



# **A novel framework for multidisciplinary environmental modeling using pseudo-natural language**

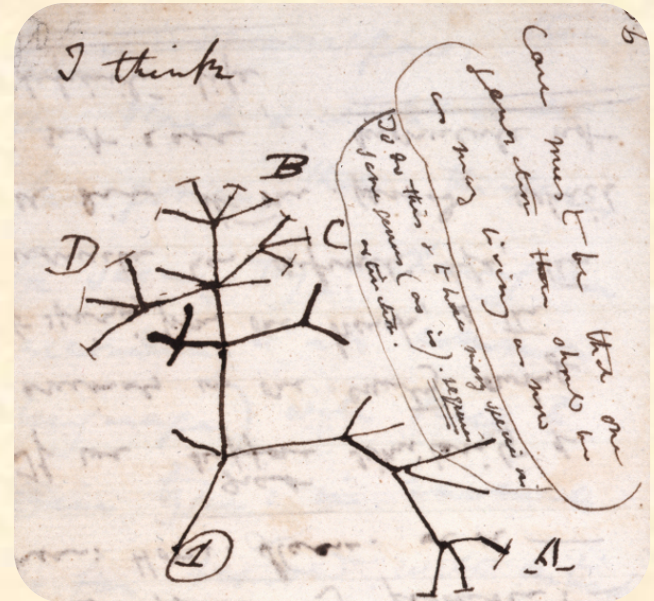
**Herman Guillermo Dolder**  
**Brigham Young University**



