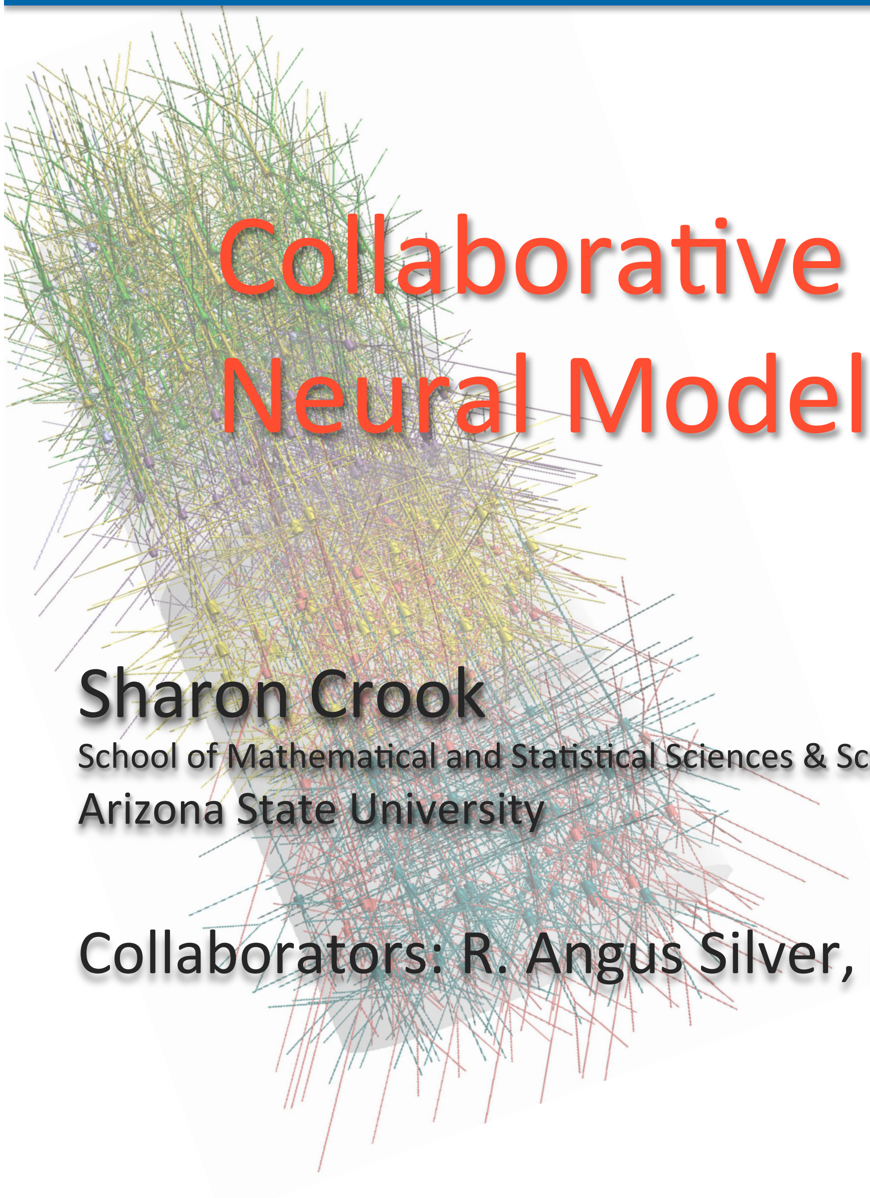


Collaborative Development of Neural Models with NeuroML

Sharon Crook

School of Mathematical and Statistical Sciences & School of Life Sciences

Arizona State University

Collaborators: R. Angus Silver, Padraig Gleeson, Robert Cannon

What is NeuroML?

Why do we need it?

How can it be used with modeling tools?

How does it support collaborative modeling?

Where is it going next?

What is NeuroML?

Towards NeuroML: Model Description Methods for Collaborative Modelling in Neuroscience

Nigel H. Goddard[1*], Michael Hucka[2], Fred Howell[1], Hugo Cornelis[3], Kavita Shankar[2] and David Beeman[4]

[1]*Institute for Adaptive and Neural Computation, Division of Informatics, University of Edinburgh, 5 Forrest Hill, Edinburgh EH1 2QL, Scotland*
[2]*Division of Biology 216-76, California Institute of Technology, Pasadena, CA 91125, USA*
[3]*Theoretical Neurobiology, Born–Bunge Foundation, University of Antwerp, Universteitsplein 1, 2610 Wilrijk, Belgium*
[4]*Department of Electrical and Computer Engineering 425 UCB, University of Colorado, 425 UCB, Boulder, CO 80309, USA*

Biological nervous systems and the mechanisms underlying their operation exhibit astonishing complexity. Computational models of these systems have been correspondingly complex. As these models become ever more sophisticated, they become increasingly difficult to define, comprehend, manage and communicate. Consequently, for scientific understanding of biological nervous systems to progress, it is crucial for modellers to have software tools that support discussion, development and exchange of computational models. We describe methodologies that focus on these tasks, improving the ability of neuroscientists to

Goddard et al. 2001, *Philos Trans R Soc Lond B Biol Sci.*

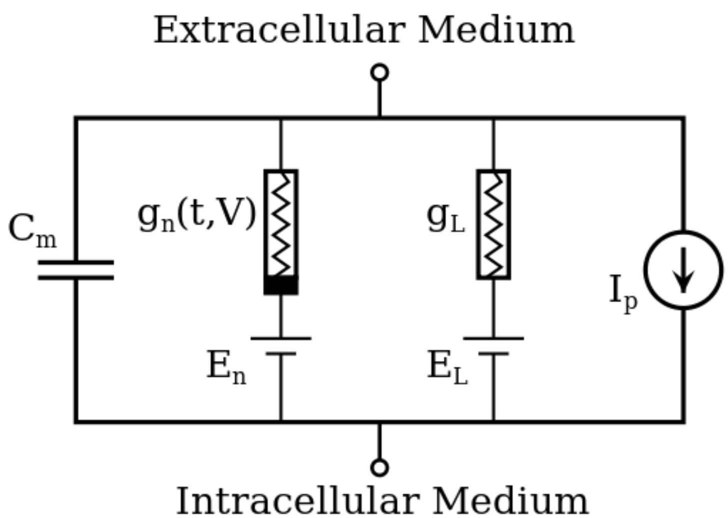
Why do we need it?

$$I_{ion} = \bar{g}_{Na}\, m^3\, h\, (V_m - E_{Na}) + \bar{g}_K\, n^4\, (V_m - E_K) + \bar{g}_L\, (V_m - E_L),$$

