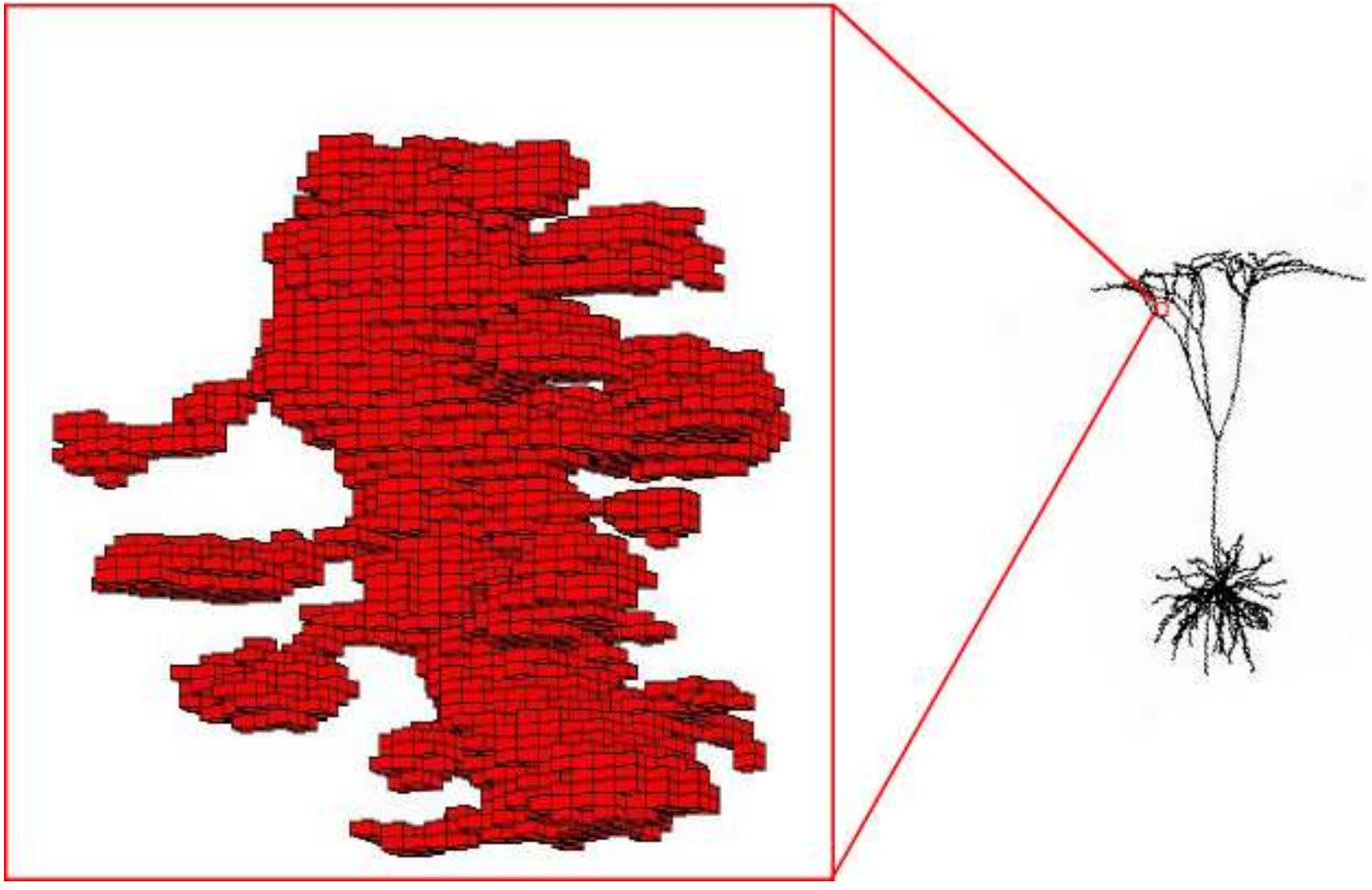
NEURON + Reaction−Diffusion

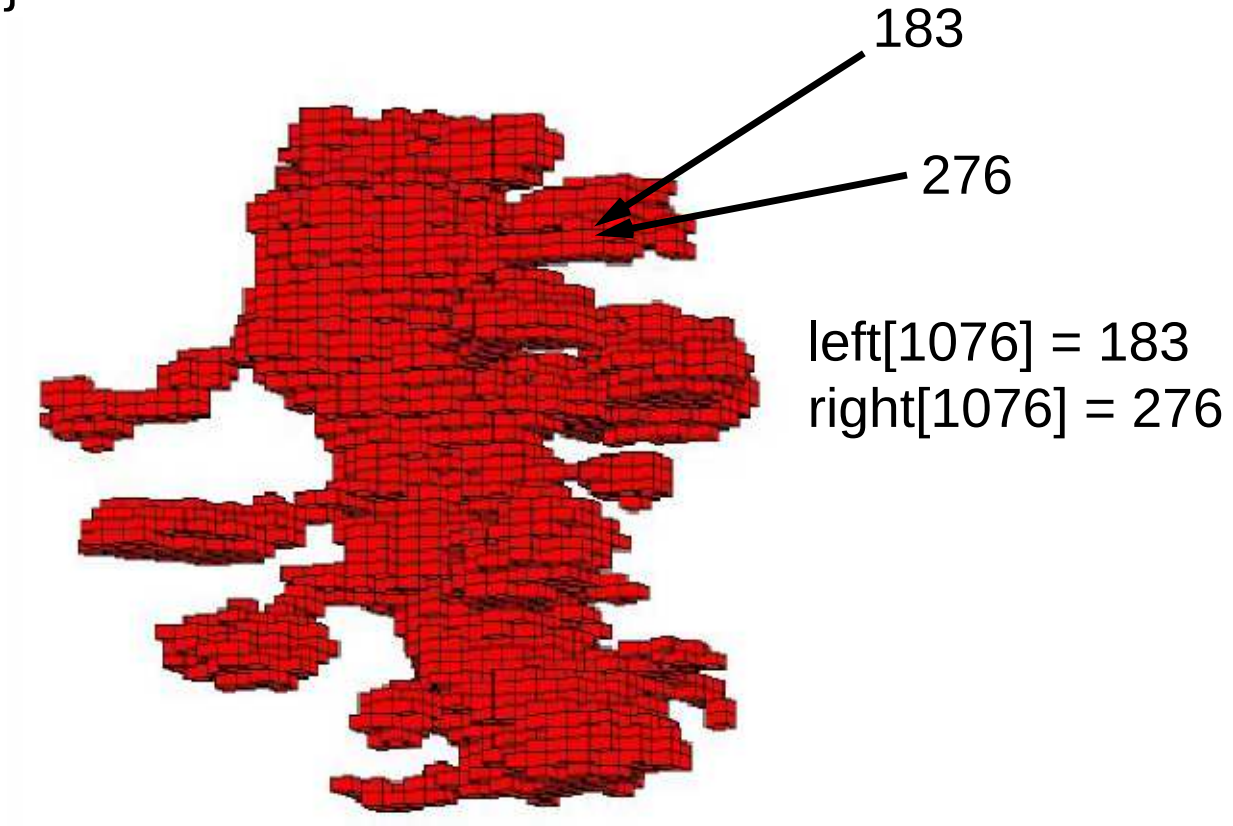
```
KINETIC conc {
  COMPARTMENT PI*diam*diam/4 {nai}
  LONGITUDINAL_DIFFUSION D*PI*diam*diam/4 {nai}
  ~ nai << (−ina/(FARADAY)*PI*diam*(1e4))
}
```

```
KINETIC state {
  COMPARTMENT i, diam*diam*vol[i] {ca CaBuffer Buffer}
  LONGITUDINAL_DIFFUSION j, DFree*diam*diam*vol[j] {ca}
  ~ ca[0] << (−ica*PI*diam*frat[0]/(2*FARADAY))
  FROM i=0 TO NANN−2 {
    ~ ca[i] <−> ca[i+1] (DFree*frat[i+1], DFree*frat[i+1])