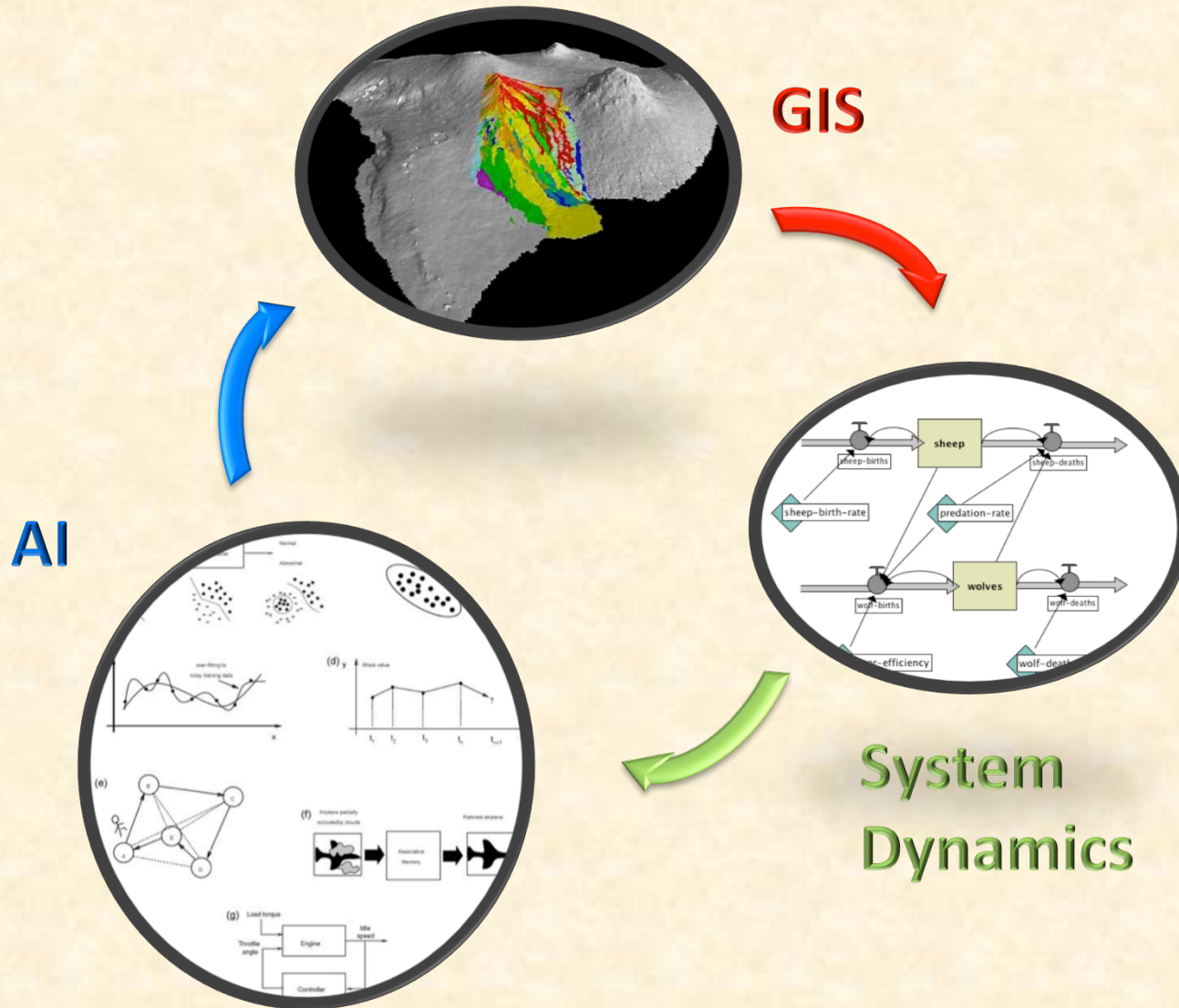
# Paradigm



+



# Technology



# Objective



“Anyone can make an Environmental Impact Assessment”



### **elevation\_nm.txt**

Call this cover elevation

Load values from C:/Models/Initial/LittleDellTest2.ele using C:/Models/Initial/LittleDellTest2.msk as a mask

### **waterdepth\_nm.txt**

Call this cover waterdepth

Create values around 0.05 plus-minus 0.005 using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

Read slopes from surroundings of elevation and call it water\_surface\_slope

Read aspects from surroundings of elevation and call it water\_surface\_aspect

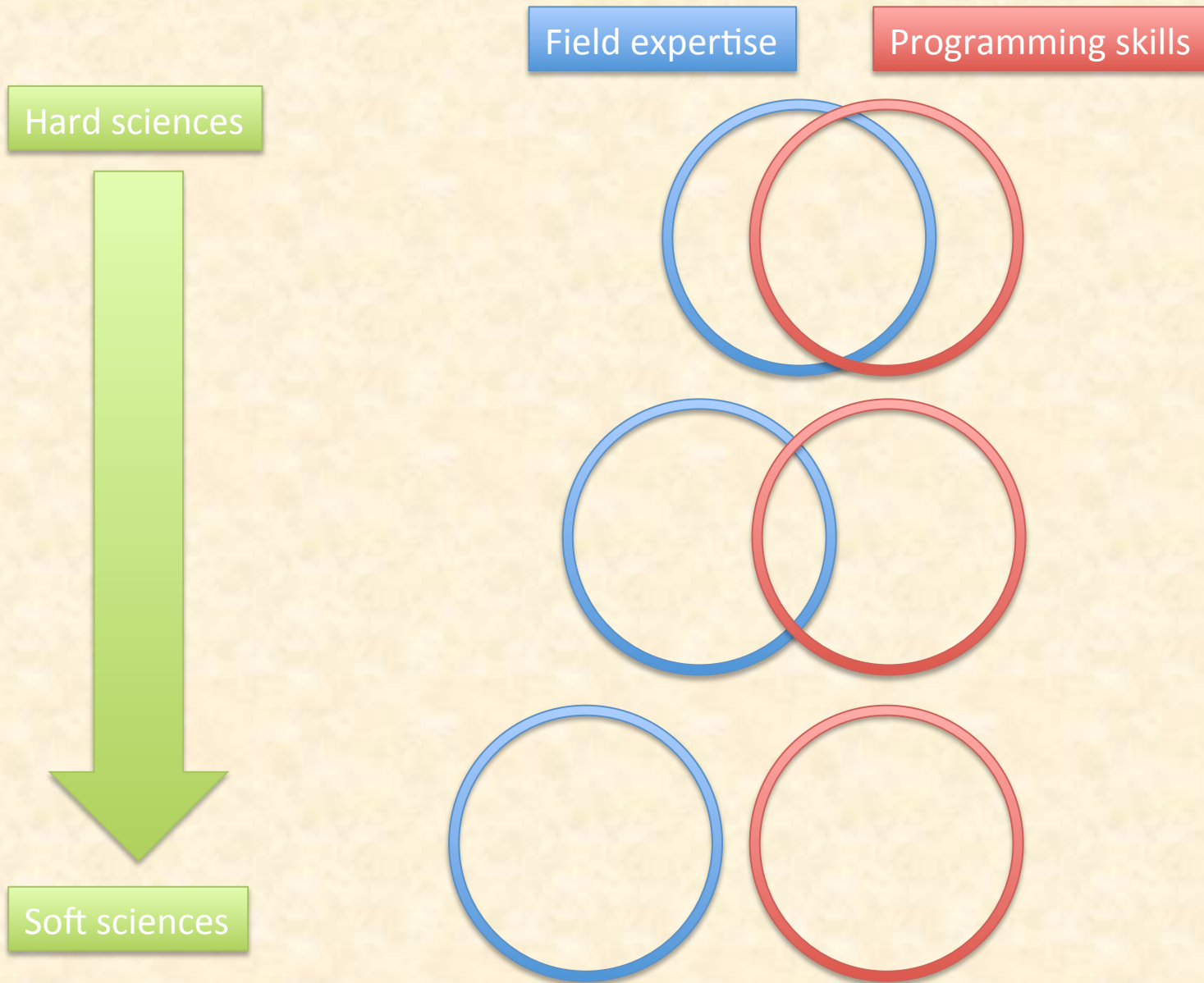
Move using slope water\_surface\_slope and aspect water\_surface\_aspect with a weight of  $1/50.0^{**2}$  and call it newval