$$\frac{dm}{dt} = \alpha_m(V)\,(1-m) - \beta_m(V)\, m,$$

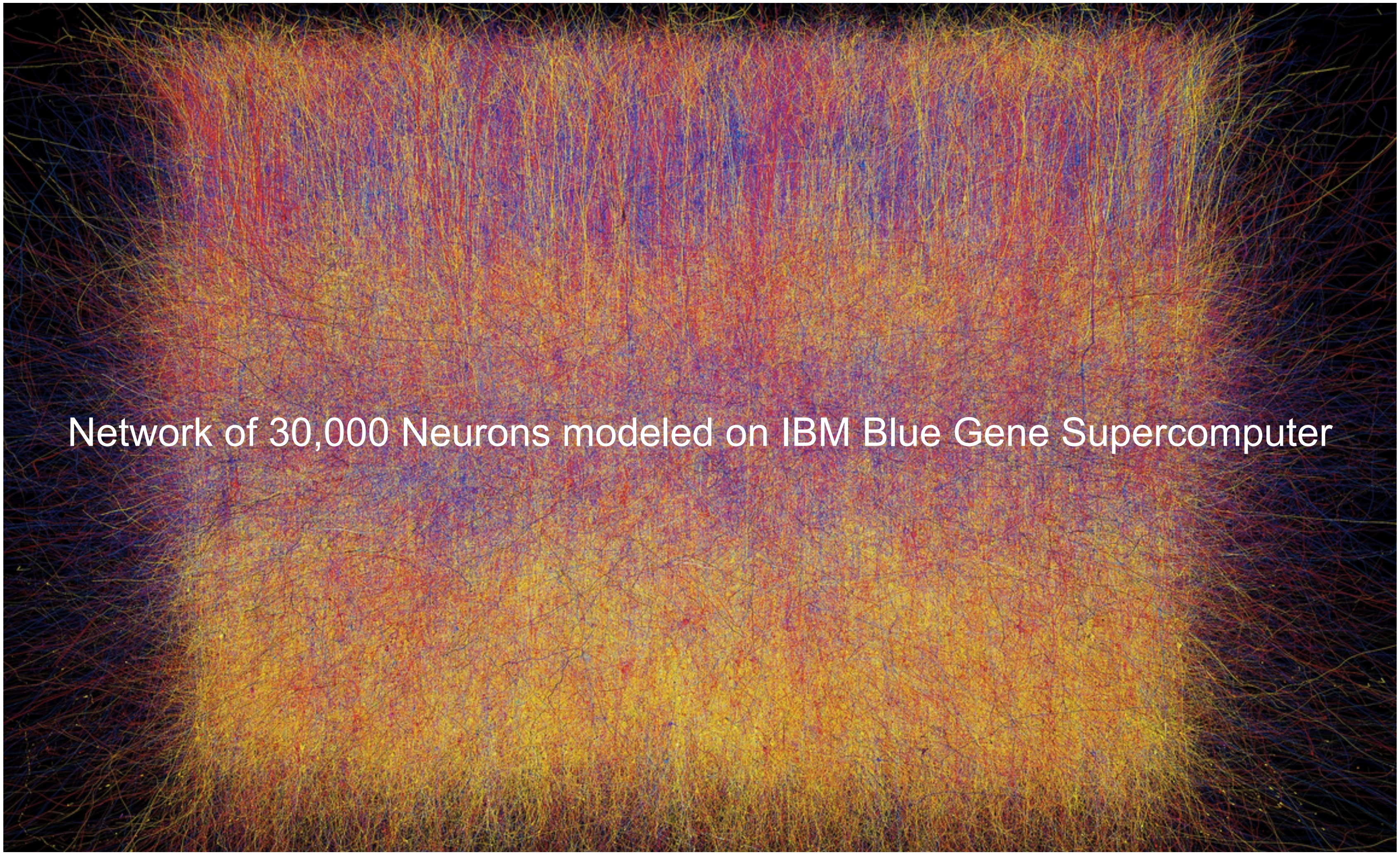
$$\frac{dh}{dt} = \alpha_h(V)\,(1-h) - \beta_h(V)\, h,$$

