

Standards and Tools in Neuroscience

A summary of the Open Source Brain workshop,
September 2019

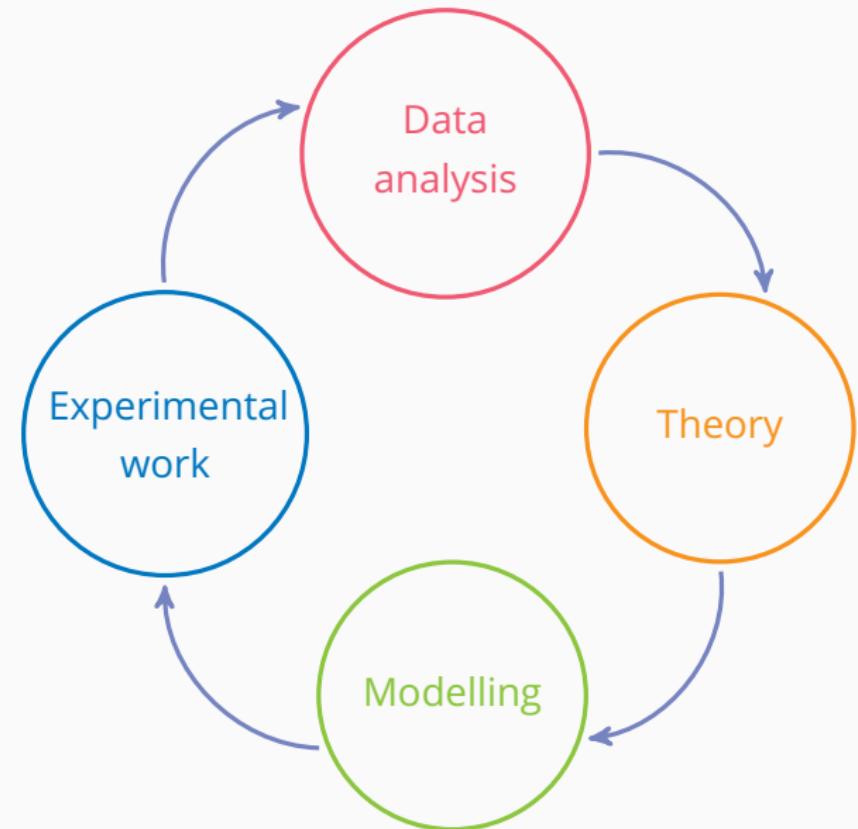
Ankur Sinha

Ph.D. candidate: UH Biocomputation Group, UK,
Volunteer: Fedora Project.

2019-10-04

The problem statement

Neuroscience is complex, and massive



Journal club

└ The problem statement

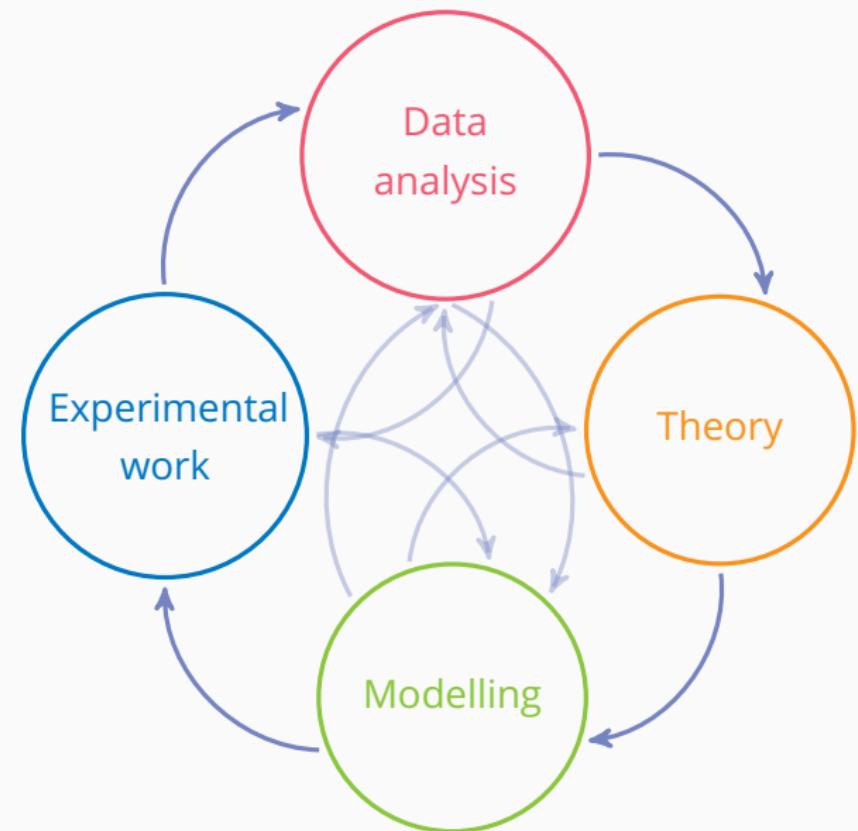
└ Neuroscience is complex, and massive

1. A simplified diagram. Actually a lot more complex
2. It is so massive that you can speak to another neuroscientist and not understand a word of what they say—we specialise
3. We won't even discuss dissemination to a non scientific audience today—a completely different problem.

Neuroscience is complex, and massive



Neuroscience is complex, and massive



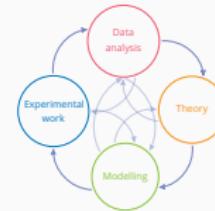
Journal club

└ The problem statement

└ Neuroscience is complex, and massive

1. A simplified diagram. Actually a lot more complex
2. It is so massive that you can speak to another neuroscientist and not understand a word of what they say—we specialise
3. We won't even discuss dissemination to a non scientific audience today—a completely different problem.

Neuroscience is complex, and massive



Free/Open science:
Scientific material should be easily, openly **accessible to all**.