Set the new value newval





## Pseudo-natural language...



## Behind the scenes...

### **waterdepth\_nm.txt**

Call this cover waterdepth

Create values around 0.05 plus-minus 0.005 using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

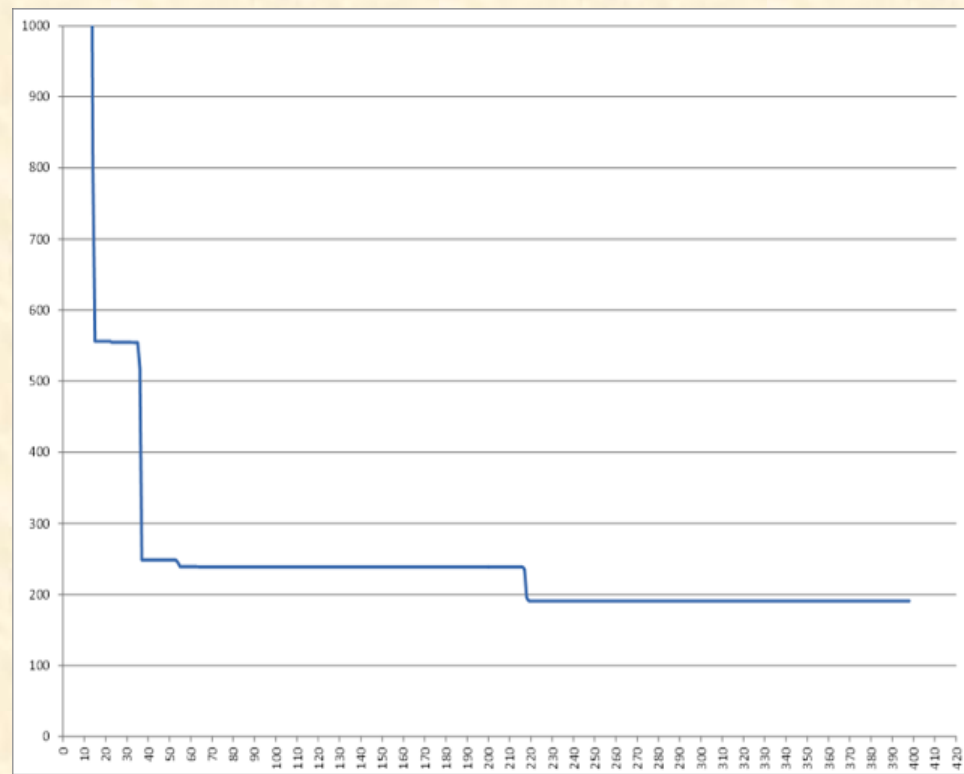
Read slopes from surroundings of elevation and call it water\_surface\_slope

Read aspects from surroundings of elevation and call it water\_surface\_aspect

Move using slope water\_surface\_slope and aspect water\_surface\_aspect with a weight of  $1/50.0^{**2}$  and call it newval