$$\frac{dn}{dt} = \alpha_n(V)\,(1-n) - \beta_n(V)\, n.$$

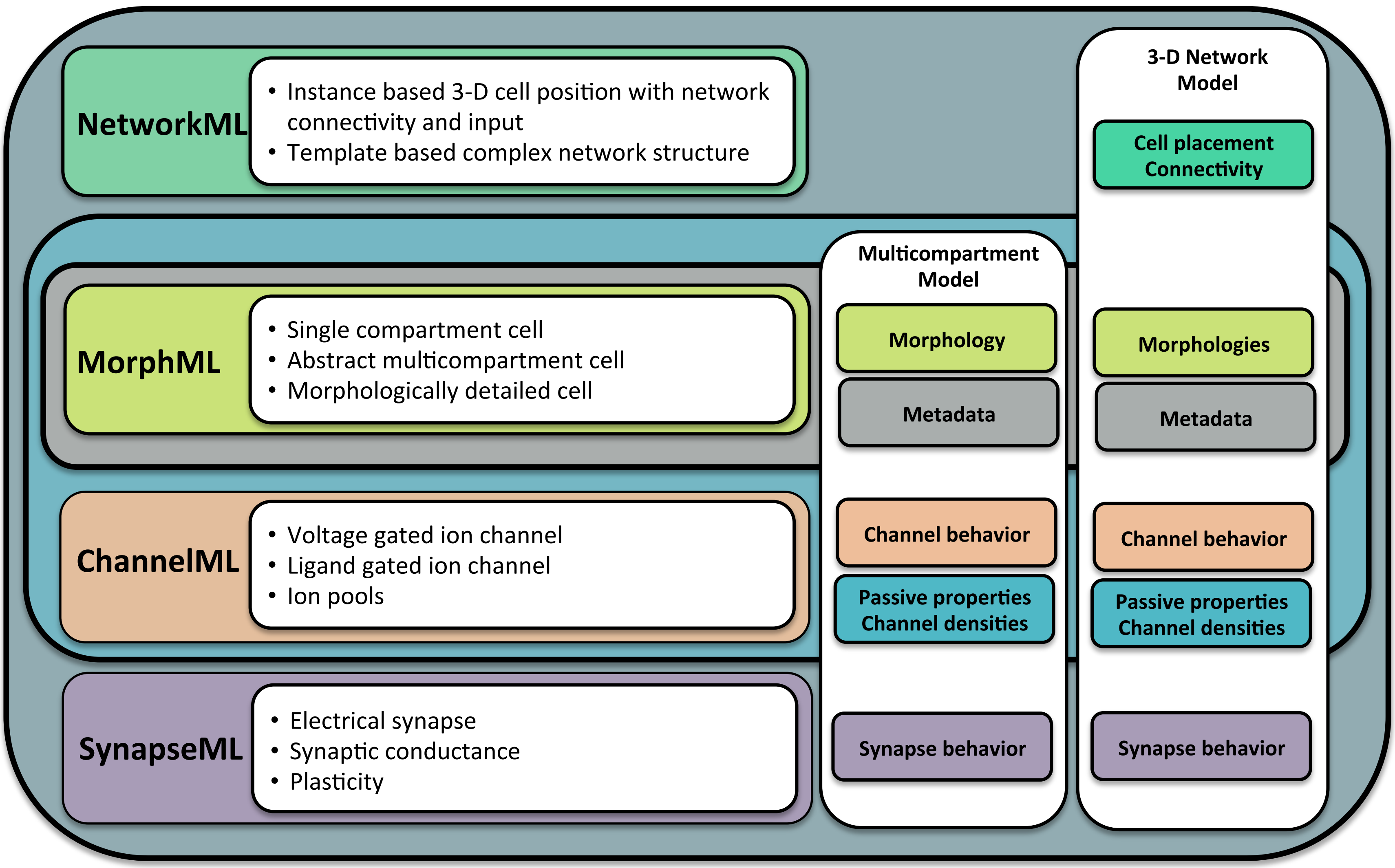


Hodgkin & Huxley 1952, *Journal of Physiology.*

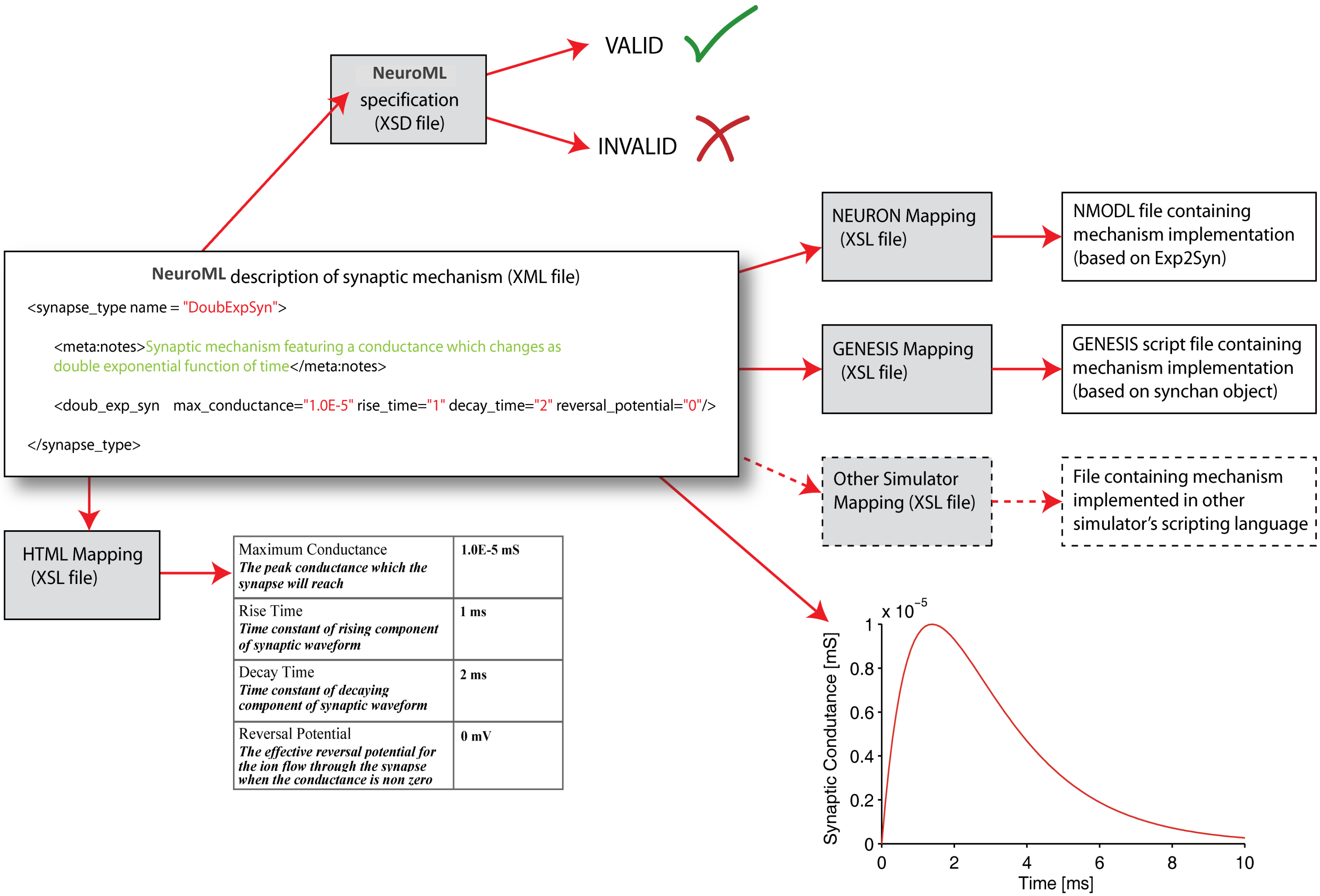
Blue Brain Project Model of Rat Cortex



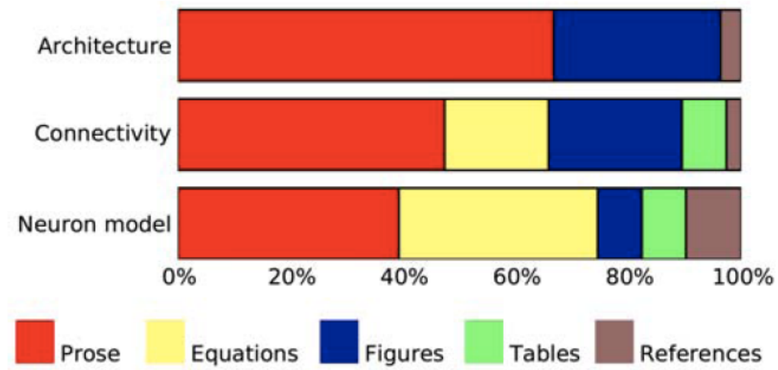
Markram et al. 2015, *Cell.*


NetworkML
Instance based 3-D cell position with network connectivity and input
Template based complex network structure
MorphML
Single compartment cell
Abstract multicompartment cell
Morphologically detailed cell
ChannelML
Voltage gated ion channel
Ligand gated ion channel
Ion pools
SynapseML
Electrical synapse
Synaptic conductance
Plasticity
Multicompartment Model
Morphology
Metadata
Channel behavior
Passive properties
Channel densities
Synapse behavior
3-D Network Model
Cell placement
Connectivity
Morphologies
Metadata
Channel behavior
Passive properties
Channel densities
Synapse behavior

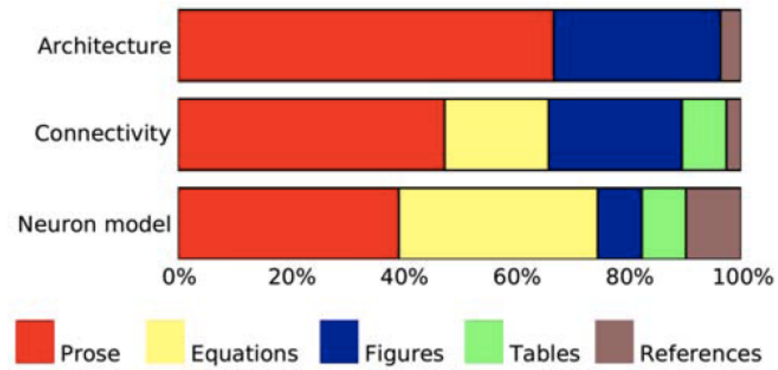
How can it be used with modeling tools?


NeuroML specification (XSD file)
VALID
INVALID
NeuroML description of synaptic mechanism (XML file)
<synapse_type name = "DoubExpSyn">
<meta:notes>Synaptic mechanism featuring a conductance which changes as double exponential function of time</meta:notes>
<doub_exp_syn max_conductance="1.0E-5" rise_time="1" decay_time="2" reversal_potential="0"/>
</synapse_type>
NEURON Mapping (XSL file)
NMODL file containing mechanism implementation (based on Exp2Syn)
GENESIS Mapping (XSL file)
GENESIS script file containing mechanism implementation (based on synchan object)
Other Simulator Mapping (XSL file)
File containing mechanism implemented in other simulator's scripting language
HTML Mapping (XSL file)
Maximum Conductance
The peak conductance which the synapse will reach
1.0E-5 mS
Rise Time
Time constant of rising component of synaptic waveform
1 ms
Decay Time
Time constant of decaying component of synaptic waveform
2 ms
Reversal Potential
The effective reversal potential for the ion flow through the synapse when the conductance is non zero
0 mV
x 10^-5
Synaptic Conductance [mS]
Time [ms]
0
0.2
0.4
0.6
0.8
1
2
4
6
8
10

Nordlie et al.(2009) *PLoS Computational Biology*: Towards reproducible descriptions of neuronal network models



Nordlie et al.(2009) *PLoS Computational Biology*: Towards reproducible descriptions of neuronal network models



A	Model Summary
Populations	Three: excitatory, inhibitory, external input
Topology	—
Connectivity	Random convergent connections
Neuron model	Leaky integrate-and-fire, fixed voltage threshold, fixed absolute refractory time (voltage clamp)
Channel models	—
Synapse model	δ-current inputs (discontinuous voltages jumps)
Plasticity	—
Input	Independent fixed-rate Poisson spike trains to all neurons
Measurements	Spike activity

B		Populations
Name	**Elements**	**Size**
E	Iaf neuron	$N_E = 4N_I$
I	Iaf neuron	N_I
E_{ext}	Poisson generator	$C_E(N_E + N_I)$

C			Connectivity
Name	**Source**	**Target**	**Pattern**
EE	E	E	Random convergent $C_E \to 1$, weight J, delay D
IE	E	I	Random convergent $C_E \to 1$, weight J, delay D
EI	I	E	Random convergent $C_I \to 1$, weight $-gJ$, delay D
II	I	I	Random convergent $C_I \to 1$, weight $-gJ$, delay D
Ext	E_{ext}	$E \cup I$	Non-overlapping $C_E \to 1$, weight J, delay D

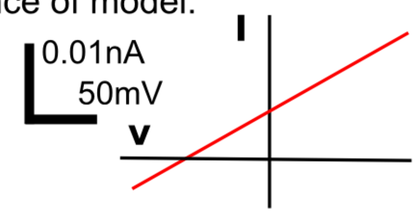
NeuroML v2 is built on LEMS

General model: **I = g * (v - E)**

I current through ion channel
g conductance of channel
v potential across channel
E reversal potential

Specific instance of model:

g = 10 pS
E = -60 mV



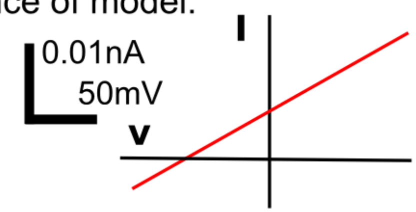
Cannon et al. 2014, LEMS: A language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2. *Frontiers in Neuroinformatics*.

