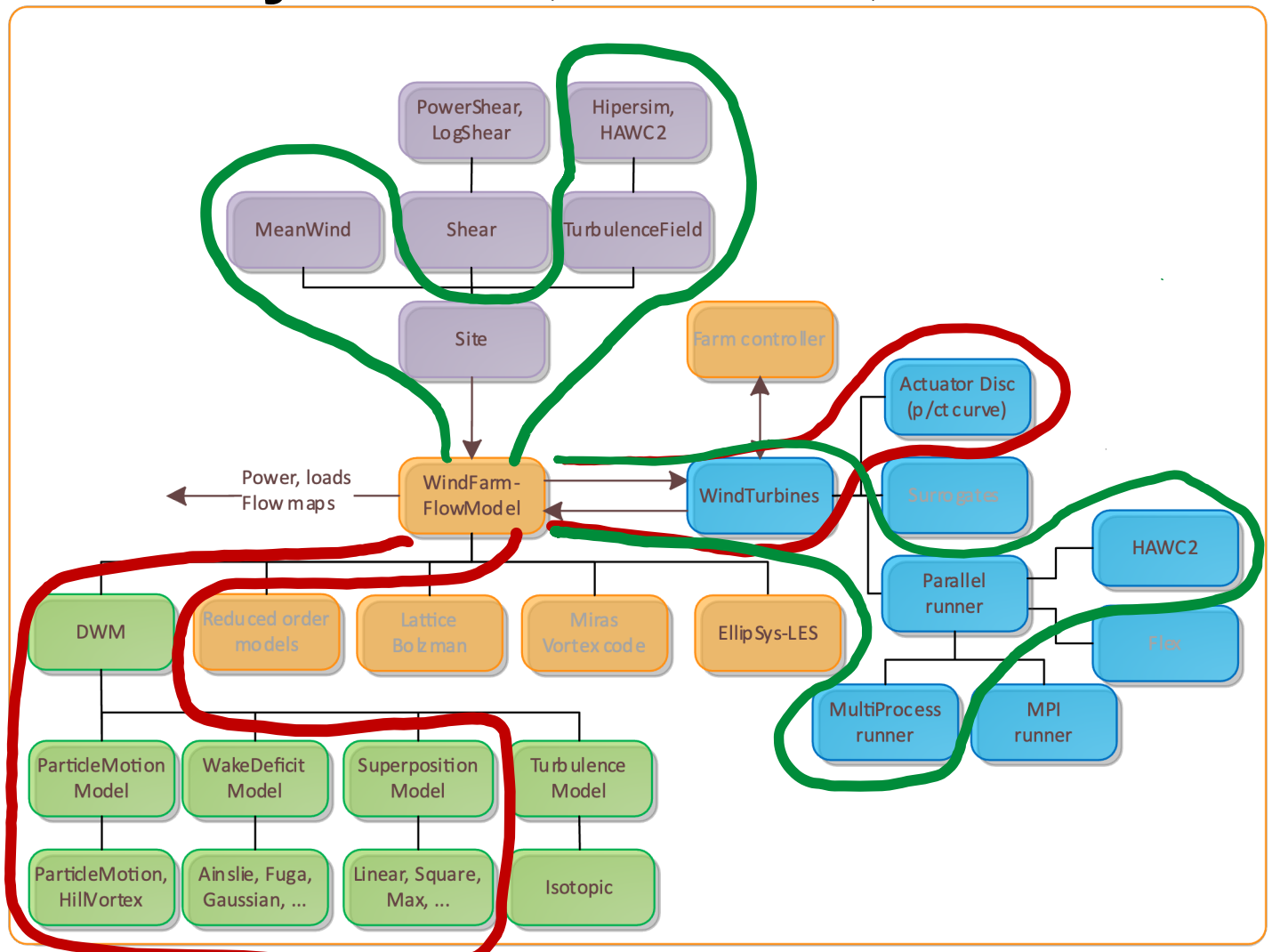



```
tf = MannTurbulenceField.generate(Nxyz, dxyz)
tf.scale_TI(TI=.06, U=10)
site = TurbulenceFieldSite(ws=10, turbulenceField=tf)
```