Set the new value newval

```
waterdepth = Cover('waterdepth')
coverlist.append(waterdepth)
waterdepth.loadFromArray(loadMap('C:/Models/Initial/LittleDellTest2.msk'), 'mask')
waterdepth.loadFromArray(createMap(waterdepth.rows, waterdepth.cols, 0.05, 0.005), 'map')
waterdepth.frequency = 5
def updateWaterdepth(rowcol):
    calc = Primitive()
    water_surface_slope = watergradslope.getSurrValue(rowcol)
    water_surface_aspect = watergradaspect.getSurrValue(rowcol)
    newval = calc.moveprop(waterdepth.getSurrValue(rowcol), water_surface_slope,
water_surface_aspect, 1/50.0**2, 0, 1)
    return newval
waterdepth.updfun = updateWaterdepth
```





### **elevation\_nm.txt**

Call this cover elevation

Load values from C:/Models/Initial/LittleDellTest2.ele using C:/Models/Initial/LittleDellTest2.msk as a mask

### **water\_elevation\_nm.txt**

Call this cover water\_elevation

Load values from C:/Models/Initial/LittleDellTest2.ele using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

Read value from elevation and call it elev

Read value from waterdepth and call it wdepth

Calculate  $\text{elev} + \text{wdepth}$  and call it newelev

Set the new value newelev

### **waterdepth\_nm.txt**

Call this cover waterdepth

Create values around 0.05 plus-minus 0.005 using C:/Models/Initial/LittleDellTest2.msk as a mask