NeuroML v2 is built on LEMS

General model: **I = g * (v - E)**
I current through ion channel
g conductance of channel
v potential across channel
E reversal potential

Specific instance of model:

g = 10 pS
E = -60 mV



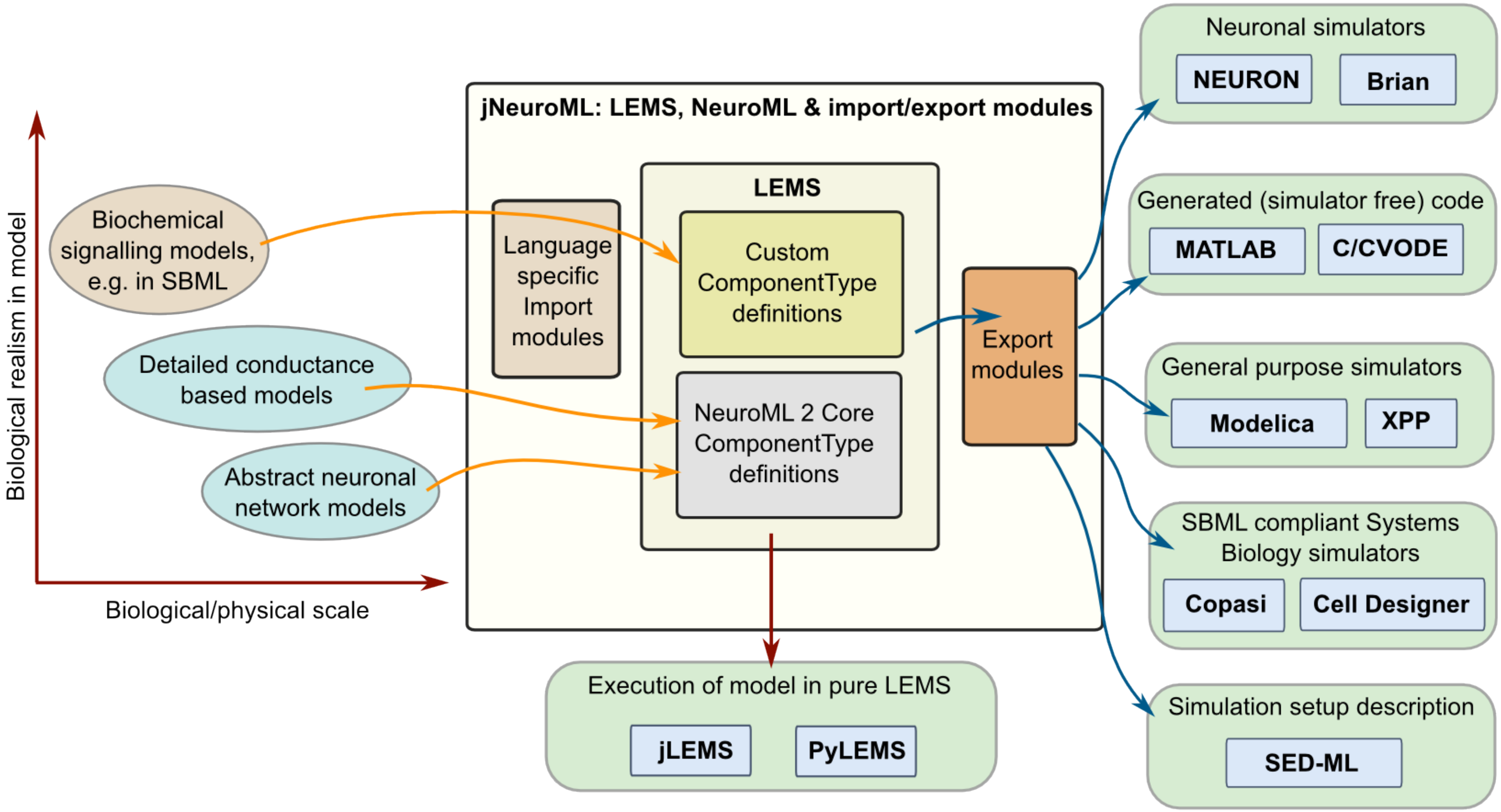
LEMS

```
<ComponentType name="passiveChannel">

    <Parameter name="g" dimension="conductance"/>
    <Parameter name="E" dimension="voltage"/>
    <Exposure name="I" dimension="current"/>
    <Requirement name="v" dimension="voltage"/>

    <Dynamics>
      <DerivedVariable name="I" dimension="current" exposure="I"
                       value="g * (E-v)"/>
    </Dynamics>

</ComponentType>
```

NeuroML

```
<passiveChannel g="10 pS" E="-60 mV"/>
```


Cannon et al. 2014, LEMS: A language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2. *Frontiers in Neuroinformatics*.



jNeuroML: LEMS, NeuroML & import/export modules
LEMS
Language specific Import modules
Custom ComponentType definitions
NeuroML 2 Core ComponentType definitions
Export modules
Biochemical signalling models, e.g. in SBML
Detailed conductance based models
Abstract neuronal network models
Biological realism in model
Biological/physical scale
Neuronal simulators
NEURON
Brian
Generated (simulator free) code
MATLAB
C/CVODE
General purpose simulators
Modelica
XPP
SBML compliant Systems Biology simulators
Copasi
Cell Designer
Simulation setup description
SED-ML
Execution of model in pure LEMS
jLEMS
PyLEMS

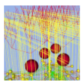
Specifications & examples

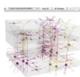
	XML SCHEMA	DOCUMENTATION	EXAMPLES	PUBLICATION
LEMS	LEMS_v0.7.1.xsd	LEMS element definitions	LEMS examples	Cannon et al. 2014
NeuroML v2beta3	NeuroML_v2beta3.xsd	NeuroML 2 Core ComponentTypes (Source)	NeuroML 2 examples (OSB models)	Cannon et al. 2014
NeuroML v1.8.1	NeuroML v1.8.1 Schemas	Specifications	NeuroML v1.x examples (OSB models)	Gleeson et al. 2010

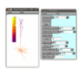
Implementations & APIs in Java and Python

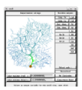
	JAVA	PYTHON
Read, validate & execute LEMS XML files	jLEMS	PyLEMS (Vella et al. 2014)
Read & write NeuroML 2 files	Java API for NeuroML 2	libNeuroML (Vella et al. 2014)
Everything...	jNeuroML *Parse & execute LEMS; validate NeuroML v1/v2; convert LEMS to graphical format, NEURON, Brian, etc.; convert SBML to LEMS...*	

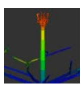
Vella et al. 2014 *Frontiers in Neuroinformatics*.