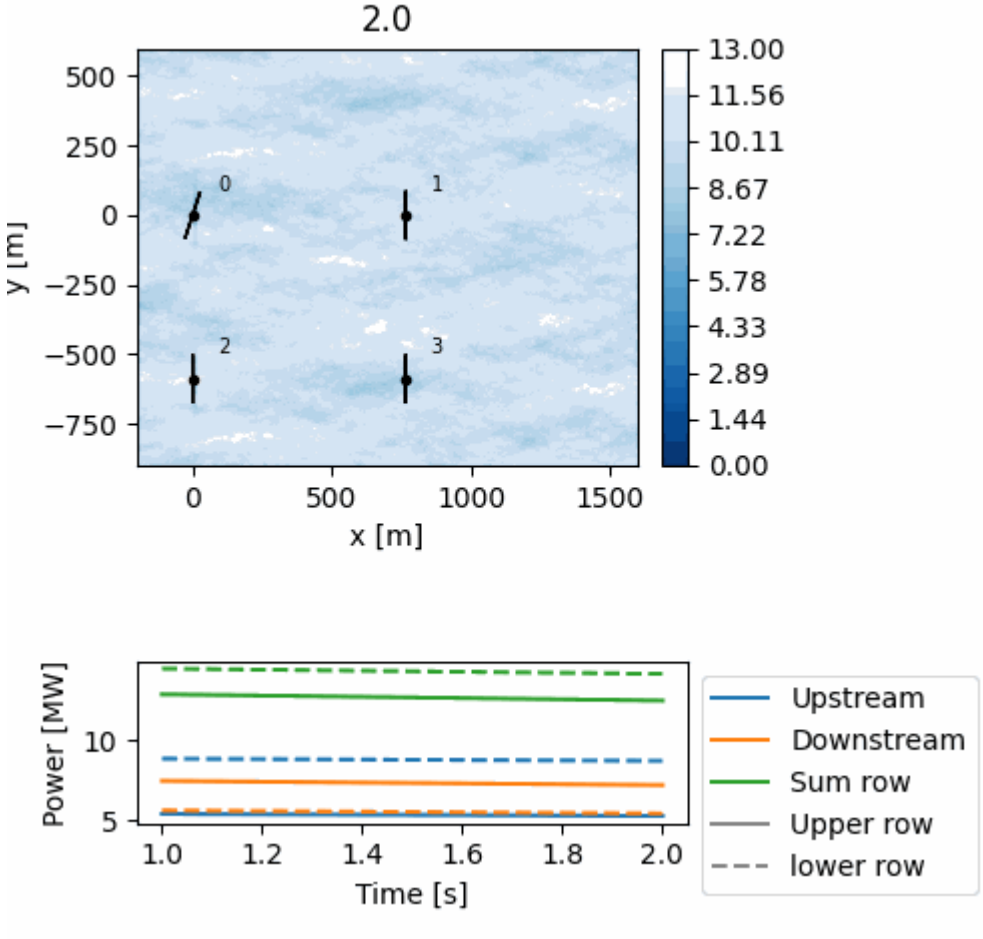




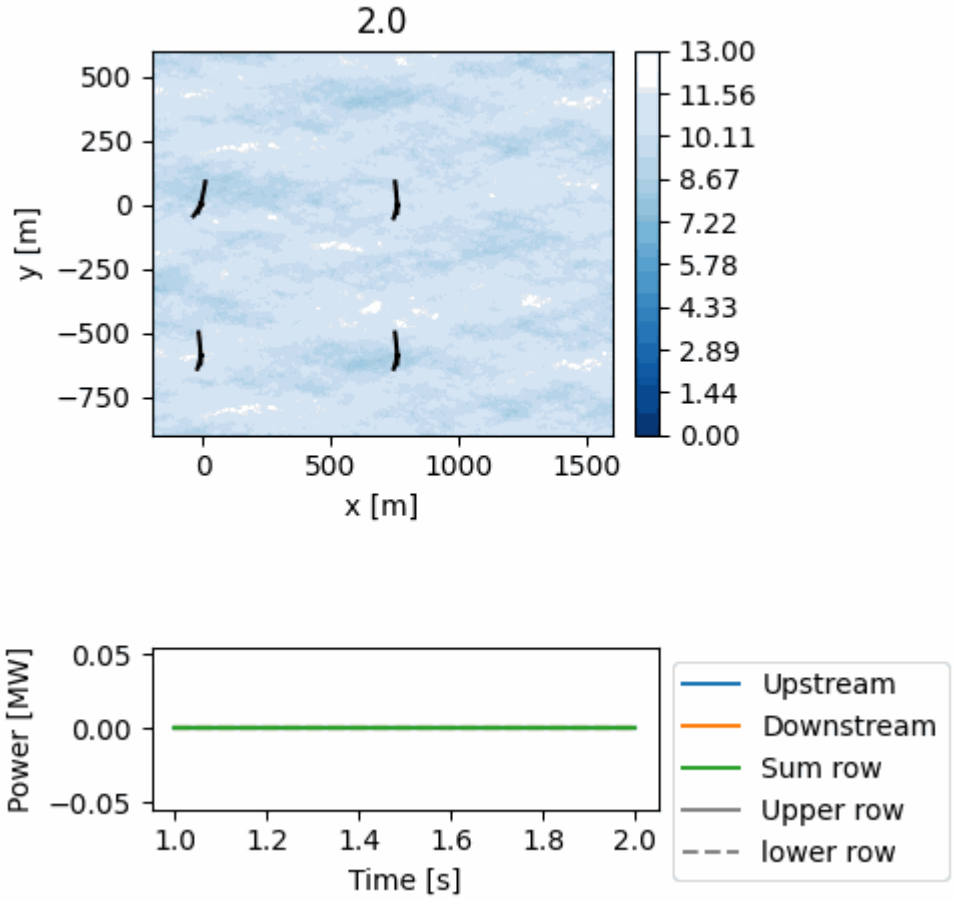
Dynamiks, modular, flexible and multifidelity



```
wts = PyWakeWindTurbines(
    x=wt_x, y=wt_y,
    windTurbine=DTU10MW())
wts.yaw = [-18,0,0,0]
```




```
wts = HAWC2WindTurbines(
    x=wt_x, y=wt_y,
    htc_filename_lst=['htc/DTU_10MW_RWT_yaw-18.htc',
                      'htc/DTU_10MW_RWT.htc'],
    types=[0,1,1,1])
```



