

# The NeuroML ecosystem for standardised multi-scale modelling in neuroscience

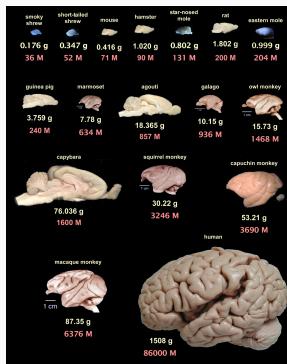
Ankur Sinha  
Silver Lab  
Department of Neuroscience, Physiology, & Pharmacology  
University College London

2024-02-26

## Notes

1 / 34

## An understanding of the brain



- ~86B neurons
- ~100T synapses
- also ~85B glia
- specialised circuits
- different neuronal types
- synaptic connections
- complex sub-cellular processes

<sup>1</sup>Herculano-Houzel, S. The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in human neuroscience* 3, 31 (2009)

<sup>2</sup>von Bartheld, C. S. et al. The search for true numbers of neurons and glial cells in the human brain: A review of 150 years of cell counting. *Journal of Comparative Neurology* 524, 3865–3895. ISSN: 1096-9861 (June 2016)

## Notes

2 / 34

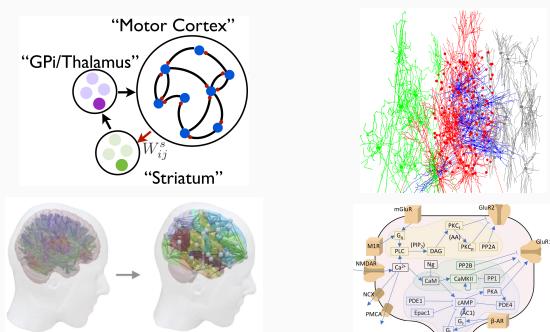
## Models complement experimental neuroscience

- Combine individual experimental results into unified theories
- Explore generalisability of experimental results over wider range of conditions
- Generate new experimentally testable, physically plausible hypotheses: dictate experiment design

## Notes

3 / 34

## Models: different scales



<sup>1</sup>Murray, J. M. Local online learning in recurrent networks with random feedback. *eLife* 8 (eds Latham, P. et al.) e43299. ISSN: 2050-084X (2019)

Schirmer, M. et al. Learning how network structure shapes decision-making for bio-inspired computing. *Nature Communications* 14, issn: 2041-1723 (May 2023)

<sup>2</sup>Yao, H. K. et al. Reduced inhibition in depression impairs stimulus processing in human cortical microcircuits. *Cell Reports* 38, issn: 2211-1242.

<sup>3</sup>Mäki-Marttunen, T. et al. A unified computational model for cortical post-synaptic plasticity. *eLife* 9 (eds Shouval, H. Z. et al.) e55714. ISSN: 2050-084X.

<https://doi.org/10.7554/eLife.55714> (July 2020)

## Notes

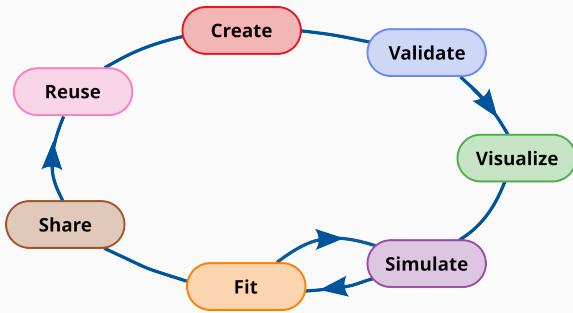
4 / 34

A **mechanistic** understanding of the brain requires **biophysically detailed** modelling

Notes

5 / 34

## The model life cycle



Notes

6 / 34

## Computational modelling software ecosystem is fragmented

- many specialist tools:
  - NEURON, NEST, Brian, GENESIS, MOOSE, STEPS, ANNarchy, TVB, LFPy, NeuroLib, EDEN, Arbor, NetPyNE...
- but:
  - different APIs, syntax:
    - increased difficulty for users
  - not well defined model descriptions:
    - models cannot be easily validated
  - custom machine readable internal representations:
    - cannot be easily inspected/analysed
  - ad-hoc utilities:
    - cannot be used with all tools