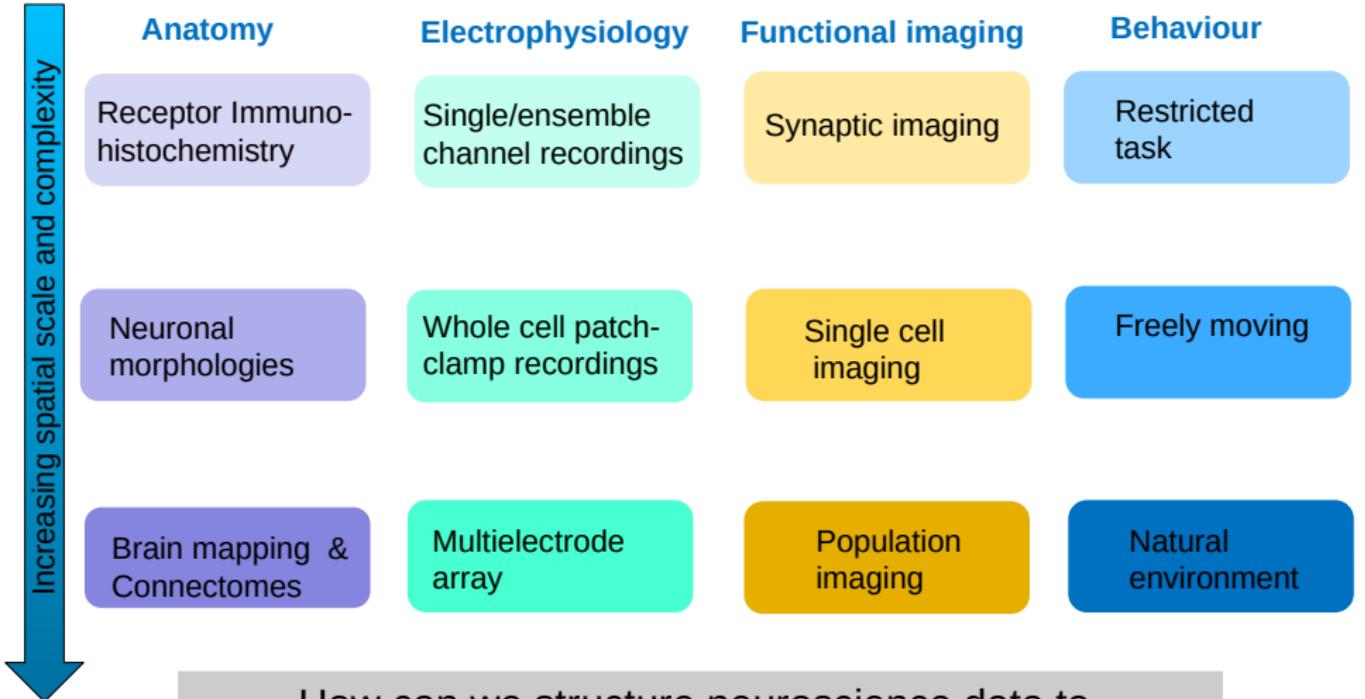
Journal club

└ The problem statement

└ Free/Open Neuroscience

2019-10-04

Experimental neuroscience data is heterogeneous, multiscale and analysis is complex



2019-10-04

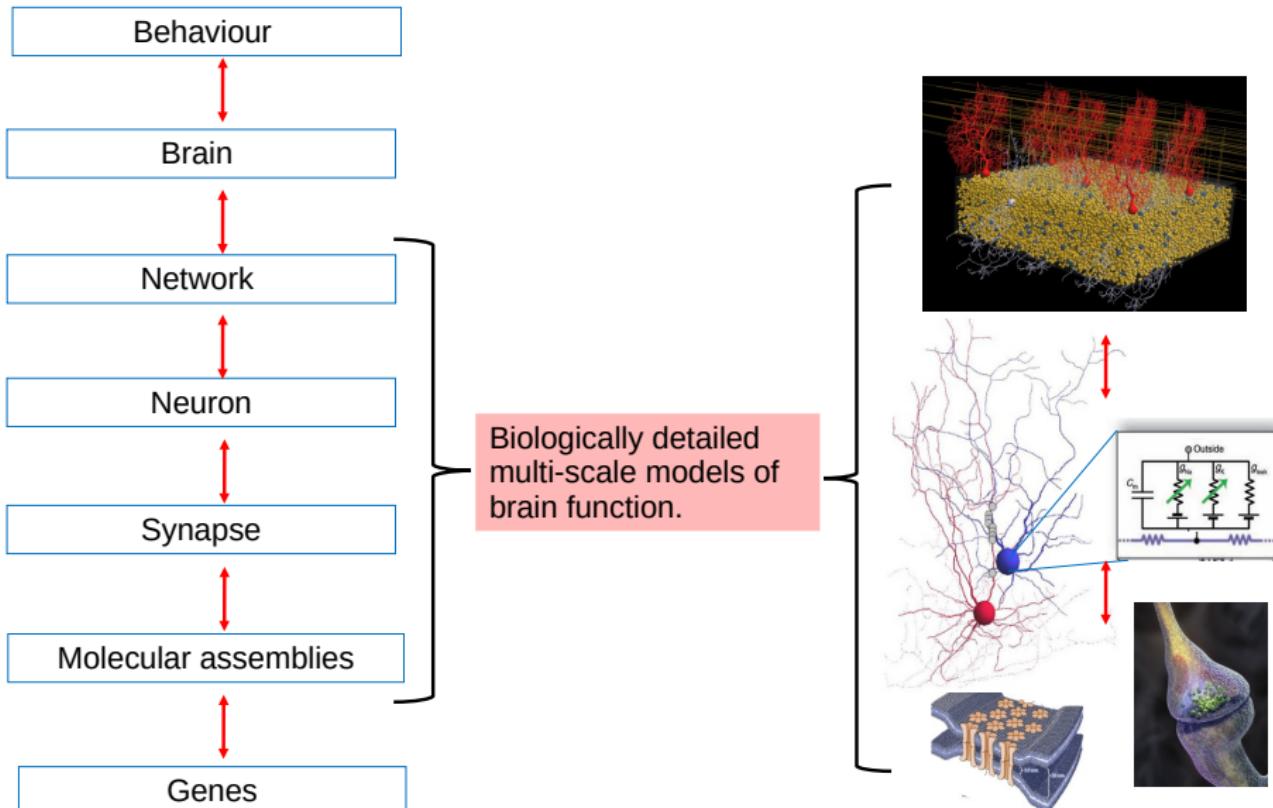
Journal club

└ The problem statement

└ Free/Open Neuroscience

Free/Open science:
Scientific material should be easily, openly [accessible to all](#).