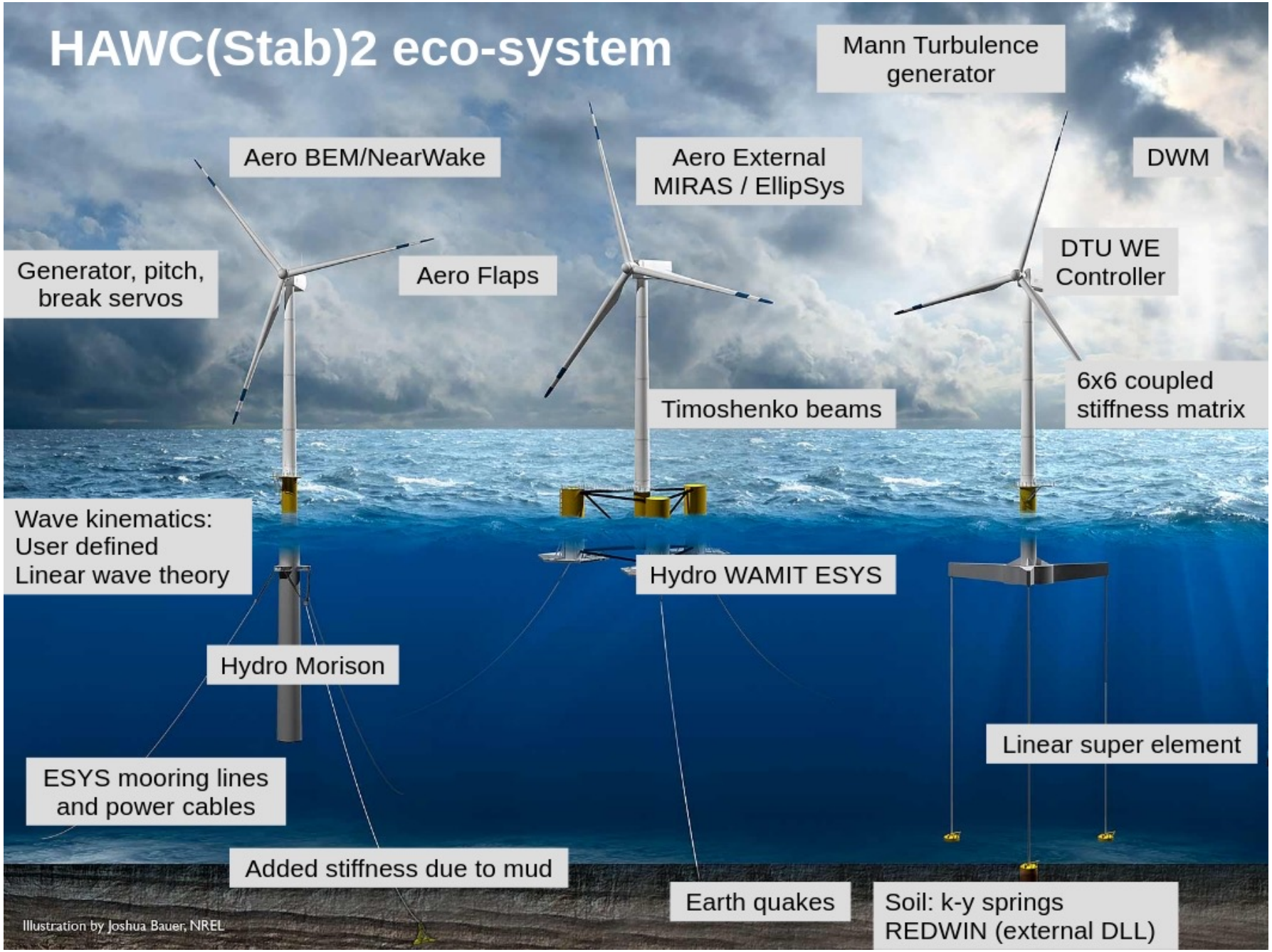
HAWC2
General non-linear time domain response

DTU WE Controller (DLLs)

External Systems (ESYS)