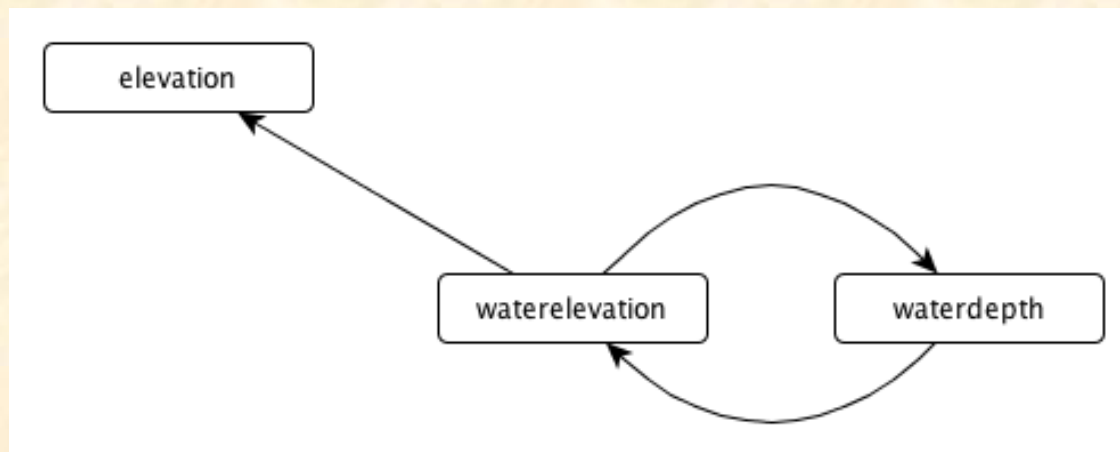
Update this cover every 1 iterations

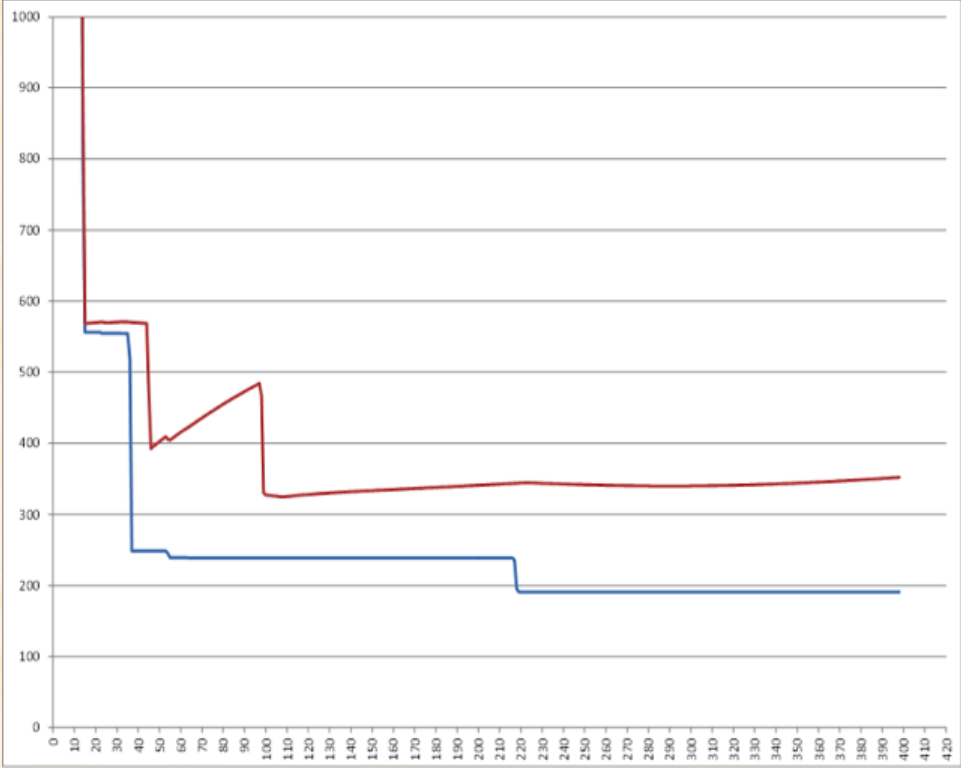
Read slopes from surroundings of water\_elevation and call it water\_surface\_slope

Read aspects from surroundings of water\_elevation and call it water\_surface\_aspect

Move using slope water\_surface\_slope and aspect water\_surface\_aspect with a weight of  $1/50.0^{**2}$  and call it newval

Set the new value newval





#### **watergradaspect\_nm.txt**

Call this cover watergradaspect

Create values around 0.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

Read slope from waterelevation and call it w\_slope

Read aspect from waterelevation and call it w\_aspect

Calculate  $w\_slope/50.0 * \cos(w\_aspect * \pi/180.0)$  and call it orislopex

Calculate  $w\_slope/50.0 * \sin(w\_aspect * \pi/180.0)$  and call it orislopey

Calculate  $\text{abs}(\text{orislopex}) ** 0.5$  and call it slopex

Calculate  $\text{abs}(\text{orislopey}) ** 0.5$  and call it slopey

If orislopex is less than 0 make slopex -slopex

If orislopey is less than 0 make slopey -slopey

Calculate  $(\text{math.atan2}(\text{slopey}, \text{slopex}) * 180 / \text{math.pi}) \% 360$  and call it newaspect

Set the new value newaspect

#### **watergradslope\_nm.txt**

Call this cover watergradslope

Create values around 0.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

Read slope from waterelevation and call it w\_slope

Read aspect from waterelevation and call it w\_aspect

Calculate  $w\_slope/50.0 * \cos(w\_aspect * \pi/180.0)$  and call it orislopex

Calculate  $w\_slope/50.0 * \sin(w\_aspect * \pi/180.0)$  and call it orislopey

Calculate  $\text{abs}(\text{orislopex}) ** 0.5$  and call it slopex

Calculate  $\text{abs}(\text{orislopey}) ** 0.5$  and call it slopey

If orislopex is less than 0 make slopex -slopex

If orislopey is less than 0 make slopey -slopey

Read value from waterdepth and call it d

Calculate  $1/0.03 * d ** (5/3.0) * (\text{slopex} ** 2 + \text{slopey} ** 2) ** 0.5$  and call it newslope

Set the new value newslope

#### **waterdepth\_nm.txt**

Call this cover waterdepth

Create values around 0.01 plus-minus 0.005 using C:/Models/Initial/LittleDellTest2.msk as a mask

