

# NeuroML with PyMOOSE

Aditya Gilra,  
Chaitanya Chintaluri,  
Niraj Dudani  
NCBS, Bangalore, India

libNeuroML meeting  
UCL, London  
July 26th, 2012

# History:

## NeuroML import in MOOSE $\beta$ 1.3

- Siji George (Upi lab)
- Uses libNeuroML v0.2:

<http://neuroml.svn.sourceforge.net/viewvc/neuroml/TestLibNeuroML/v0.2/libneuroml/>

- C++, libxml2
- moose/branches/moose-Beta-1.3.0/external/neuroML\_src
- moose/branches/moose-Beta-1.3.0/neuroML\_IO
- moose/branches/moose-Beta-1.3.0/DEMOS/NeuroML\_Reader
  - Granule, Purkinje, CA1 (Ca pool, K\_AHP, etc)

# Why python importer then?

- libNeuroML v0.2: no NetworkML
- Bhalla & Bower mitral cell '93 (284 comps):
  - GENESIS→NEURON (Andrew Davison)
  - NEURON→NeuroML morphology exporter exports <cablegroup> tags to group cables
  - libNeuroML v0.2 only supports <meta:group> tags inside <cable> tags.
  - Redundancy in NeuroML1 – bad?
- Bug importing a validated H channel
- Python easier than debugging C++