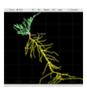
 NeuroML Validator

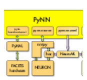
 neuroConstruct

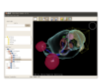
 Neuron

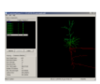
 GENESIS

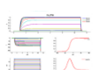
 MOOSE

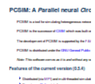
 PSICS

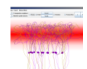
 PyNN

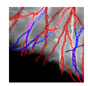
 Whole Brain Catalog

 NeuronLand

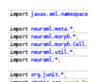
 Channelpedia

 PCSim

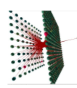
 CX3D

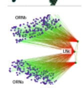
 Neuromantic

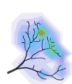
Neurospaces/GENESIS 3

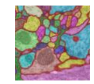
 NeuroML Java API

SplitNeuron

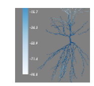
 NeurAnim

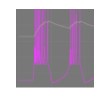
 CNrun

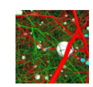
 Trees Toolbox

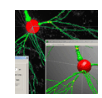
 TrakEM2

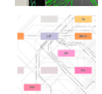
 NeuroMorpho.org

 Neuronvisio

 LEMS Interpreter

 NETMORPH

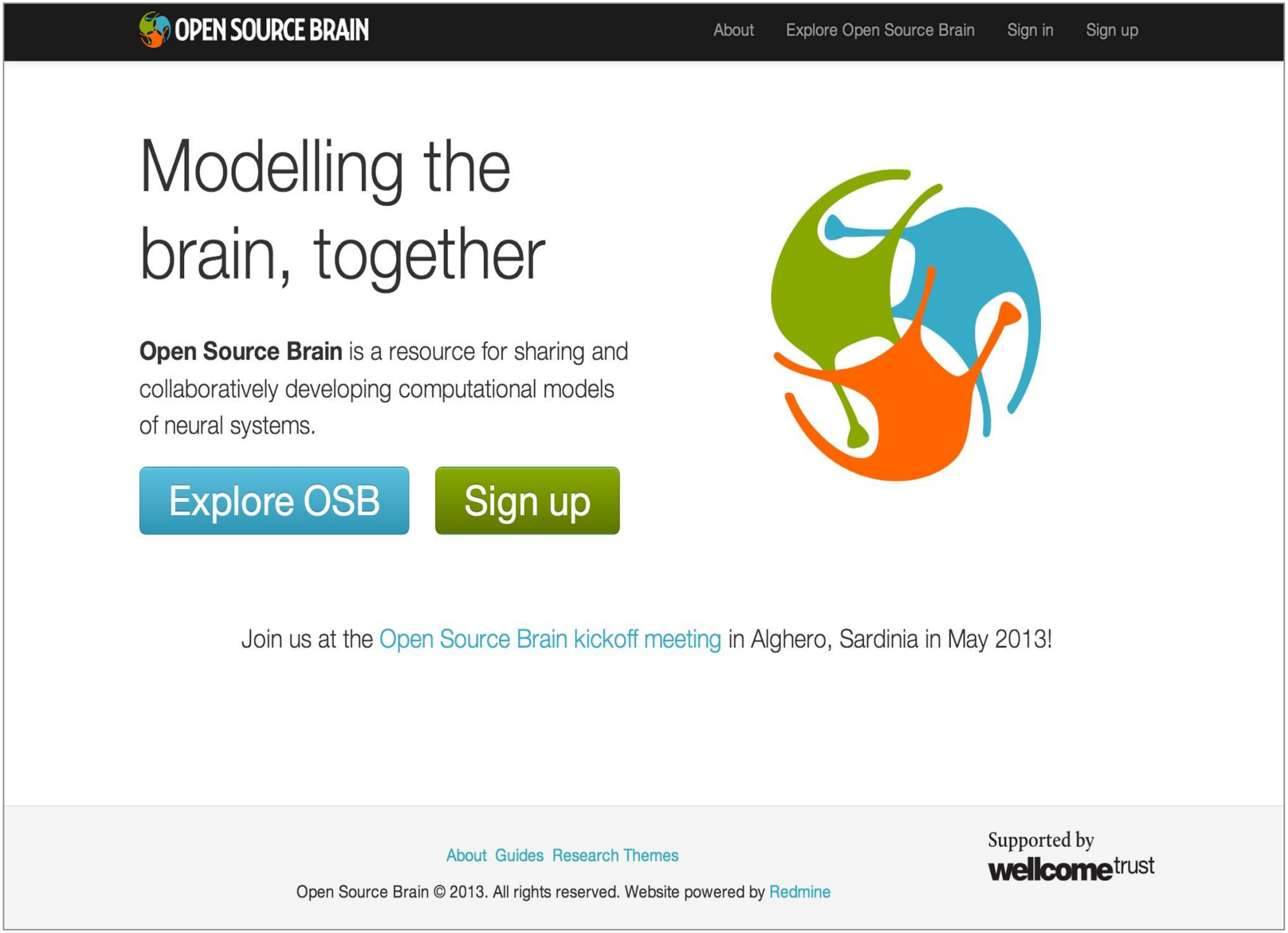
 NeuronStudio

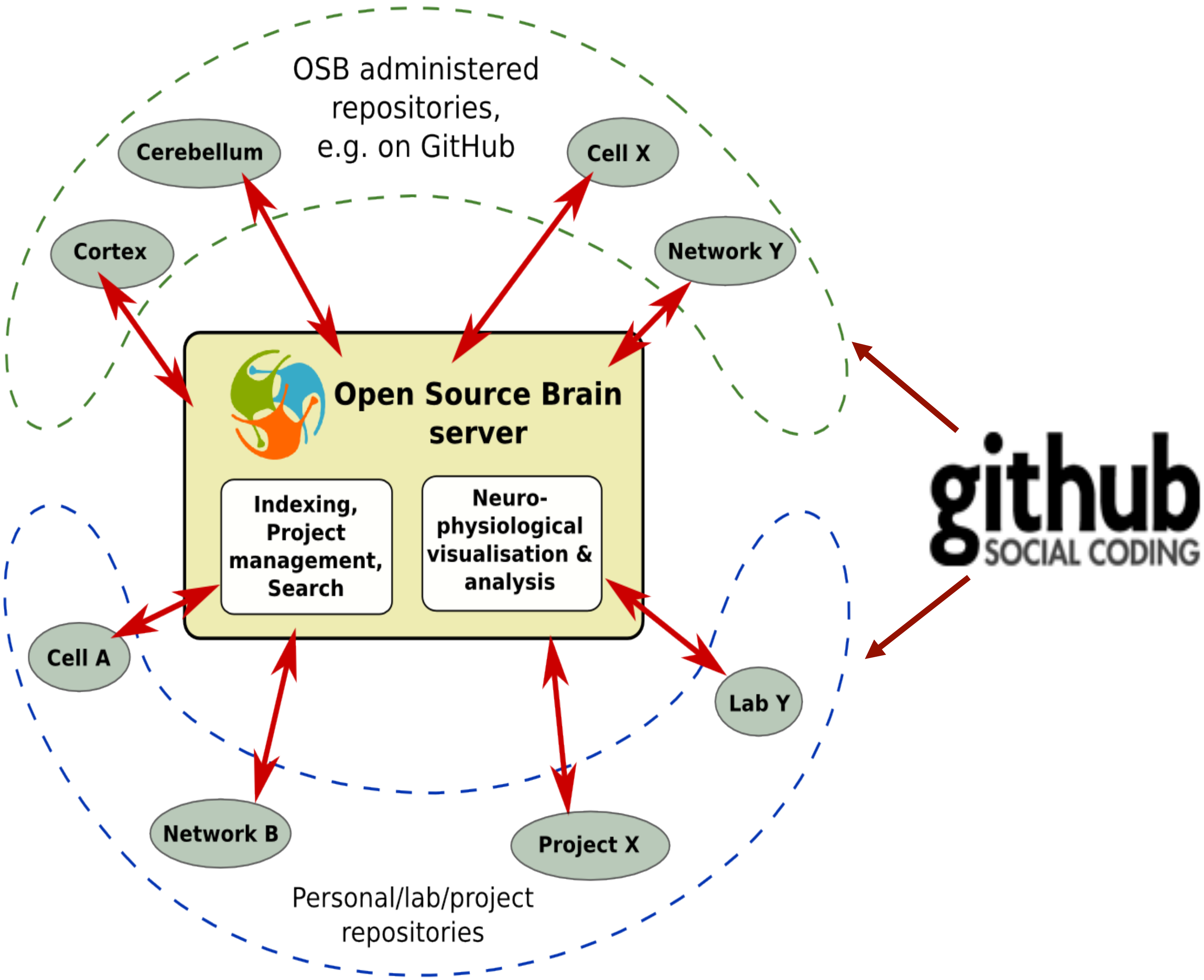
 Neuroptikon

 NeuroRD

OpenWorm

How does it support collaborative modeling?


OPEN SOURCE BRAIN
About
Explore Open Source Brain
Sign in
Sign up
Modelling the brain, together
Open Source Brain is a resource for sharing and collaboratively developing computational models of neural systems.
Explore OSB
Sign up
Join us at the Open Source Brain kickoff meeting in Alghero, Sardinia in May 2013!
About Guides Research Themes
Open Source Brain © 2013. All rights reserved. Website powered by Redmine
Supported by
wellcome trust


OSB administered repositories, e.g. on GitHub
Cerebellum
Cell X
Cortex
Network Y
Open Source Brain server
Indexing, Project management, Search
Neuro-physiological visualisation & analysis
github
SOCIAL CODING
Cell A
Lab Y
Network B
Project X
Personal/lab/project repositories

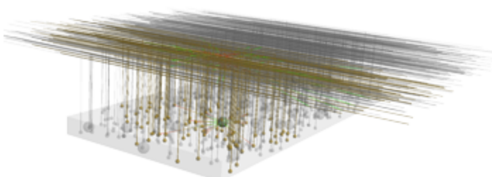
Granule Cell Layer Maex and De Schutter 1998

OSB endorsed project

Curation against published models: Good ★★★

Description
Status
Members
References

Description



An extension in 3D of the Granule Cell Layer model from: Maex, R and De Schutter, E. Synchronization of Golgi and Granule Cell Firing in a Detailed Network Model of the Cerebellar Granule Cell Layer J Neurophysiol, Nov 1998; 80: 2521 - 2537.