Notes

7 / 34

Makes computational neuroscience models  
less  
FAIR  
(Findable, Accessible, Interoperable, Reusable)

Notes

8 / 34

## Standards enable FAIR neuroscience



<sup>1</sup> Abrams, M. B. et al. A Standards Organization for Open and FAIR Neuroscience: the International Neuroinformatics Coordinating Facility. *Neuroinformatics* 20, 25–36. ISSN: 1559-0289. <https://doi.org/10.1007/s12081-020-09509-1> (2022). <https://incf.org/>

<sup>2</sup> Computational Modeling in Biology Network (COMBINE) <https://co.mbine.org/>

9 / 34

## Notes

---

---

---

---

---

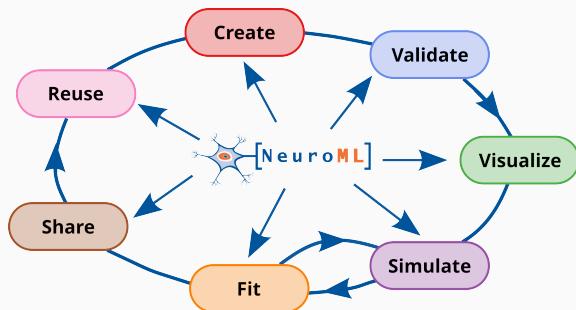
---

---

---

---

## NeuroML ecosystem supports all stages of the model cycle



10 / 34

## Notes

---

---

---

---

---

---

---

---

---

## NeuroML ecosystem

- standard/specification
- software ecosystem

11 / 34

## Notes

---

---

---

---

---

---

---

---

---

## NeuroML standard

### Model specification (schema: XSD)

- elements
- attributes
- hierarchical relationships

### Dynamics (LEMS component type definitions)

- dynamical behaviour

## Notes

---

---

---

---

---

---

---

---

---

12 / 34

## NeuroML standard: schema: XSD

Way of specifying the structure of an XML document.

- allows defining **types** and **extensions/restrictions** on types to create new types.
- allows generation of **APIs**

One can validate a model description against the schema before simulation

<sup>1</sup><https://www.w3.org/TR/xmlschema-1/>

## Notes

---

---

---

---

---

---

## NeuroML standard: schema: XSD

```
<xsd:simpleType name="Mdl2Quantity_voltage"> <!-- For params with dimension voltage -->
<xsd:restriction base="xsd:string">
<xsd:pattern value="-?([0-9]+(\.[0-9]+)?)([eE]-?[0-9]+)?([a]*(V|v))"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="izhikevich2007Cell">
<xsd:annotation>
  <xsd:documentation>Cell based on ...</xsd:documentation>
</xsd:annotation>
<xsd:complexContent>
  <xsd:extension base="BaseCellMembPotCap">
    <xsd:attribute name="v0" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="k" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="tau_m" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="vt" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="vpeak" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="a" type="Mdl2Quantity_pertime" use="required"/>
    <xsd:attribute name="b" type="Mdl2Quantity_constance" use="required"/>
    <xsd:attribute name="c" type="Mdl2Quantity_voltage" use="required"/>
    <xsd:attribute name="d" type="Mdl2Quantity_current" use="required"/>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

<sup>1</sup>Izhikevich, E. M. *Dynamical systems in neuroscience*. (MIT Press, 2007)

## Notes

---

---

---

---

---

---

## NeuroML standard: LEMS component type definitions

### Low Entropy Model Specification language

- domain independent
- allows creation of "Component Types" (**classes**) from which "Components" (**objects**) can be instantiated by providing the necessary parameters
- provides a **reference implementation/simulator**
- machine readable: **translatable** into other formats

<sup>1</sup>Cannon, R. C. et al. LEMS: a language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2. *Frontiers in Neuroinformatics* 8 (2014)

## Notes

---

---

---

---

---

---

## NeuroML standard: dynamics (LEMS)

```
<ComponentType name="izhikevich2007Cell" extends="baseCellMembPotCap"
  description="Cell based ...">