Models of brain function span multiple spatial scales



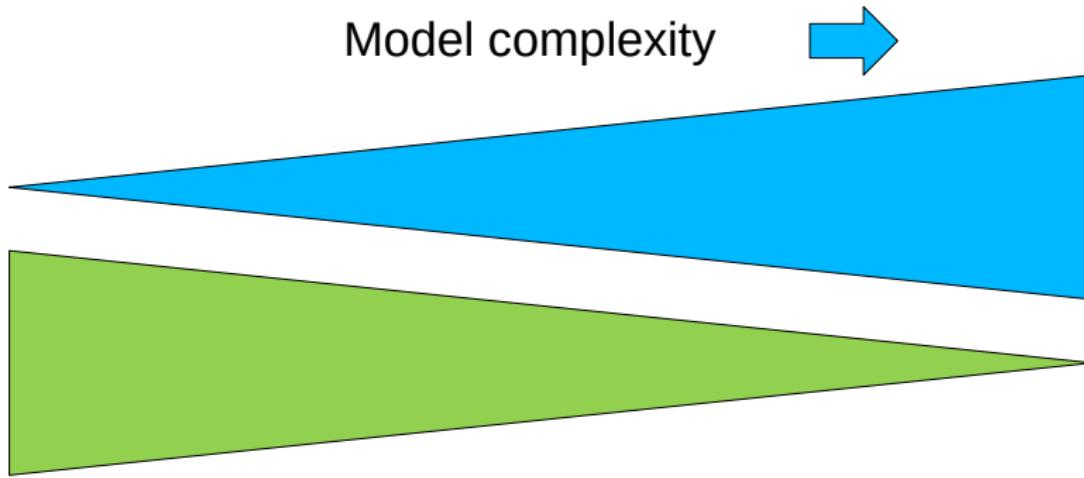
Journal club

└ The problem statement

2019-10-04

└ Free/Open Neuroscience

A scaling problem



Model complexity →
← Transparency, accessibility, reproducibility,
reuse.....and utility as a scientific tool

Journal club

└ The problem statement

2019-10-04

└ Free/Open Neuroscience

Free/Open science:
Scientific material should be easily, openly [accessible to all](#).

2019-10-04

Standards: the common tongue

NWB:N 2.0: An Ecosystem for Neurophysiology Data Standardization

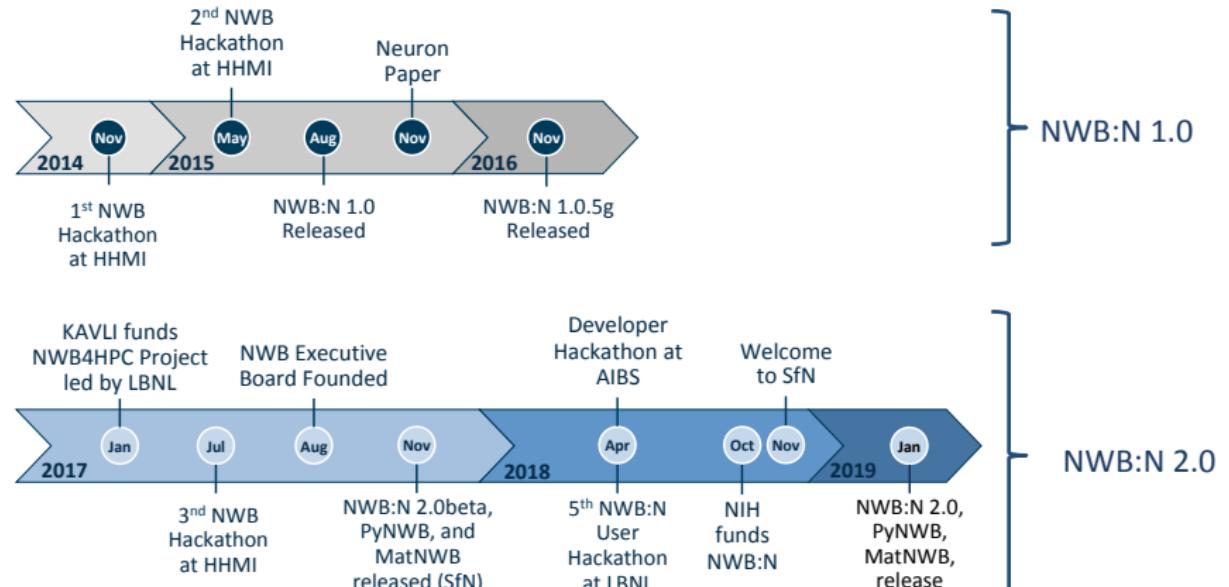
Oliver Rübel

Computational Research Division, Lawrence Berkeley National Laboratory

Open Source Brain Workshop
Alghero, Sardinia
September 10, 2019



A brief history of NWB:N



2019-10-04

Overview

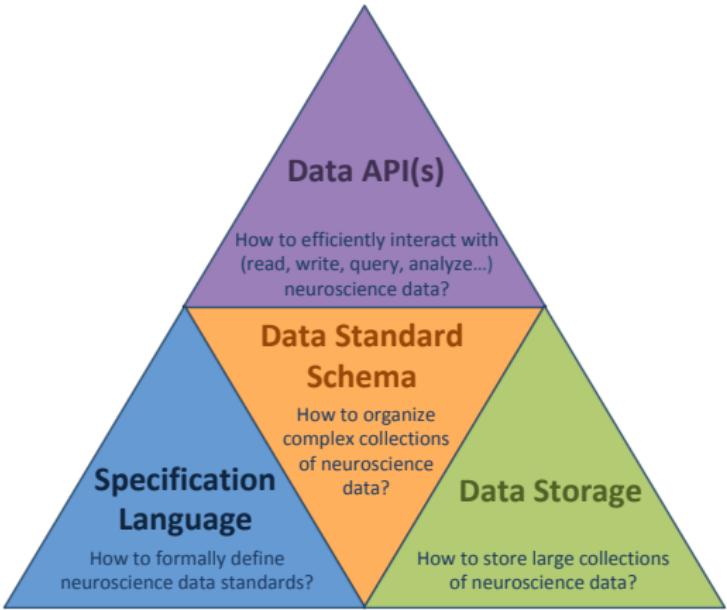
Motivation: Lack of standards for neurophysiology data and related metadata is the single greatest impediment to fully extracting return-on-investment from neurophysiology experiments, impeding interchange and reuse of data and reproduction of derived conclusions