3D Visualisation

Pre- and post-processing



HAWC2:
Examples and applications

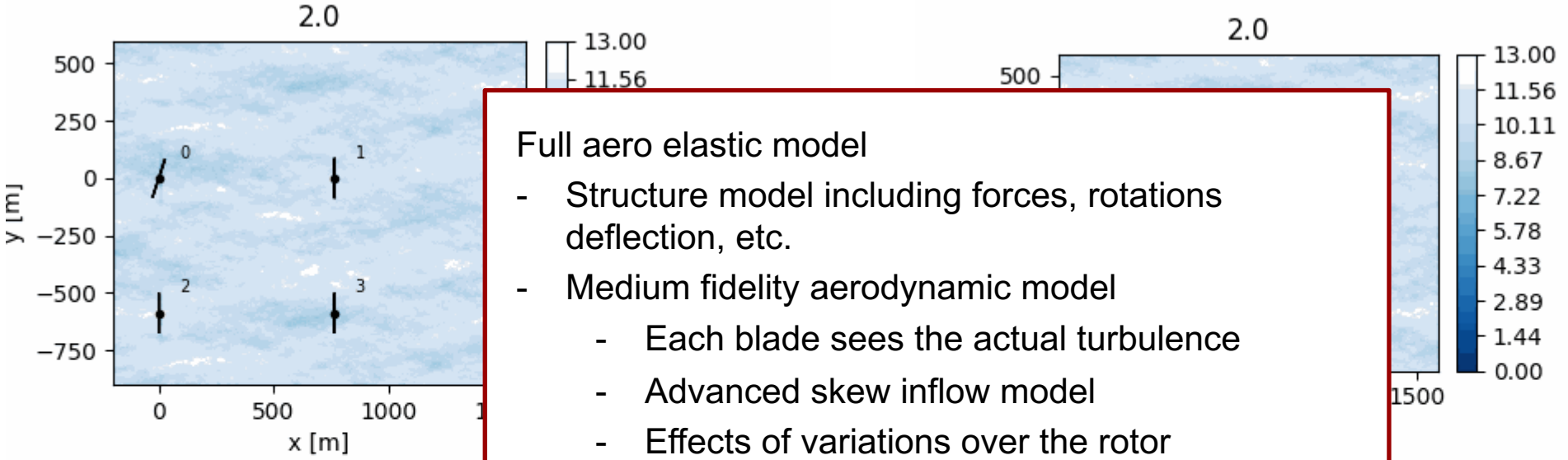
MINGYANG SMART ENERGY
明阳智能
地蕴天成·能动无限

Vestas

aerodyn engineering gmbh

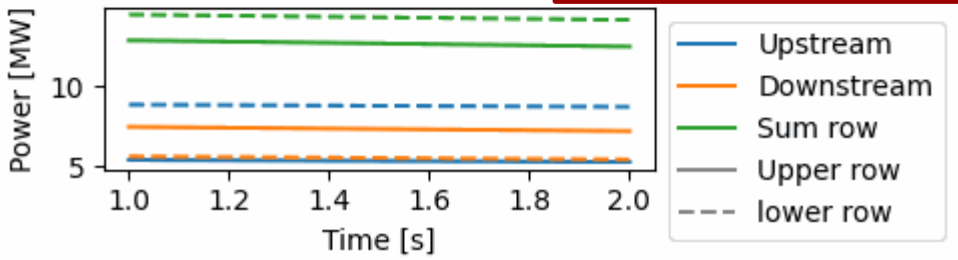
```
wts = PyWakeWindTurbines(
    x=wt_x, y=wt_y,
    windTurbine=DTU10MW())
wts.yaw = [-17,0,0,0]
```

```
wts = HAWC2WindTurbines(
    x=wt_x, y=wt_y,
    htc_filename_lst=['htc/DTU_10MW_RWT_yaw-17.htc',
                      'DTU_10_MW/htc/DTU_10MW_RWT.htc'],
```