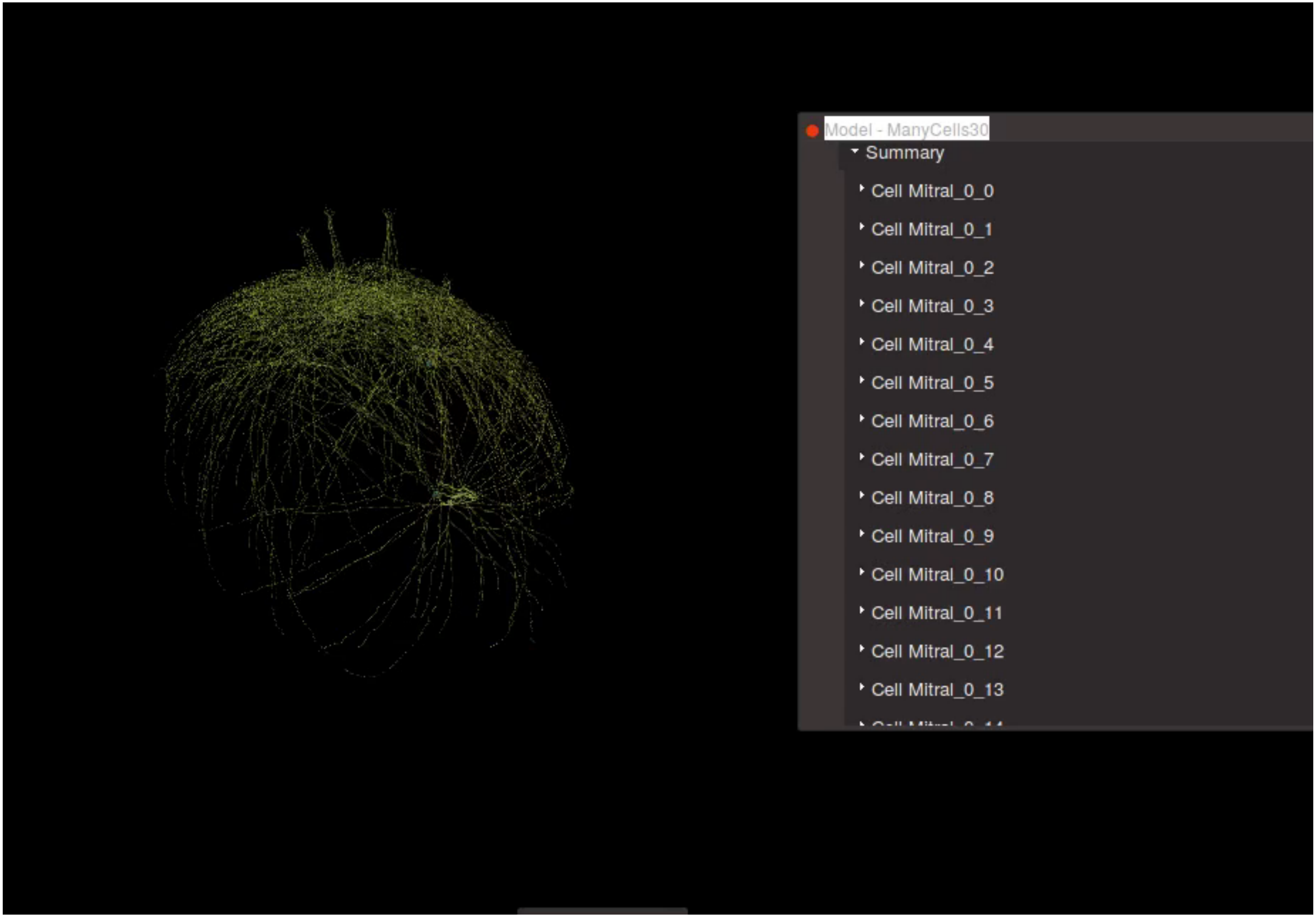
More...

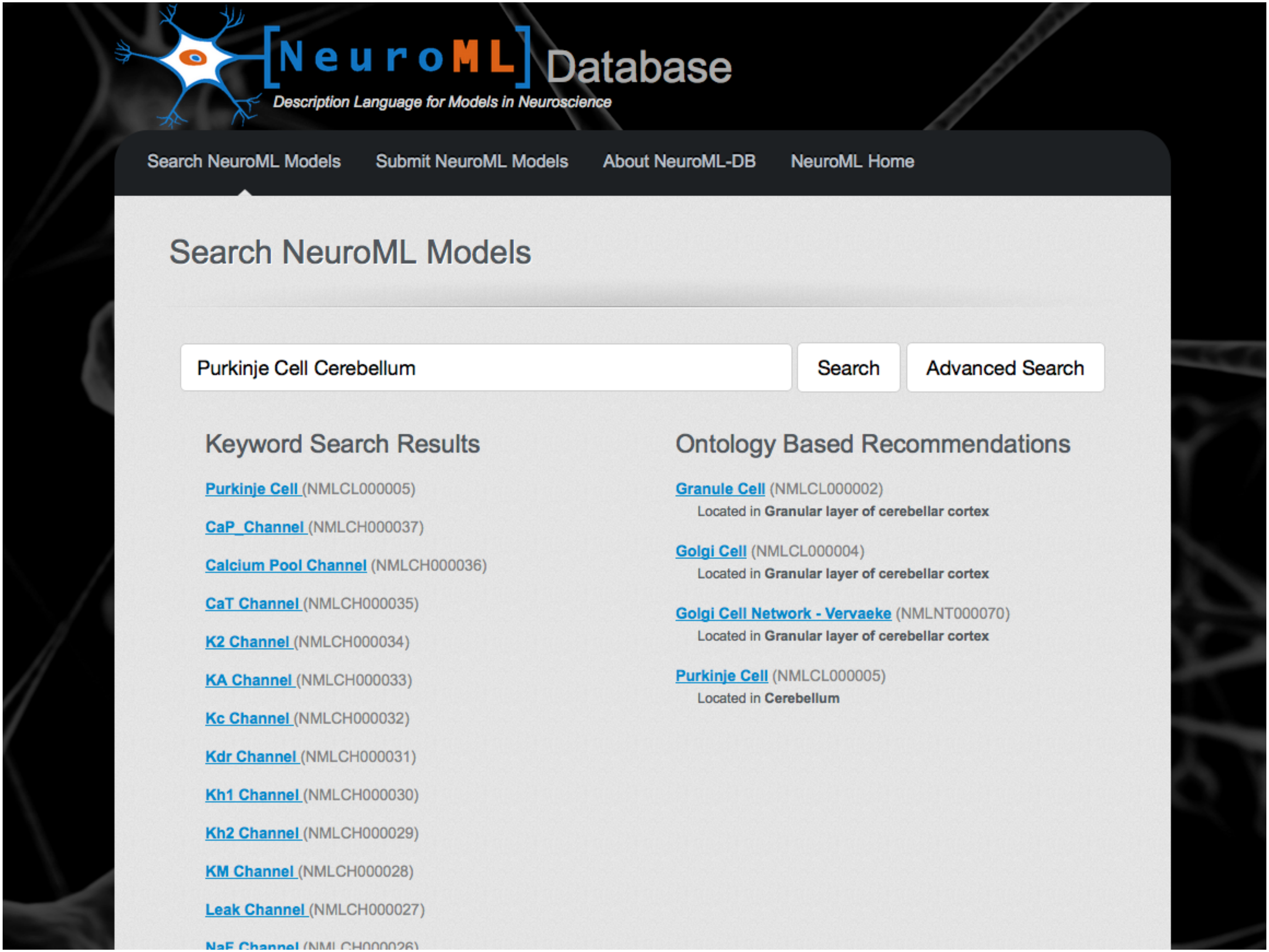
Status

Stable implementation on NEURON and GENESIS

NeuroML v2.x support ★ NeuroML v1.x support ★★★ PyNN support ?