NWB:N – An Ecosystem for Neuroscience Data Standardization

- The NWB:N data standard defines a unified data format for neurophysiology data, focused on the dynamics of groups of neurons measured under a large range of experimental conditions
- NWB:N is more than just a file format but it defines an ecosystem of tools, methods, and standards for storing, sharing, and analyzing complex neurophysiology data

Goal: With NWB:N we aim to develop a next generation data format and software ecosystem that will enable standardization, sharing, and reuse of neurophysiology data and analyses, enhancing discovery and reproducibility

Main components of the NWB:N ecosystem



Specification language

Goal: Enable the formal definition and programmatic interpretation of neuroscience data standards

Format specification language: Schema for defining hierarchical data schemas

Main primitives of the specification language:

- **Object primitives:**

- **Group:** A group is a collection of objects i.e. subgroups, datasets, links
- **Dataset:** n-dimensional array with associated data type, dimensions, etc.
- **Attribute:** small metadata dataset defined on a Group or Dataset
- **Link:** Like a POSIX soft link to given target neurodata_type (specifying a Group or Dataset)

- **Data type specifications**

- **Basic data types:** strings (ascii, utf8), numeric types (float, int, uint, bool etc.) etc.
- **Compound data types:** build complex data type, similar to structs
- **Isodatetime:** ISO8601 datetime string, e.g. 2018-09-28T14:43:54.123+02:00
- **Object reference:** Like a link to given neurodata_type but stored as values of a dataset (or attribute)
- **Region reference:** Links to regions (i.e., select subsets) of datasets stored as as values of a dataset (or attribute)

- **Namespace specification:**

- Used to define a namespace for format specifications
- Needed to define and avoid collisions between extensions and to enable the creation of new data standards



Journal club

└ Standards: the common tongue

2019-10-04

Standards: the common tongue

Data storage

Primary function: Map NWB:N primitives (Groups, Datasets, Attributes etc.) to storage

NWB:N uses HDF5 as its main file storage backend:

- Supports large-scale storage of complex data collections in a single file
- Optimized for performance (parallel I/O, advanced I/O filters etc.)
- Supported across platforms and programming languages (Matlab, Python, C/C++, Fortran, R...)
- Targets long-term support

NWB:N supports advanced I/O features:

- Lazy data load → Fast file open and efficient memory usage even for very large files
- Chunking → Optimize data layout, storage, and I/O
- I/O filter → E.g. use compression to reduce storage cost
- Self-contained data storage → Store all data in a single file (e.g. for sharing)
- Modular data storage → Store data across multiple files and integrated via external links
- Iterative data write → Support data streaming, reduce memory usage, and optimize I/O