Aditya

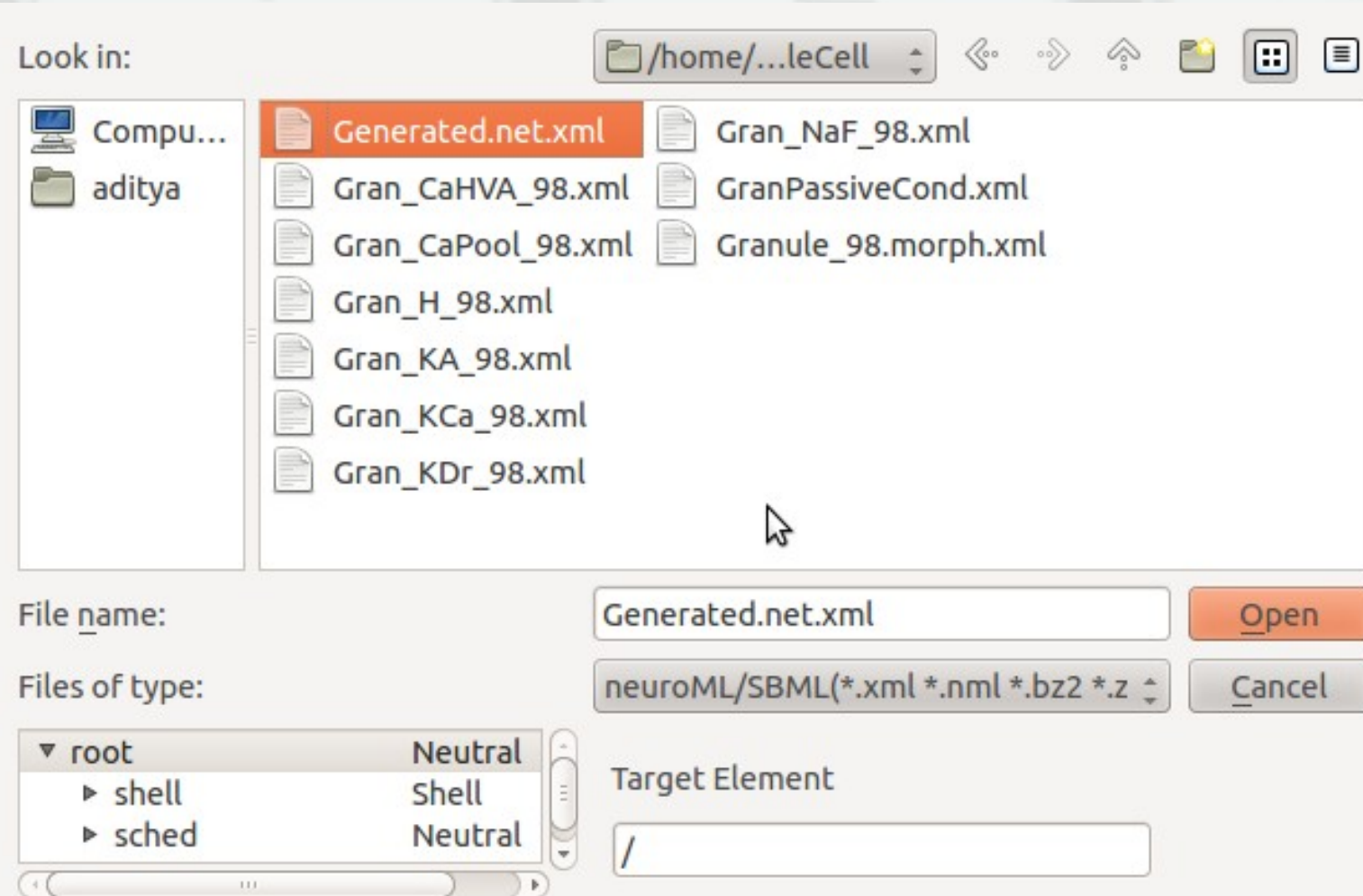
# Importer MOOSE $\beta$ 1.4 – Aditya

- ElementTree module
  - Included / standard in Python2.5
  - Mostly compatible with the faster cElementTree or lxml, can switch later
- ChannelML, MorphML, NetworkML from file/node.
- Load channels, then cells into /library. Then create populations & projections. OR
- Load NeuroML file and load cells / channels from it, or separate files in its directory.