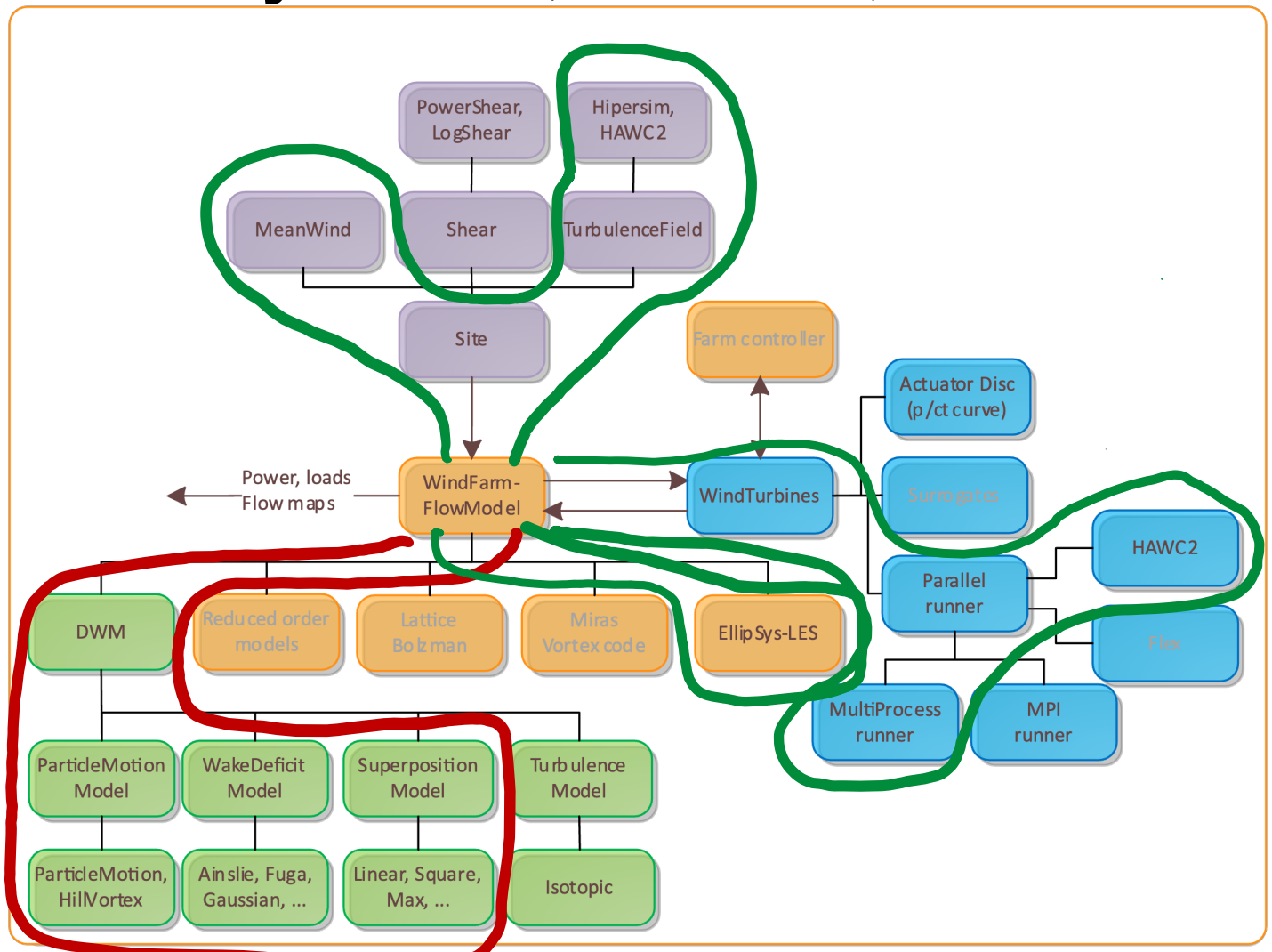


Full aero elastic model

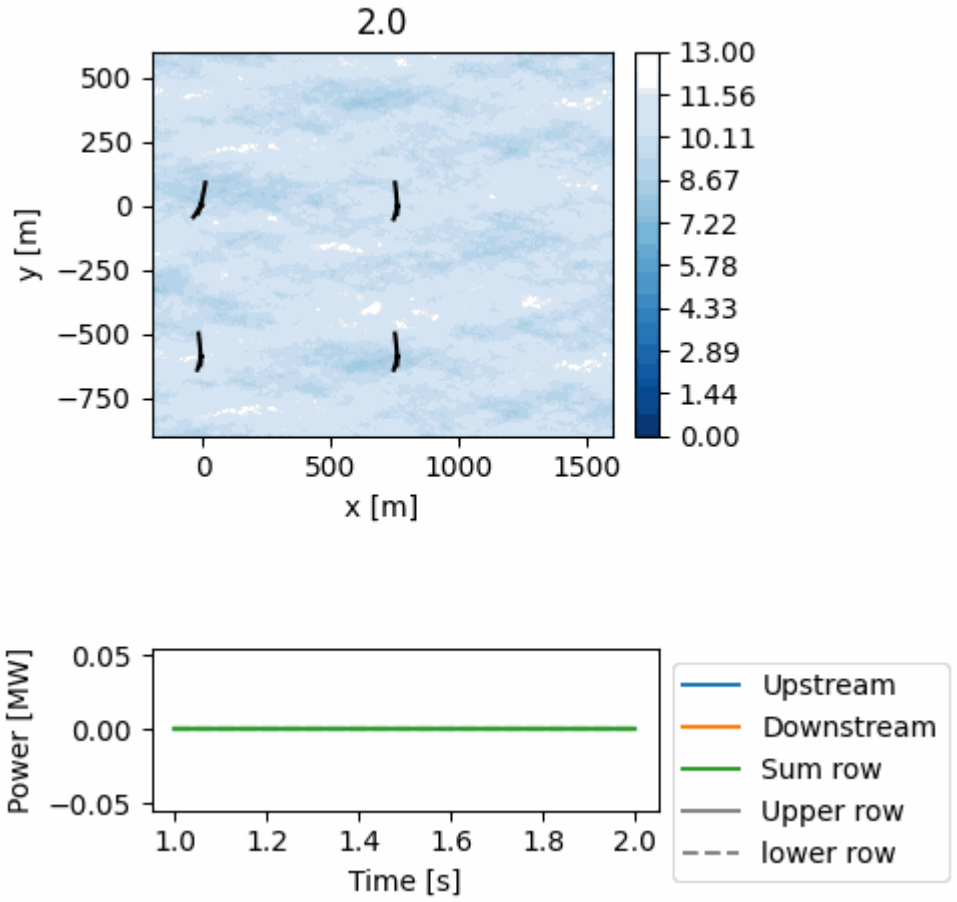
- Structure model including forces, rotations deflection, etc.
- Medium fidelity aerodynamic model
  - Each blade sees the actual turbulence
  - Advanced skew inflow model
  - Effects of variations over the rotor
- Load evaluation