  <Parameter name="v0" dimension="voltage" description="Initial membrane potential"/>
  <!--
  Defined in baseCellMembPotCap:
  <Parameter name="C" dimension="capacitance"/>
  ...
  <Parameter name="k" dimension="conductance_per_voltage"/>
  -->
  <Parameter name="vr" dimension="voltage" description="Resting membrane potential"/>
  <Parameter name="vt" dimension="voltage" description="Spike threshold"/>
  <Parameter name="vpeak" dimension="voltage" description="Peak action potential value"/>

  <Parameter name="a" dimension="per_time" description="Time scale of recovery variable a"/>
  <Parameter name="b" dimension="conductance" description="Sensitivity of recovery variable u to subthreshold
  Ia / Ia_max" description="initial membrane potential v"/>
  <Parameter name="c" dimension="voltage" description="After-spike reset value of v"/>
  <Parameter name="d" dimension="current" description="After-spike increase to u"/>

  <Attachments name="synapses" type="basePointCurrent"/>
  <Exposure name="u" dimension="current" description="Membrane recovery variable"/>
  <Dynamics><!-- snipped --></Dynamics>
</ComponentType>
```

## Notes

---

---

---

---

---

---

## NeuroML standard: XSD and LEMS

XSD:

```
<x:attribute name="v0" type="Nml2Quantity_voltage" use="required"/>
<x:attribute name="k" type="Nml2Quantity_conductance_per_voltage" use="required"/>
<x:attribute name="vr" type="Nml2Quantity_voltage" use="required"/>
<x:attribute name="vt" type="Nml2Quantity_voltage" use="required"/>
<x:attribute name="a" type="Nml2Quantity_pertime" use="required"/>
<x:attribute name="b" type="Nml2Quantity_conductance" use="required"/>
<x:attribute name="c" type="Nml2Quantity_voltage" use="required"/>
<x:attribute name="d" type="Nml2Quantity_current" use="required"/>
```

LEMS:

```
<Parameter name="v0" dimension="voltage" description="Initial membrane potential"/>
<Parameter name="k" dimension="conductance_per_voltage"/>
<Parameter name="vr" dimension="voltage" description="Voltage at resting membrane potential"/>
<Parameter name="vt" dimension="voltage" description="Spike threshold"/>
<Parameter name="vpeak" dimension="voltage" description="Peak action potential value"/>
<Parameter name="a" dimension="per_time" description="Time scale of recovery variable u"/>
<Parameter name="b" dimension="conductance" description="Sensitivity of recovery variable u to subthreshold fluctuations of membrane potential v"/>
<Parameter name="c" dimension="voltage" description="After-spike reset value of v"/>
<Parameter name="d" dimension="current" description="After-spike increase to u"/>
```

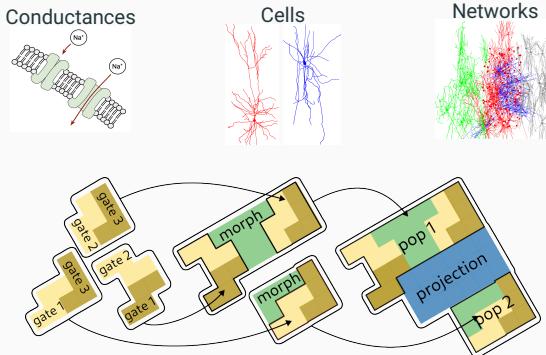
17 / 34

## NeuroML standard: dynamics (LEMS)