  }
  dsq = diam*diam
  FROM i=0 TO NANN−1 {
    dsqvol = dsq*vol[i]
    ~ ca[i] + Buffer[i] <−> CaBuffer[i] (k1buf*dsqvol,k2buf*dsqvol)
  }
  cai = ca[0]
}
```

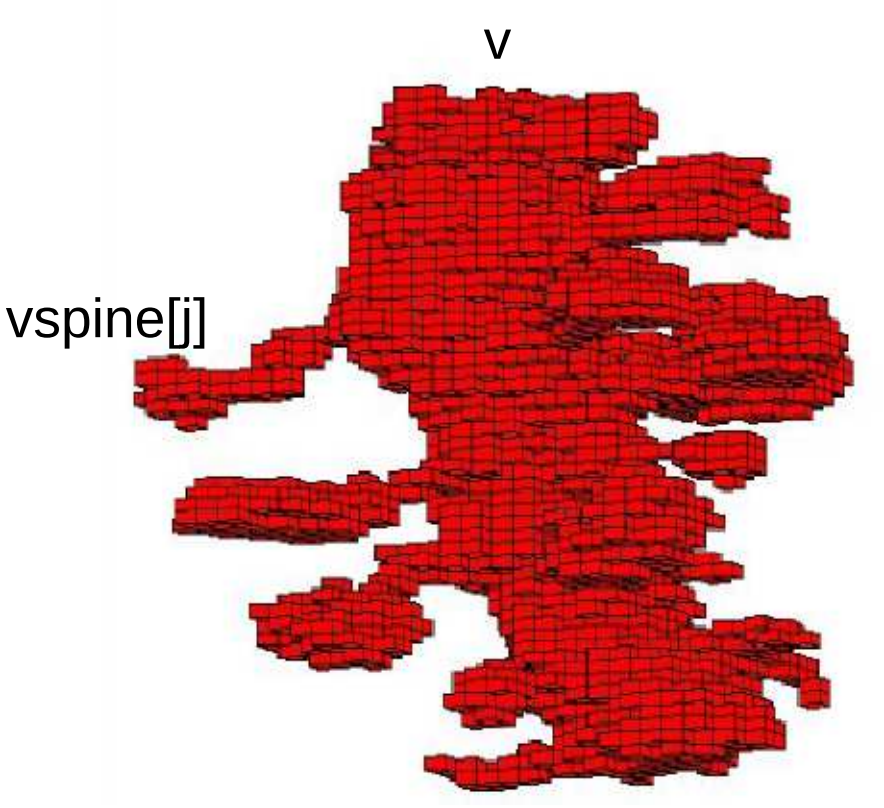


From Dan Keller, Blue Brain Project, EPFL

```
KINETIC scheme {
  COMPARTMENT i, vol[i] {ca caBuffer}
  FROM i = 0 TO N_EDGE−1 {
    ca[left[i]] <−> ca[right[i]] (f[i], b[i])
  }
  FROM i = 0 TO N_COMPART−1 {
    ca[i] <−> caBuffer[i]
  }
}
```



```
BREAKPOINT {
  :ispine[j] = gspine[j]*(vspine[j] − eca)
  ispine[j] = gspine[j]*(v − eca)/(1 + rspine[j]*gspine[j])
  vspine[j] = v − ispine[j]*rspine[j]
```



```
region interior
  ca <−> cabuffer (forward, reverse)
  ca diffusible DFree = ...
  cabuffer not−diffusible
region membrane
  pump not−diffusible
  capump not−diffusible
  interior.ca + pump <−> capump (k1, k2)
  capump <−> exterior.ca + pump (k3, k4)
region exterior
  ca diffusible  DFree = ...
  // does not matter if diffusible or not
  // since there is only one exterior
  // compartment with infinite volume
```