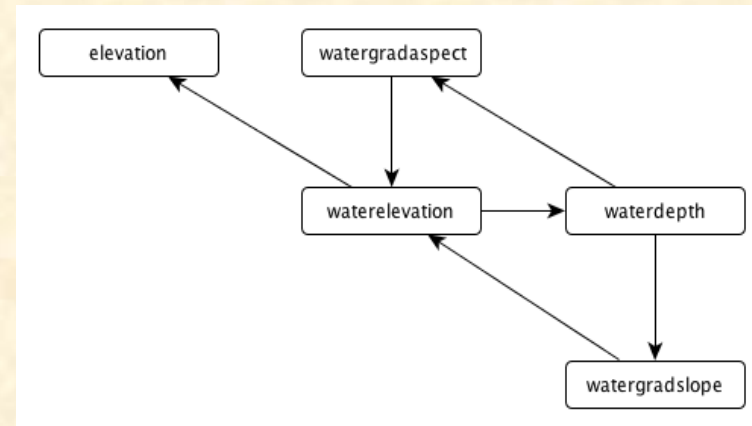
Update this cover every 1 iterations

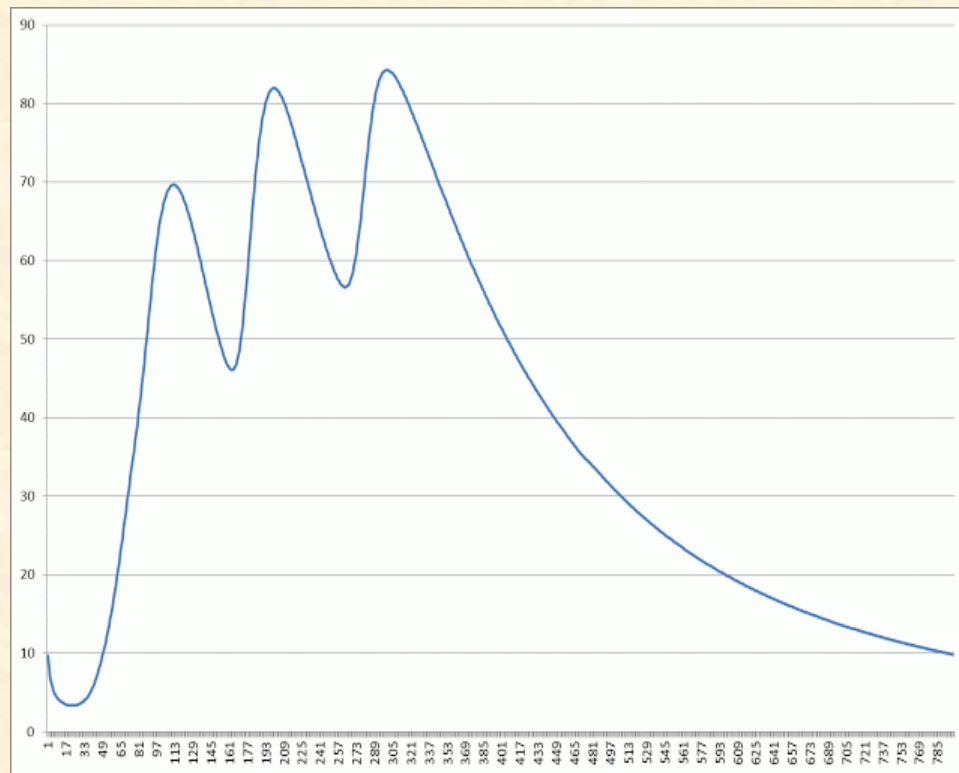
Read slopes from surroundings of watergradslope and call it water\_surface\_slope

Read aspects from surroundings of watergradaspect and call it water\_surface\_aspect

Move using slope water\_surface\_slope and aspect water\_surface\_aspect with a weight of 0.02 and call it newval

Set the new value newval







#### **watervelocity\_nm.txt**

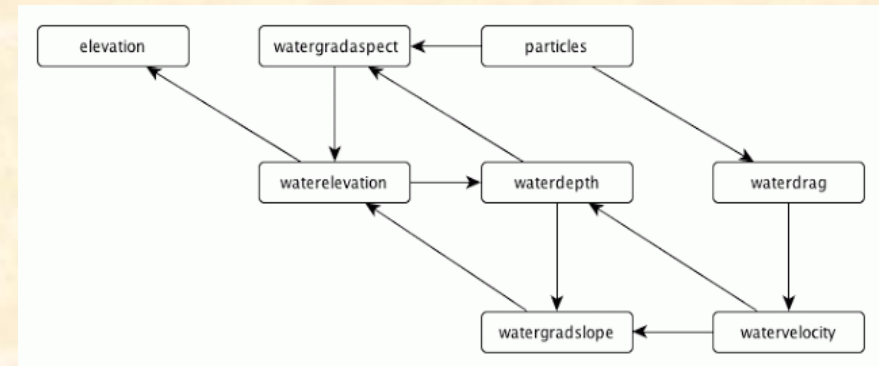
Call this cover watervelocity  
 Create values around 0.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk  
 as a mask  
 Update this cover every 5 iterations  
 Read value from watergradslope and call it flow  
 Read value from waterdepth and call it water\_depth  
 Calculate  $\text{flow}/\text{water\_depth}$  and call it newval  
 Set the new value newval