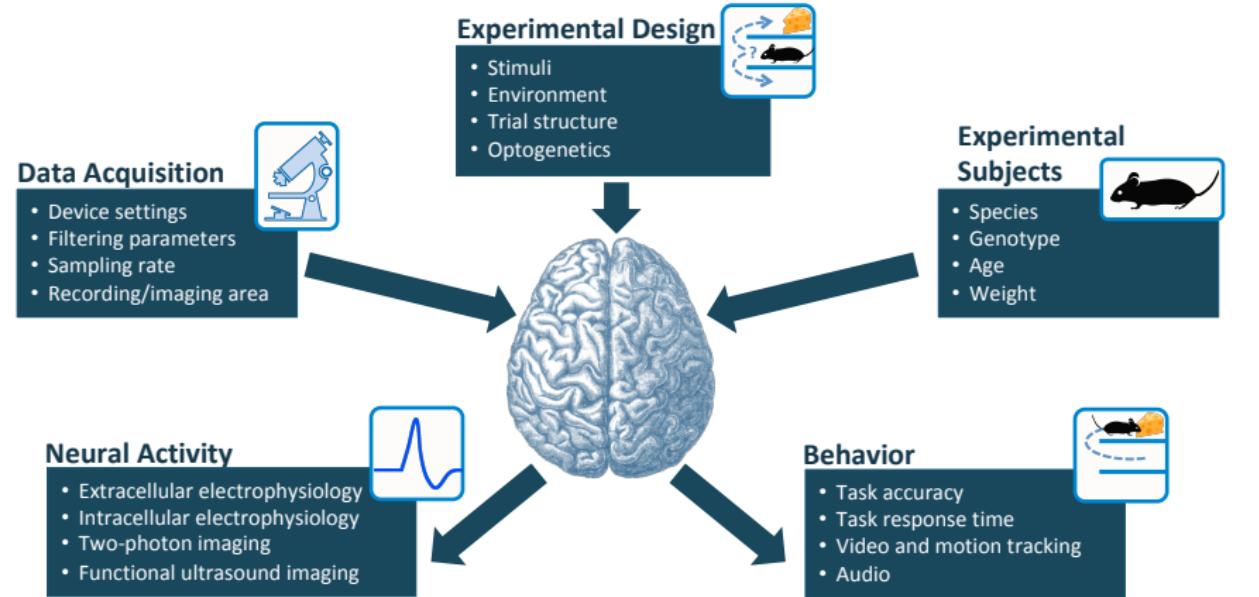
2019-10-04

Journal club

└ Standards: the common tongue

Standards: the common tongue

NWB:N supports complex collections of data required for understanding the brain

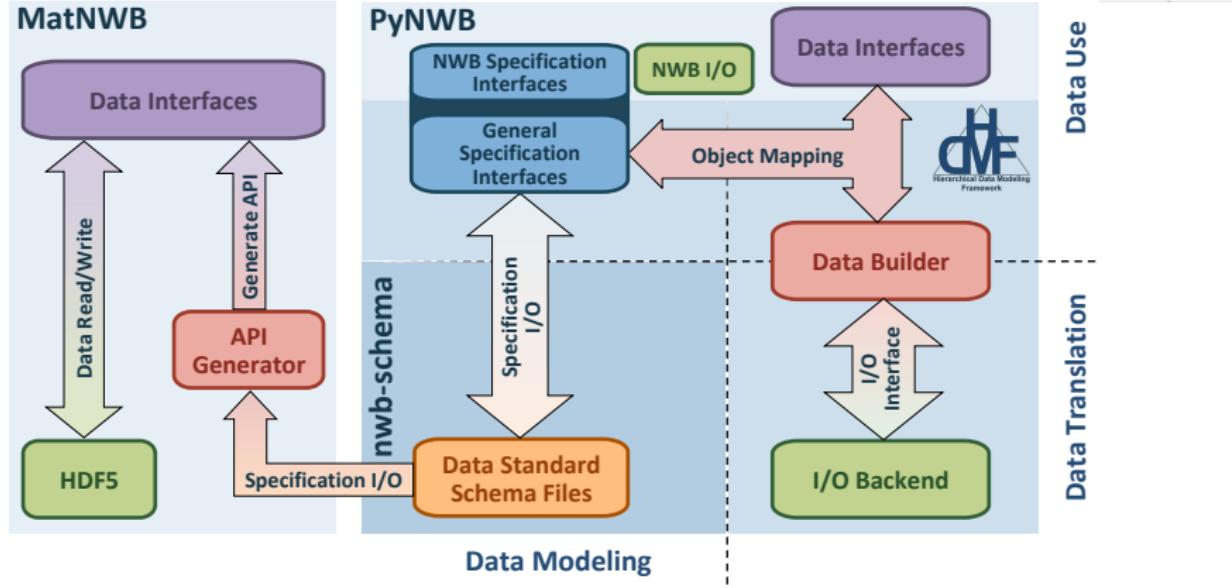


2019-10-04

Journal club
└ Standards: the common tongue

Standards: the common tongue

Advanced software architecture for data standardization

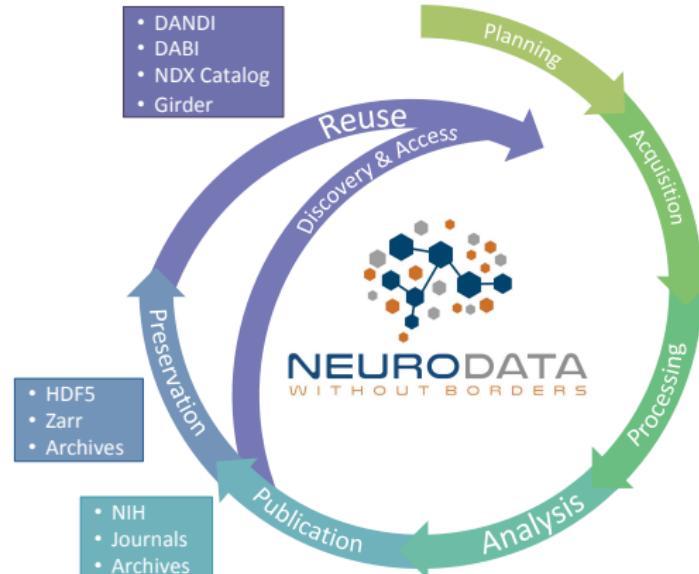


2019-10-04

2019-10-04

Neurophysiology Tools

NWB:N technologies at the heart of the neurodata lifecycle and applications



| Electrophysiology | Optical Physiology |
|--|---|
| DataJoint, NWB Tutorials | |
| OpenEphys, Plexon, Neuralynx, Intan, TDT, SpikeGadgets, SpikeGLX | MiniScope ScanImage |
| PyNWB, MatNWB, HDMF, HDF5 tools/libs | |
| SpikeInterfaces MountainSort, KiloSort, npzsorting BrainStorm WaveClust, UltraMegaSort2000, KiloSort Kluster | NoRMCorre CNMF-E CELLMax EXTRACT |
| NWB-Explorer (OpenSourceBrain) NWB-JupyterWidgets | |
| BrainStorm, RAVE, Neo, SpikeWidgets, EcogVis, ephys-viz, ... | CalmAn |

Community engagement and outreach

Science Engagement

- NWB User/Developer Days:
 - 65+ scientists from 20 major institutions attended in 2018 at LBNL and AIBS
 - 43 users and developers from 29 major labs and research institutions in 2019 at HHMI
- The NIH proposal included 55 letters of support
- Many groups are already adopting NWB:N, e.g., FrankLab, ChangLab (UCSF), BouchardLab (LBNL), SvobodaLab (HHMI), MeisterLab (CalTech), Redwood (UCB), Allen Institute for Brain Science, among others.
- Broad outreach to tool builders:
- Broad outreach to foster publication of data:
- We have reached out to and are working with NIH BRAIN Initiative U19 projects
- KAVLI seed grants
- Simons data pilots

Industry Engagement

- Kitware (Visualization/CI)
- Vidrio (MatNWB)
- MathWorks (MATLAB)
- Vathes (DataJoint)



Governance

- NWB Executive Board
- NWB Technical Advisory Board
- KAVLI foundation
- NIH BRAIN Initiative