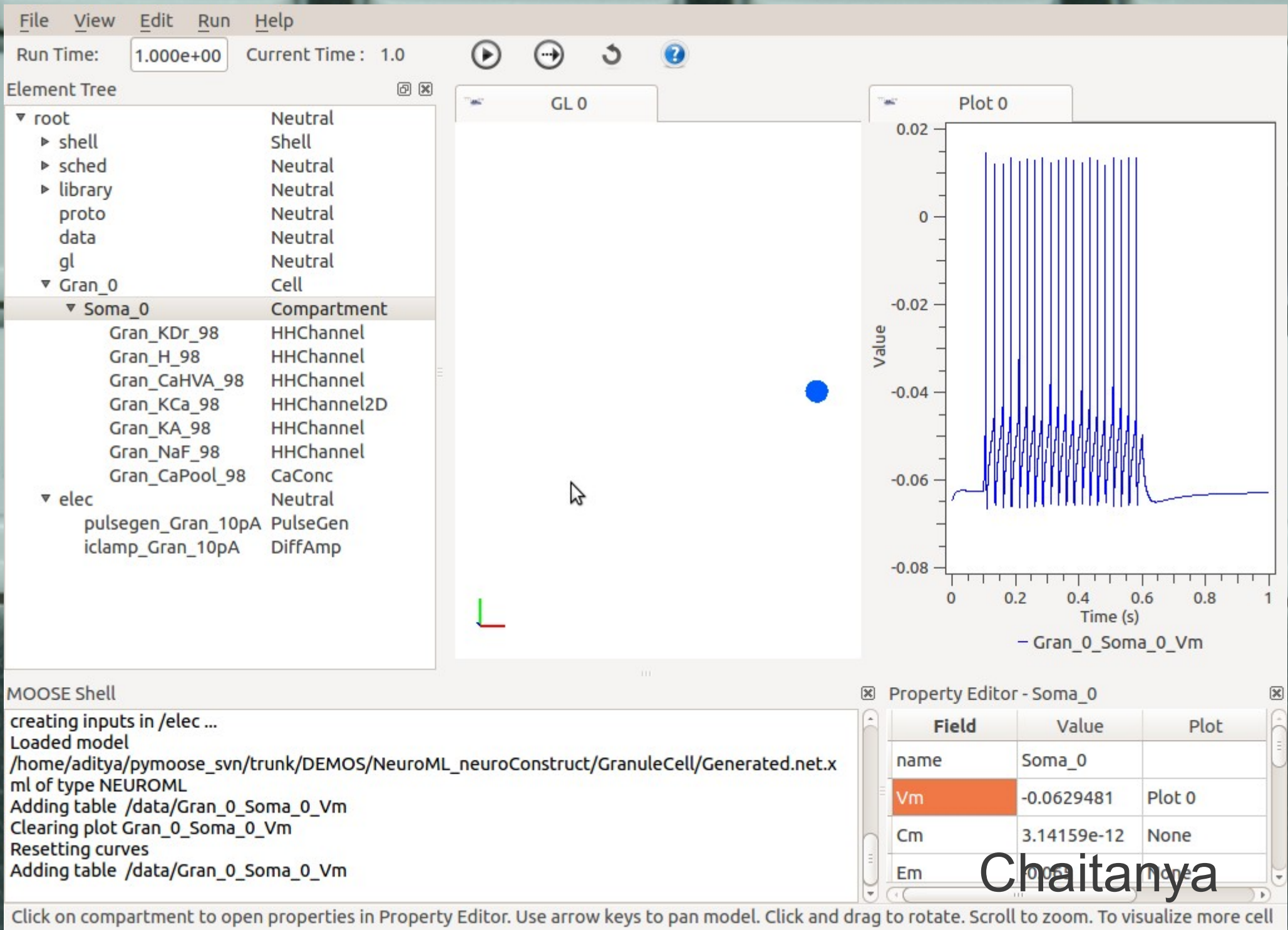
Aditya



# MOOSE $\beta$ 1.4 GUI – Chaitanya



Chaitanya



# OB model – Aditya: Usage

```
from moose.neuroml import *

load_channels()

CML = ChannelML({'temperature':CELSIUS})
CML.readChannelMLFromFile('../channels/lh_cb.xml')

cellSegmentDict = load_cells()

MML = MorphML({'temperature':CELSIUS})
mitral_dict =
    MML.readMorphMLFromFile('../cells/mitral_bbmit1993davison_neuroML
        _L1_L2_L3_mod.xml',{'})

cellSegmentDict.update(mitral_dict)

## populationDict = { 'populationname1':(cellname,{instanceid1:moosecell, ... }) , ... }

## projectionDict = { 'projectionname1':(source,target,
[(syn_name1,pre_seg_moosepath,post_seg_moosepath),...]) , ... }

(populationDict,projectionDict) = \

NML.readNetworkMLFromFile(filename,cellSegmentDict,params=tweaks)
```

Aditya



# ChannelML1 issues

- Global properties like temperature (meta?)
- Lookup table channels – Na in BBmit93
- Destexhe et al 's receptor-saturating synapse (implemented in MOOSE)

<http://cns.iaf.cnrs-gif.fr/files/synapse.pdf>



# MorphML1 issues

- `<cablegroup>`s inside `<cables>` (NEURON export)  
versus
- `<meta:group>` tags within `<cable>`
  - Former: easy to implement
  - Latter: easy to read / understand model
  - XSLT converter between these?

# MorphML1 issues

- <segment> and <cable>
  - 1 segment to a compartment?
- Soma with dendrites at opp ends:
  - <segment> has opt attrib cable
  - <cable> & fract\_along\_parent attrib.
  - Need to parse <segments> and <cables> to get connectivity
- Attrib for every segment: connect proximal vs distal to parent.

# NetworkML1 issues

- Rotation of cell instances
- Inhomogeneous populations?
  - Override say RMP of individual cells
  - cell\_type as a list in <population>
  - Collections of populations with same potential synaptic locations
  - Projections should not be affected

# Implementation issues

- Rotation of cell instances by <meta:notes>
- Asym compartment converted to sym compartment by Hines solver in MOOSE: axial of parent to raxial of child segment
- Changing params of channel in MorphML not implemented
- Need clean separation of NeuroML reader to in-memory model VS sim-dependent model creation code!



# XSLT to X3D – Aditya

- Enhancements to the XSL file by Padraig:
  - NetworkML support:
  - Populations and projections
  - Cells as spheres / morphology if present
  - Cell morphology with dendritic widths
- On the NeuroML website
- Olfactory bulb model (Aditya) visualized in X3D ...