#### **waterdrag\_nm.txt**

Call this cover waterdrag  
 Create values around 0.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk  
 as a mask  
 Update this cover every 5 iterations  
 Read value from watervelocity and call it velocity  
 Read value from waterdepth and call it depth  
 Calculate  $(0.5 + \text{velocity}) ** 2$  and call it newdrag  
 If velocity is less than 0.5 make newdrag 0  
 If depth is less than 0.01 make newdrag 0  
 Set the new value newdrag

#### **particles\_nm.txt**

Call this cover particles  
 Create values around 1.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk  
 as a mask  
 Update this cover every 5 iterations  
 Read values from surroundings of waterdrag and call it drag  
 Read values from surroundings of watergradaspect and call it direction  
 Move using slope drag and aspect direction with a weight of 1 and call it newval  
 Set the new value newval







## Primitives

Call this cover ...

Load values from ... using ... as a mask

Create values around ... plus-minus ... using ... as a mask

Update this cover every ... iterations

Read value from ... and call it ...

Read slope from ... and call it ...

Read aspect from ... and call it ...

Read change rate from ... and call it ...

Read slopes from surroundings of ... and call it ...

Read aspect from surroundings of ... and call it ...

Read change rate from surroundings of ... and call it ...

Move using slope ... and aspect ... with a weight of ... and call it ...

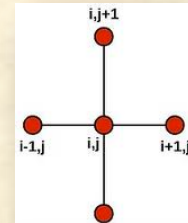
Propagate with a weight of ... and call it ...

Move and propagate using slope ... and aspect ... with a moving weight of ... and a propagation weight of ... and call it ...

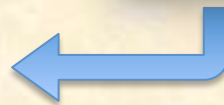
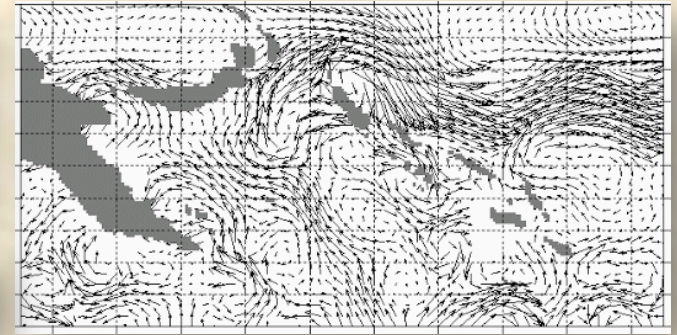
Calculate ... and call it ...

If ... is less than | is greater than | (etc) ... make ...

Set the new value ...



$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0.$$



### roughness\_nm.txt

Call this cover roughness

Load values from C:/Models/Initial/roughness.rgn using C:/Models/Initial/LittleDellTest2.msk as a mask

### watergradslope\_nm.txt

Call this cover watergradslope

Create values around 0.0 plus-minus 0.0 using C:/Models/Initial/LittleDellTest2.msk as a mask

Update this cover every 1 iterations

Read slope from waterelevation and call it w\_slope

Read aspect from waterelevation and call it w\_aspect

Calculate  $w\_slope/50.0 * \cos(w\_aspect * \pi/180.0)$  and call it orislopex

Calculate  $w\_slope/50.0 * \sin(w\_aspect * \pi/180.0)$  and call it orislopy

Calculate  $\text{abs}(\text{orislopex})^{**0.5}$  and call it slopex

Calculate  $\text{abs}(\text{orislopy})^{**0.5}$  and call it slopey

If orislopex is less than 0 make slopex -slopex

If orislopy is less than 0 make slopey -slopey

Read value from waterdepth and call it d

Read value from roughness and call it n

Calculate  $1/n * d^{** (5/3.0) * (\text{slopex}^{**2} + \text{slopey}^{**2})^{**0.5}}$  and call it newslope

Set the new value newslope