Community Resources

Open Source

GitHub

<https://github.com/NeurodataWithoutBorders>

Open Documentation

Read the Docs

<https://neurodatawithoutborders.readthedocs.io>

Developer/User Channels

slack

<https://nwb-users.slack.com>

Google

<https://groups.google.com/forum/#forum/neurodatawithoutborders>

User Engagement

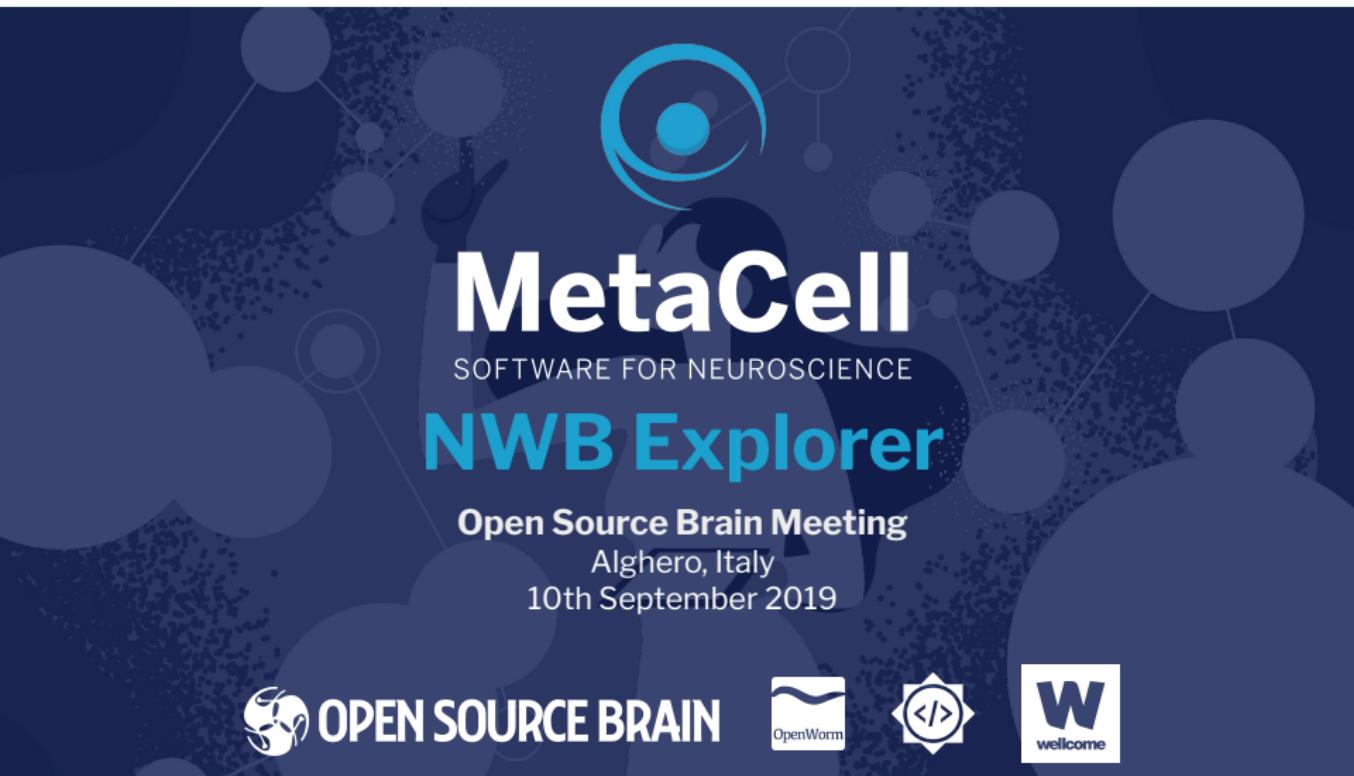
- User- and developer hackathons
- Outreach at neuroscience conferences (CoSyNe, SfN, BRAIN Initiative Investigators Meeting)
- Community discussion and surveys

2019-10-04

Journal club

└ Standards: the common tongue

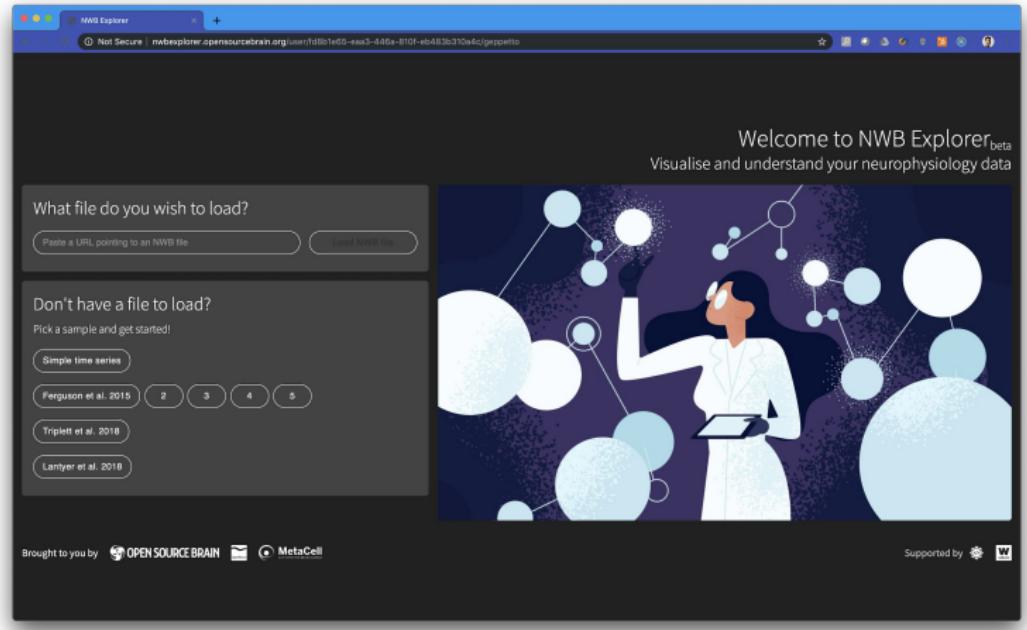
Standards: the common tongue



Journal club
└ Standards: the common tongue

2019-10-04

Standards: the common tongue



Copyright © MetaCell LLC., Ltd. 2011-2019

Journal club
└ Standards: the common tongue

2019-10-04

Standards: the common tongue



NWB EXPLORER

DEMO

Try it out at
<https://nwbexplorer.opensourcebrain.org>

2019-10-04

Journal club
└ Standards: the common tongue

Standards: the common tongue

Creating cortical models across scales in NeuroML

Open Source Brain workshop 2019

11th Sept 2019



Padraig Gleeson
p.gleeson@ucl.ac.uk
University College London

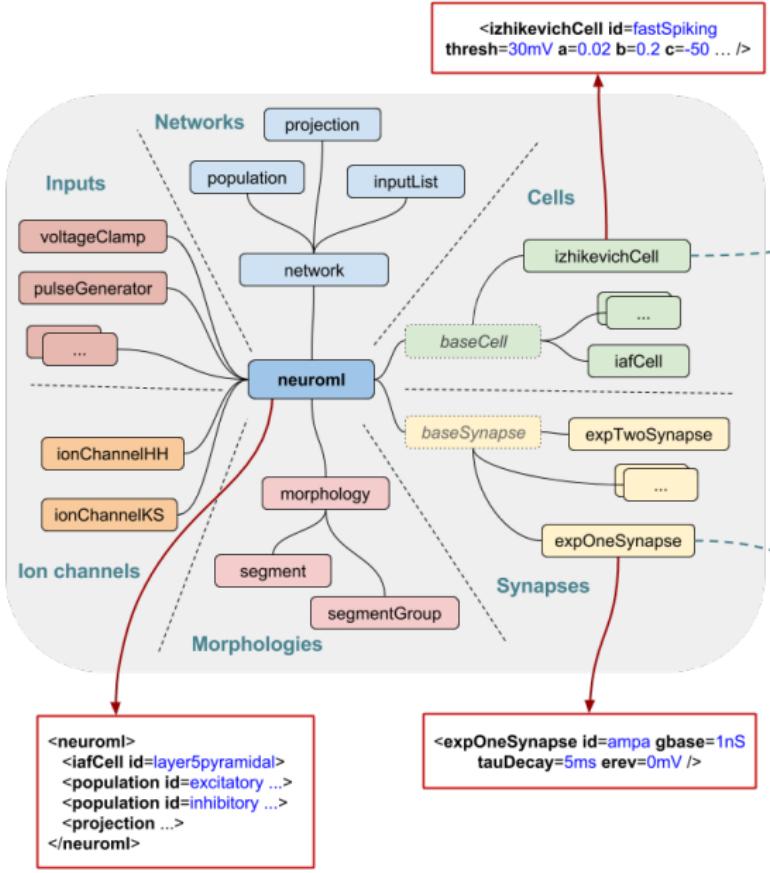
Journal club
└ Standards: the common tongue