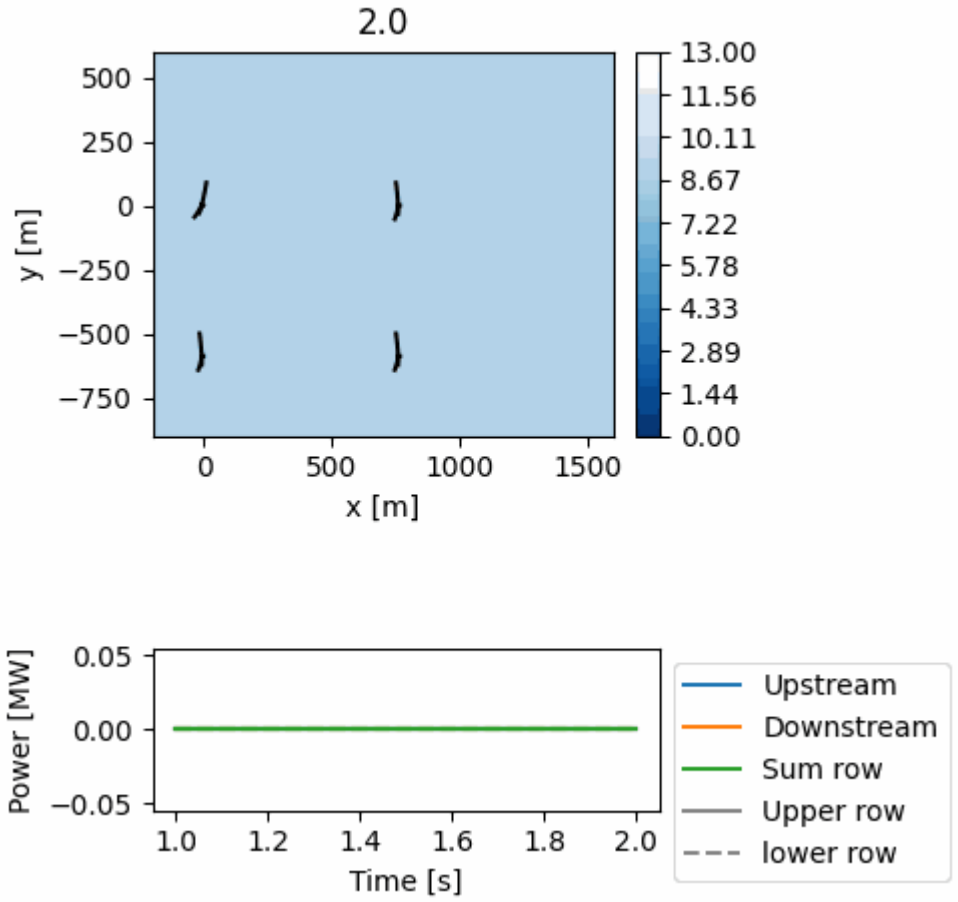
Dynamiks, modular, flexible and multifidelity



```
fs = DWMFlowSimulation(
    windTurbines=wts,
    site=site)
```