NEURON support ★★★ GENESIS 2 support ★★★ MOOSE support ★ Brian support ?


Model - ManyCells30
Summary
Cell Mitral_0_0
Cell Mitral_0_1
Cell Mitral_0_2
Cell Mitral_0_3
Cell Mitral_0_4
Cell Mitral_0_5
Cell Mitral_0_6
Cell Mitral_0_7
Cell Mitral_0_8
Cell Mitral_0_9
Cell Mitral_0_10
Cell Mitral_0_11
Cell Mitral_0_12
Cell Mitral_0_13



[NeuroML] Database
Description Language for Models in Neuroscience
Search NeuroML Models
Submit NeuroML Models
About NeuroML-DB
NeuroML Home
Search NeuroML Models
Purkinje Cell Cerebellum
Search
Advanced Search
Keyword Search Results
Purkinje Cell (NMLCL000005)
CaP_Channel (NMLCH000037)
Calcium Pool Channel (NMLCH000036)
CaT Channel (NMLCH000035)
K2 Channel (NMLCH000034)
KA Channel (NMLCH000033)
Kc Channel (NMLCH000032)
Kdr Channel (NMLCH000031)
Kh1 Channel (NMLCH000030)
Kh2 Channel (NMLCH000029)
KM Channel (NMLCH000028)
Leak Channel (NMLCH000027)
Ontology Based Recommendations
Granule Cell (NMLCL000002)
Located in Granular layer of cerebellar cortex
Golgi Cell (NMLCL000004)
Located in Granular layer of cerebellar cortex
Golgi Cell Network - Vervaeke (NMLNT000070)
Located in Granular layer of cerebellar cortex
Purkinje Cell (NMLCL000005)
Located in Cerebellum

[NeuroML] Database

Description Language for Models in Neuroscience

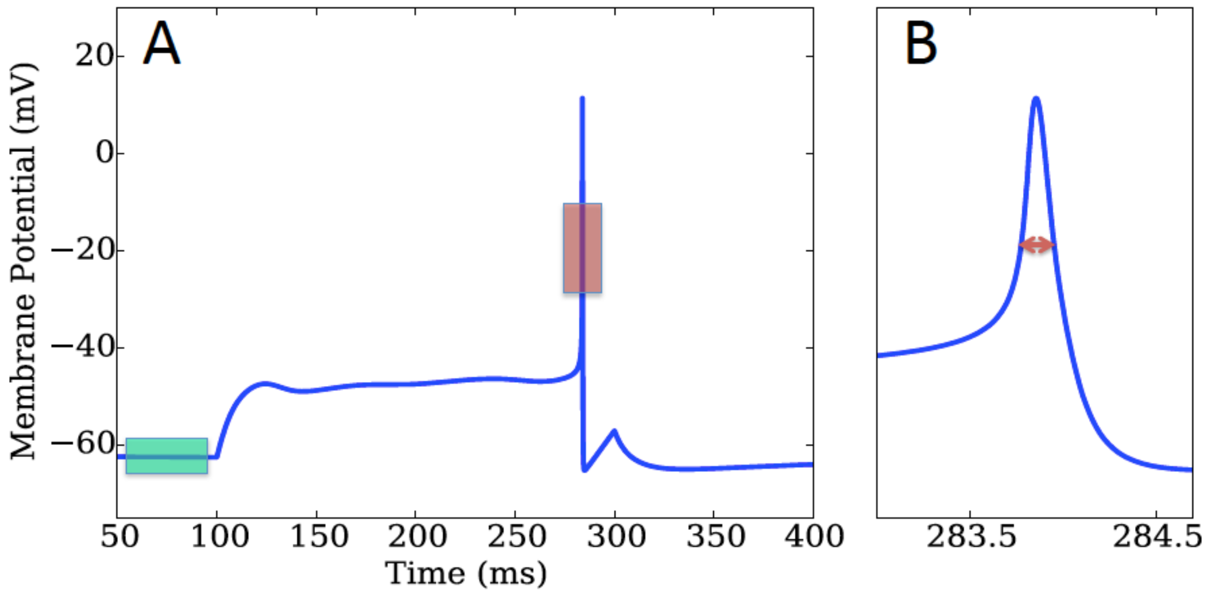
Search NeuroML Models Submit NeuroML Models About NeuroML-DB NeuroML Home

Model Information

Model Id:	NMLCL000002 *(Obtained from Open Source Brain)*
Model Type:	Cell
Model Name:	Granule Cell
Authors:	Reinoud Maex Erik De Schutter
Translators:	Padraig Gleeson
Publication:	Synchronization of Golgi and granule cell firing in a detailed network model of the cerebellar granule cell layer
Neurolex IDs:	Cerebellum granule cell, Cerebellar, neuron
References:	ModelDB Open Source Brain
Keywords:	None
Associated Channels:	Gran_NaF_98 Channel Gran_KDr_98 Channel Gran_KCa_98 Channel Gran_KA_98 Channel Gran_H_98 Channel Gran_CaPool_98 Channel Gran_CaHVA_98 Channel GranPassiveCond Channel
Associated Synapses:	AMPA_GranGol GABAA MF_AMPA NMDA
Associated Networks:	Cerebellar Granule Layer Network
NeuroML Files	/var/www/NeuroMLmodels/NMLCL000002/Granule_98.morph.xml

Where is NeuroML going next?