Aditya

## Mitral Cells [Bhalla & Bower 1993]

10 gloms, 20 cells, 400 ORNs-->mit per mit

Glomerulus  
Primary Dendrite  
Soma  
Lateral Dendrite

## PG Cells [self],

~400 aggregated cells

500 ORNs-->PG per PG,

25 mitral-->PG per PG

100 PG--|mitral per mitral

## Granule Cells [Migliore & Shepherd 2008]

~8,500 aggregated cells

100 mitral-->granule per granule,

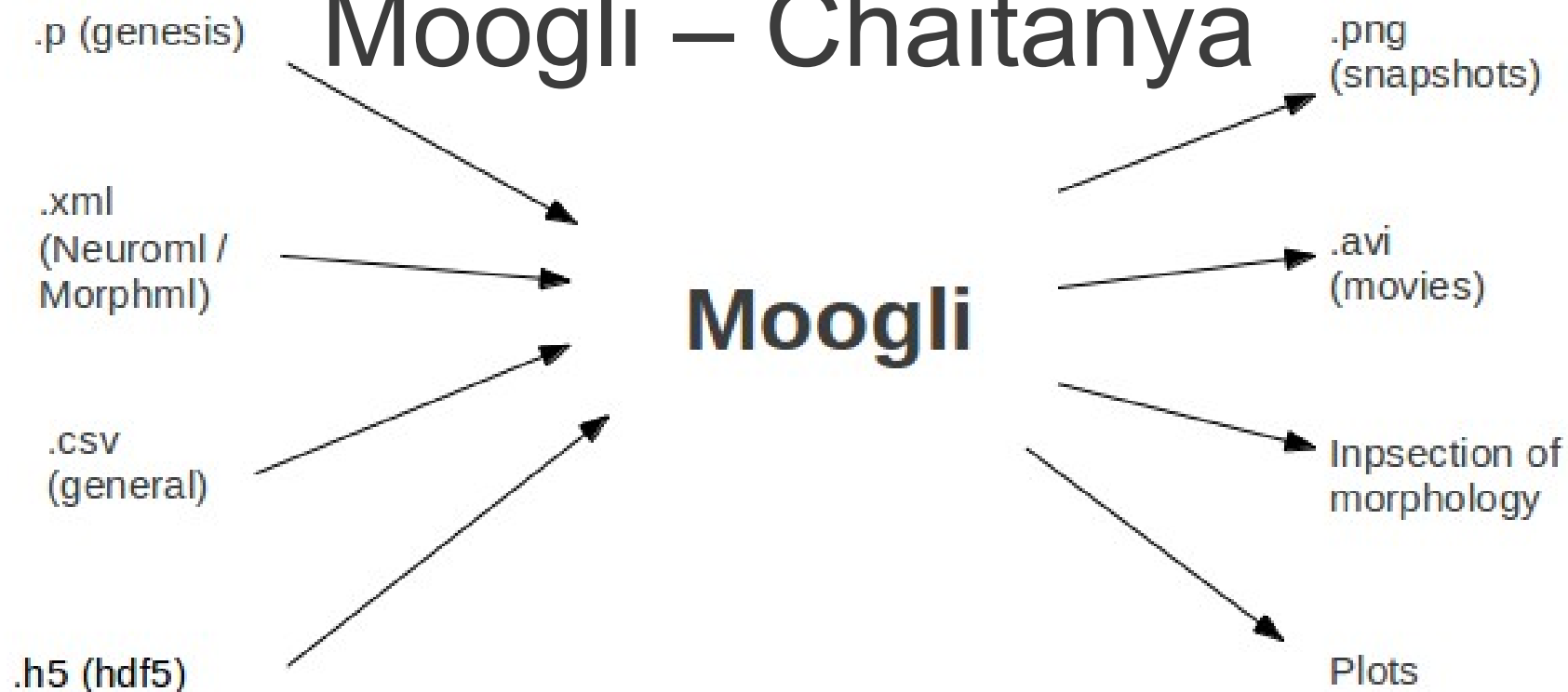
1000 granule--|mitral per mitral

Aditya



Post simulation analysis tool

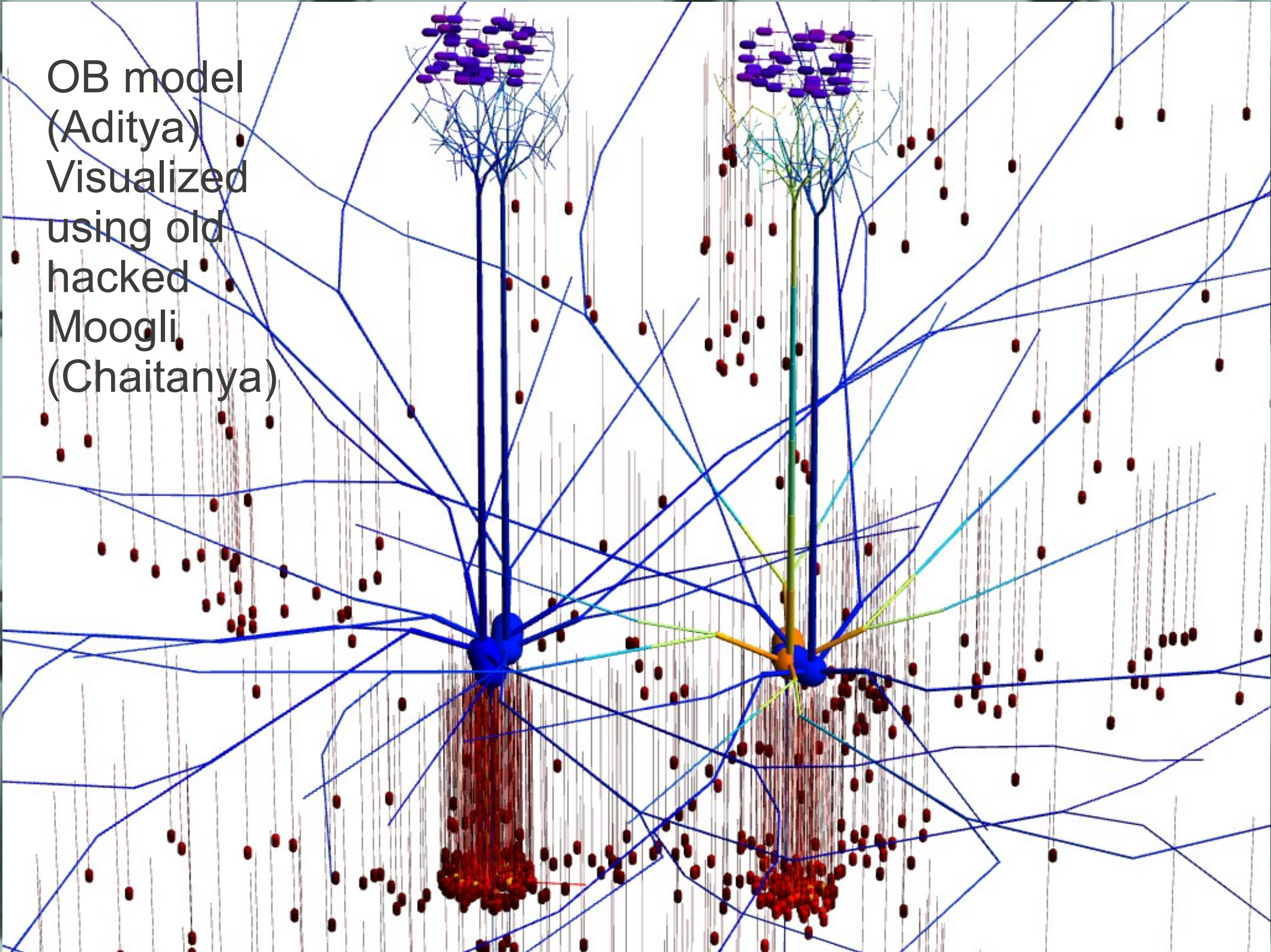
# Moogli – Chaitanya



- hdf5 import: file includes morphology of the cell(s) being visualized.
- NEO support coming

Chaitanya

OB model  
(Aditya)  
Visualized  
using old  
hacked  
Moogli  
(Chaitanya)





# LEMS

- Core set of ComponentTypes should correspond to existing objects in simulators
  - Performance issues: Niraj – hsolve
- Markov channels implemented in MOOSE
  - Vishaka, Niraj
  - Kinetic Schemes directly translate? – Niraj

# Acknowledgements

- MOOSE team:
  - Upi, Subha, Niraj, Chaitanya, Harsha, Siji, Aditya
- Lab 13 members
- Funding:
  - NCBS/TIFR,
  - DAE/SRC,
  - SBCNY/NIGNS
  - INCF