```
<ComponentType name="izhikevich2007Cell" extends="baseCellMembranePotCap"
  description="Cell based ...">
  <!-- snipped -->
  <Attachments name="synapses" type="basePointCurrent"/>
  <Exposure name="u" dimension="current" description="Membrane recovery variable"/>
  <Dynamics>
    <StateVariable name="v" dimension="voltage" exposure="v"/>
    <StateVariable name="u" dimension="current" exposure="u"/>
    <DerivedVariable name="iSyn" dimension="current" exposure="iSyn" select="synapses[*]/i" reduce="add" />
    <DerivedVariable name="iMemb" dimension="current" exposure="iMemb" value="k * (v-vr) + iSyn - u"/>
    <TimeDerivative variable="v" value="iMemb / C" />
    <TimeDerivative variable="u" value="a * (b * (v-vr) - u)"/>
  <OnStart>
    <StateAssignment variable="v" value="v0"/>
    <StateAssignment variable="u" value="0"/>
  </OnStart>
  <OnDefinition test="v . >= vpeak">
    <StateAssignment variable="u" value="c"/>
    <StateAssignment variable="u" value="u + d"/>
    <EventOut port="spike"/>
  </OnDefinition>
  <Dynamics>
  </Dynamics>
</ComponentType>
```

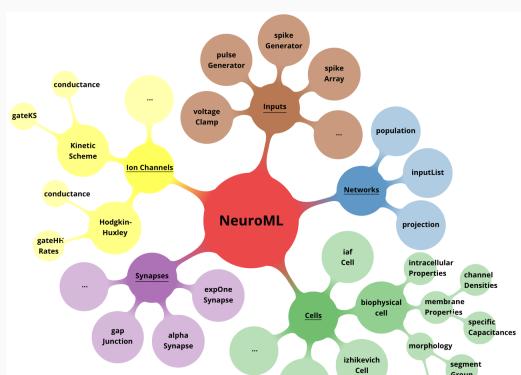
18 / 34

## NeuroML is declarative, modular, structured, hierarchical



19 / 34

## NeuroML provides users with a set of curated model elements



<sup>1</sup>Full standard is at: <https://docs.neuroml.org/Userdocs/Specification.html>

Notes

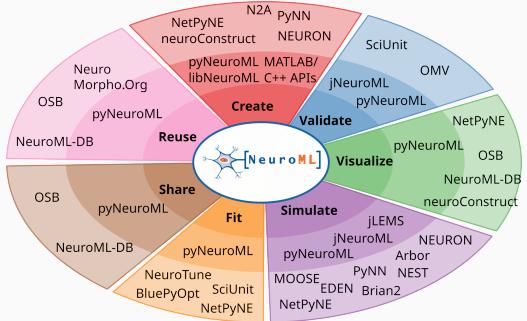
Notes

Notes

Notes

20 / 34

## NeuroML software ecosystem



21 / 34

## Notes

---

---

---

---

---

---

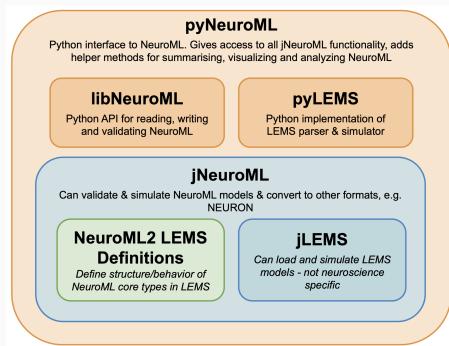
---

---

---

---

## NeuroML software ecosystem: core tools



pip install pyneuroml

22 / 34

## Notes

---

---

---

---

---

---

---

---

---

---

## NeuroML: creating/simulating models

Python script to create a new network, and validate it:

```
from neuroml import * # NeuroML API libNeuroML

newdoc = NeuroMLDocument(id="new_doc")
newcell = IafTauCell(id="cell_0", leakReversal="-60mV", thresh="0mV", tau="5ms", reset="-70mV")
newdoc.add(newcell)

network = newdoc.add(Network, id="new_net", validate=False)
population = network.add(Population, id="new_pop", size=10, component=newcell.id)

# Helper method to ensure all parameters
# present and appropriate
newdoc.validate(recursive=True)
```

Resultant NeuroML XML serialization:

```
<neurone id="new_doc">
<iafTauCell id="cell_0" leakReversal="-60mV" thresh="0mV" reset="-70mV" tau="5ms"/>
<network id="new_net">
<population id="new_pop" component="cell_0" size="10"/>
</network>
</neurone>
```

23 / 34

## Notes

---

---

---

---

---

---

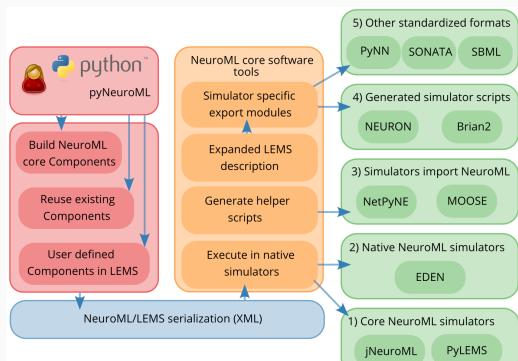
---

---

---

---

## NeuroML: creating/simulating models



24 / 34

## Notes

---

---

---

---

---

---

---

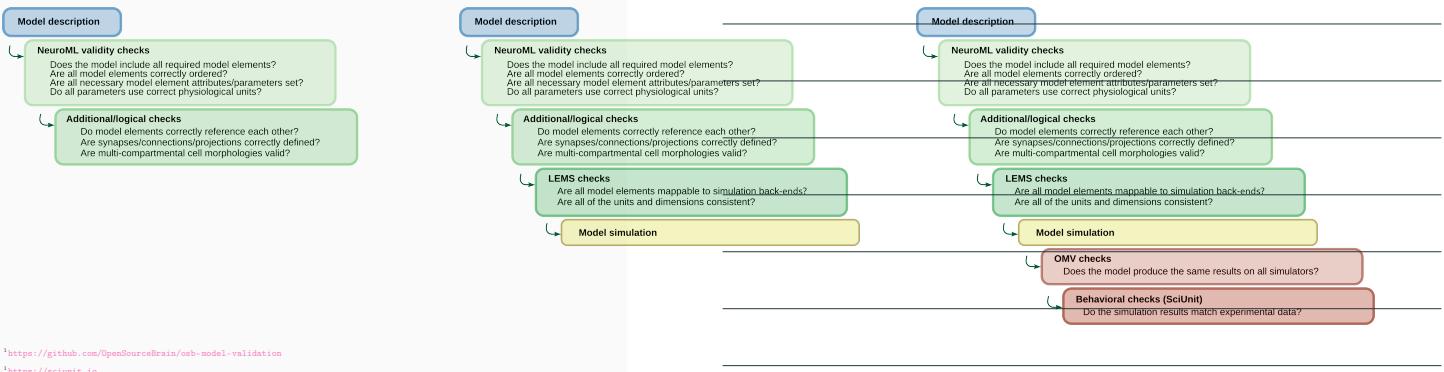
---

---

---

## NeuroML: validating models

### Notes



<sup>1</sup><https://github.com/OpenSourceBrain/esb-model-validation>

<sup>2</sup><https://sciumit.io>

25 / 34

## NeuroML: visualising/analysing models

### Notes



<sup>1</sup>3D interactive visualisation of Migliore et al. [10] using pymml-pplotmorph

<sup>2</sup>3D interactive visualisation of Sadot et al. [11] on Open Source Brain: <http://v1.opensourcebrain.org>

<sup>3</sup>3D interactive visualisation using NetPyNE-UI on Open Source Brain v2: <https://opensourcebrain.org>

\*Connectivity metrics for NeuroML conversion of Yao et al. [3]

26 / 34

## NeuroML: fitting models

### Notes

- Figure from docs
- Mention inspyred

27 / 34

## NeuroML: sharing and re-using models

### Notes

- GitHub, OSBv1, OSBv2, NeuroML-DB

28 / 34

- Jupyterbook

29 / 34

## Notes

---

---

---

---

---

---

---

---

- Editorial board, scientific committee, communication channels

30 / 34

## Notes

---

---

---

---

---

---

---

---

- Open Source Brain Video

31 / 34

## Notes

---

---

---

---

---

---

---

---

- GSoC, Outreachy, good computer science students

32 / 34

## Notes

---

---

---

---

---

---

---

---

Sinha, A. et al. **The NeuroML ecosystem for standardized multi-scale modeling in neuroscience.** *bioRxiv*. eprint:  
[https://www.biorxiv.org/content/early/2023/12/11/2023.12.07.570537](https://www.biorxiv.org/content/early/2023/12/11/2023.12.07.570537.full.pdf) (2023)(in review)

<https://docs.neuroml.org>  
<https://opensourcebrain.org>

33 / 34

- Python API

34 / 34

### Notes

### Notes

### Notes

### Notes