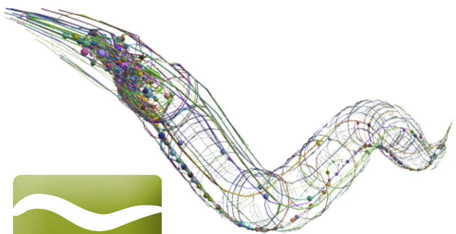
NeuronUnit

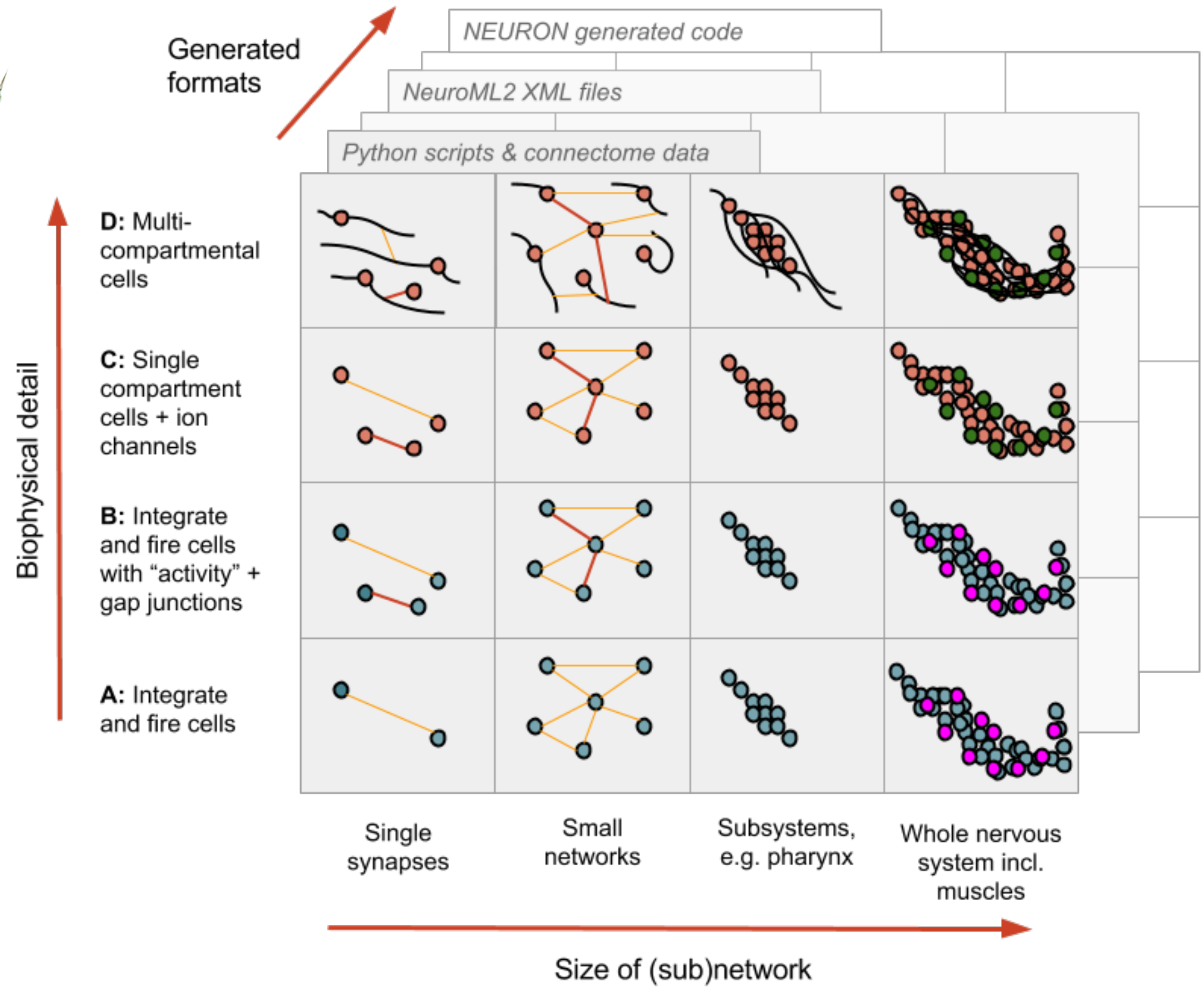


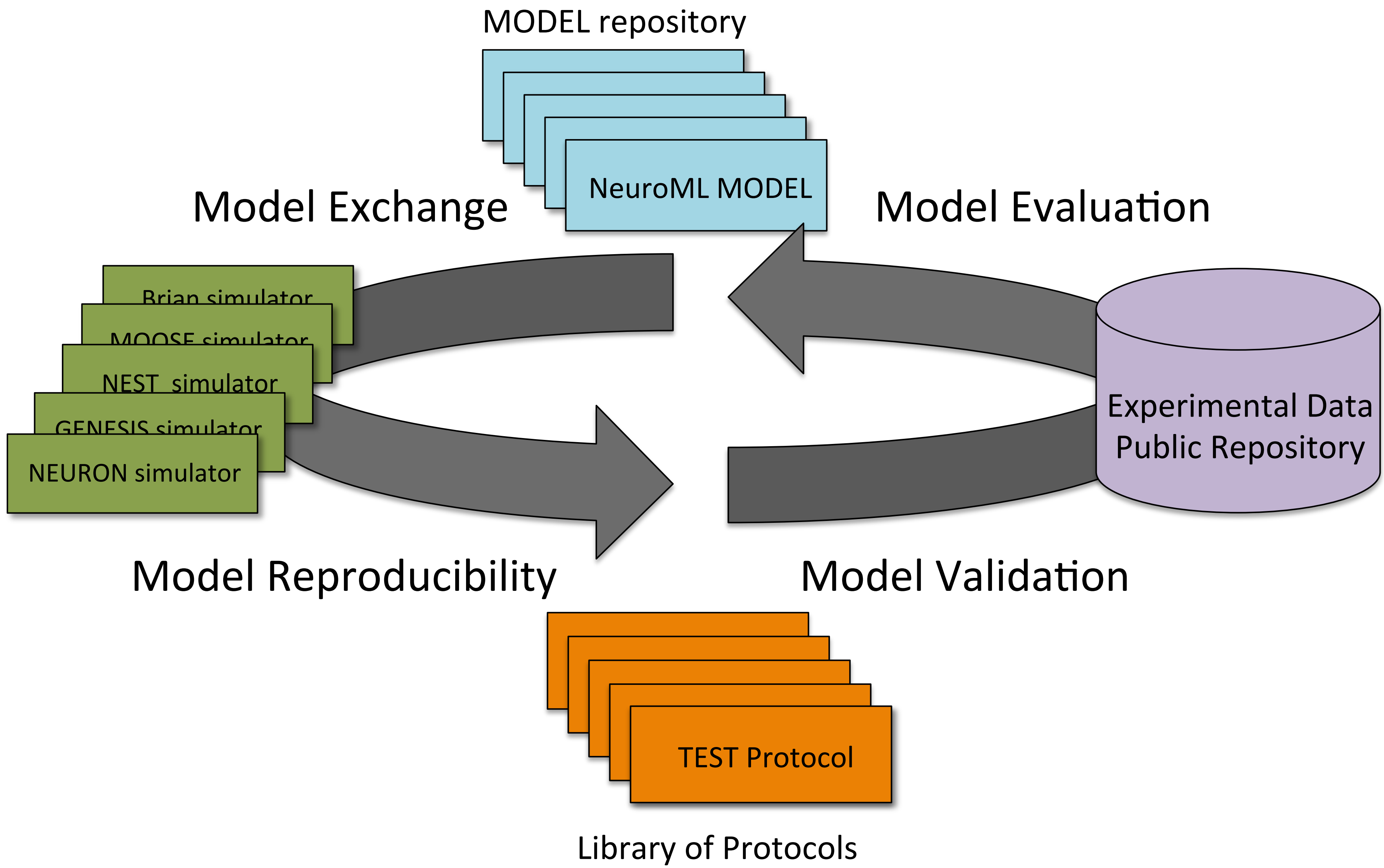
C

Granule Cell Models from OSB	Tests from NeuroElectro data	
	Resting Potential	Spike Width
Berends 2005	+1.15	**+0.18**
Maex 1998	+0.89	-1.71
Diwakar 2009	+1.10	*n/a*
Solinas 2010	**+0.13**	*-3.18*

c302 - multiscale simulation framework for C. elegans






MODEL repository
NeuroML MODEL
Model Exchange
Model Evaluation
Brian simulator
MOOSE simulator
NEST simulator
GENESIS simulator
NEURON simulator
Experimental Data
Public Repository
Model Reproducibility
Model Validation
TEST Protocol
Library of Protocols

What is NeuroML?

- Well-tested description language for multiscale models in neuroscience, including ion channels, dynamic synapses, morphologies, abstract cell models, and 3D network structure.

Why do we need it?

- Important for model sharing, efficient model development, scientific reproducibility, and collaborative modeling.

How can it be used with modeling tools?

- Over 40 simulators, databases, visualization tools, applications and libraries provide support for NeuroML, offering innovative pipelines.

How does NeuroML support collaborative modeling?

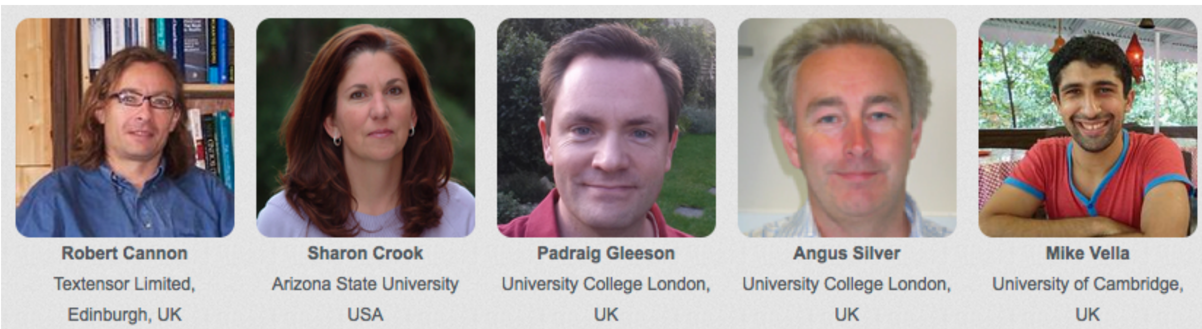
- NeuroML-DB and Open Source Brain provide models and tools for model evaluation, re-use, and collaborative development.

Where is it going next?

- Automated model validation and extensions/tools needed by the community.

[NeuroML] http://neuroml.org

NeuroML Editorial Board



New Board Members:
Andrew Davison
Upi Bhalla

Funding

UK Group: Medical Research Council, EUSynapse, Wellcome Trust
Workshop Funding: National Science Foundation, INCF
US Group: National Institutes of Health: NIMH, NIBIB