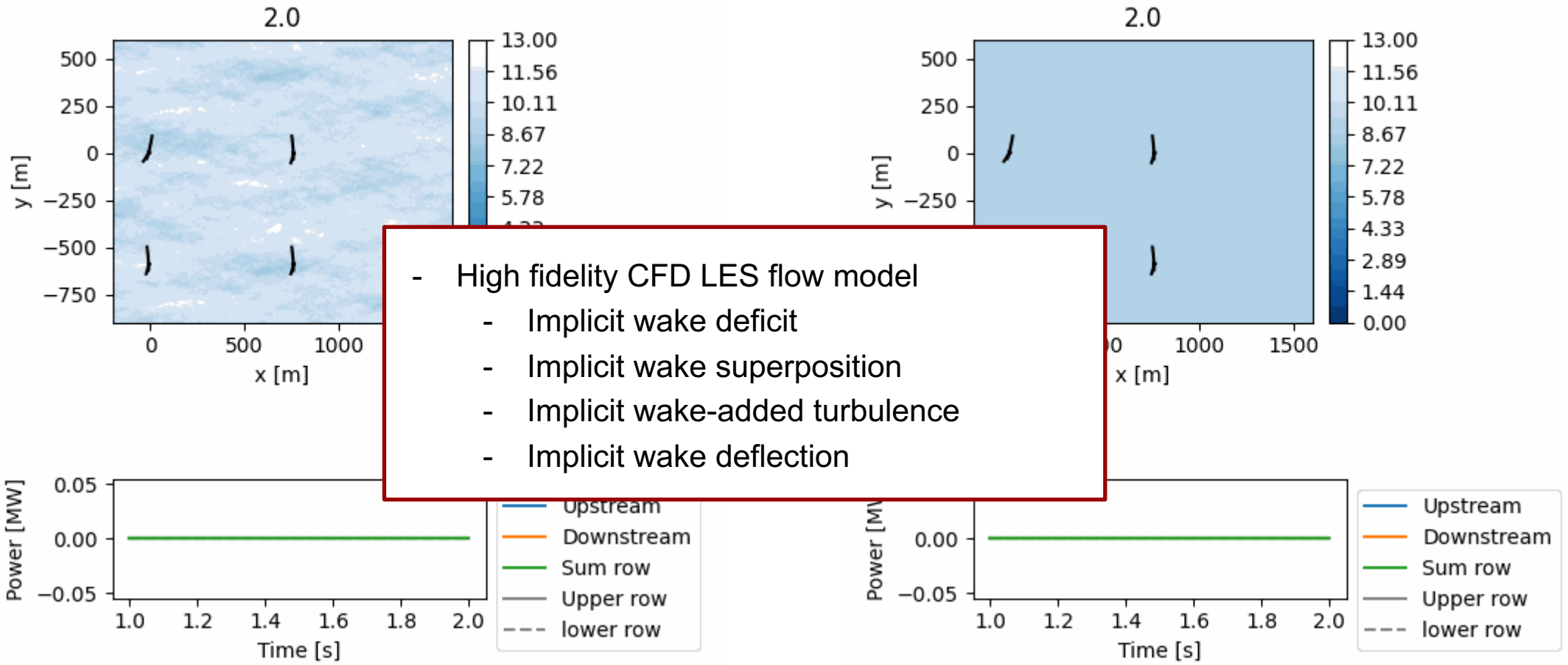



```
fs = Ellipsys3D_MPI_FlowSimulation(
    windTurbines=wts,
    input_file="./input.dat")
```



```
fs = DWMFlowSimulation(
    windTurbines=wts,
    site=site)
```

```
fs = Ellipsys3D_MPI_FlowSimulation(
    windTurbines=wts,
    input_file="./input.dat")
```



- High fidelity CFD LES flow model
  - Implicit wake deficit
  - Implicit wake superposition
  - Implicit wake-added turbulence
  - Implicit wake deflection

What if

What if:

– Wind direction changes

Then:

– Yaw controller estimates wind direction from the turbulent wind speed
– Yaw controller waits to ensure that the mean wind direction really has changed
– Yaw controller realize that the wind is aligned with the row
– Yaw controller gives a 18deg misalignment setpoint to the yaw actuator
– Yaw actuator starts to yaw
– Turbine reaches its misaligned angle
– Wake starts to deflect
– Deflected wake advects to the downstream turbine
– Downstream turbine starts to produce more power

What if

What if:

– Wind direction changes

Then:

– Yaw controller estimates wind direction from the turbulent wind speed

– Yaw controller waits to ensure that the me direction really has changed

– Yaw controller realize that the wind with the row

– Yaw controller gives a 18deg m nt setpoint to the yaw actuator

– Yaw actuator starts to yaw

– Turbine reaches its m angle

– Wake starts to d

– Deflected w s to the downstream turbine

– Downstream ne starts to produce more power

Wind direction changes again