2019-10-04

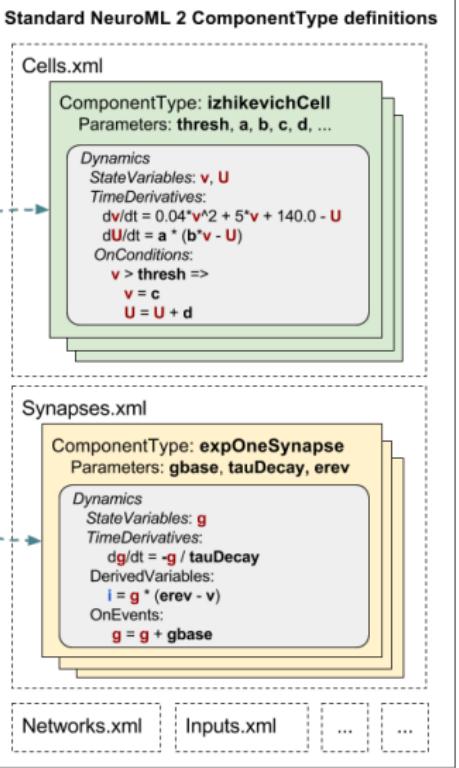
Standards: the common tongue



NeuroML 2



LEMS

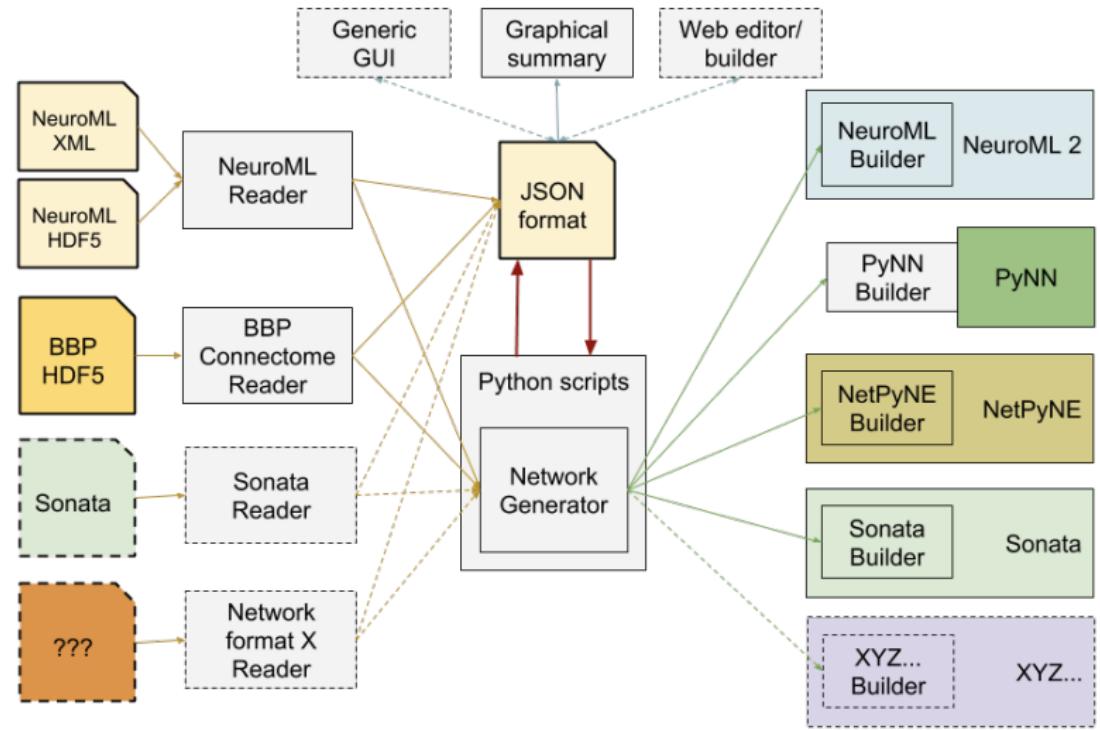


Journal club

Standards: the common tongue

2019-10-04

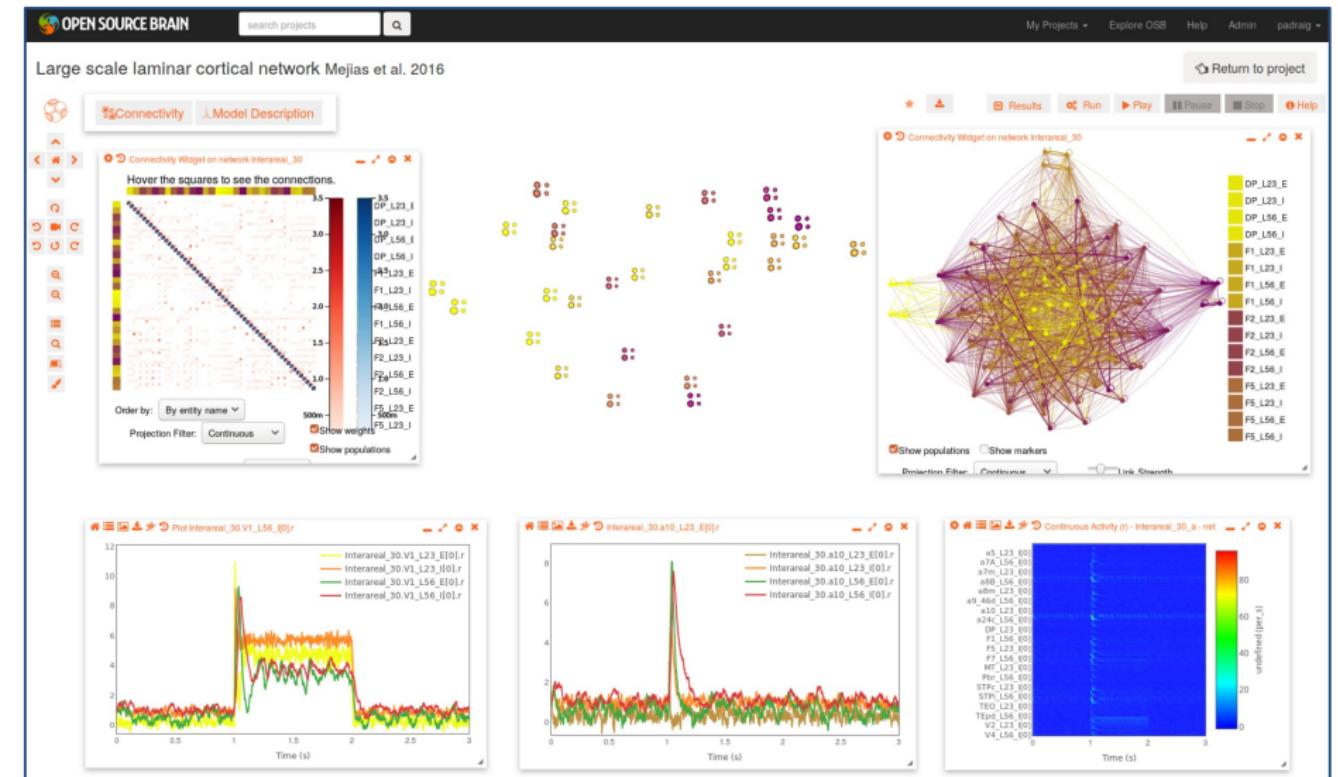
Standards: the common tongue



Journal club
└ Standards: the common tongue

2019-10-04

Standards: the common tongue

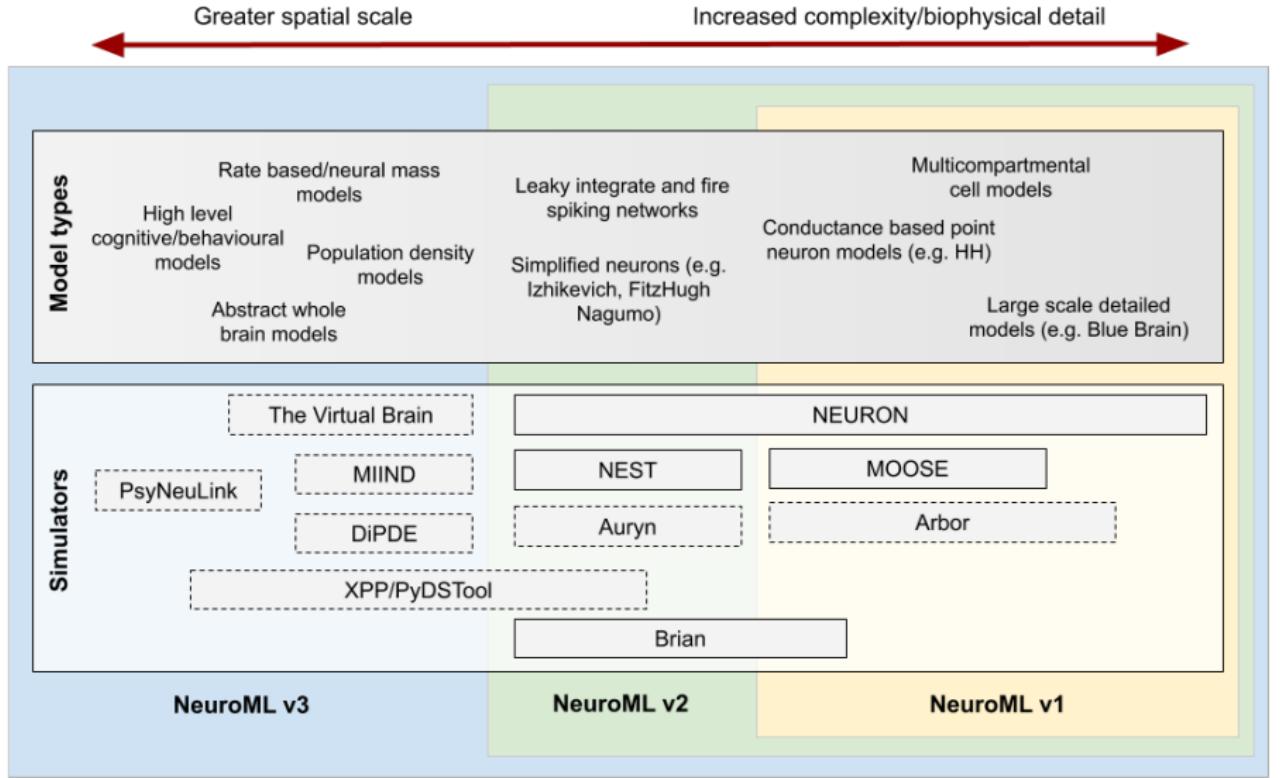


Journal club

Standards: the common tongue

2019-10-04

Standards: the common tongue



Journal club

Standards: the common tongue

2019-10-04

Standards: the common tongue

INCF SIG on Standardised Representations of Network Structures

This SIG deals with the various tools and formats for creating and sharing representations of biological neuronal networks, and will work towards ensuring these are as interoperable and usable as possible for computational neuroscientists.

Contact info: p.gleeson@uc.ac.uk

Members

Anton Arkhipov, Allen Institute, USA

Tom Close, Monash University, Australia

Sharon Crook, Arizona State University, USA

Kael Dai, Allen Institute, USA

Andrew Davison, UNIC, CNRS, France

Lia Domide, Codemart, Romania & Aix-Marseille Université, France

Salvador Durá-Bernal, SUNY Downstate Medical Center, USA

Viktor Jirsa, Aix-Marseille Université, France

Padraig Gleeson, University College London, UK

Sascha von Almada, Würzburg Research Centre, Germany

Converting simulator specific formats to **NeuroML2**

Open Source Brain Meeting 2019



Boris Marin

boris.marin@ufabc.edu.br



Universidade Federal do ABC

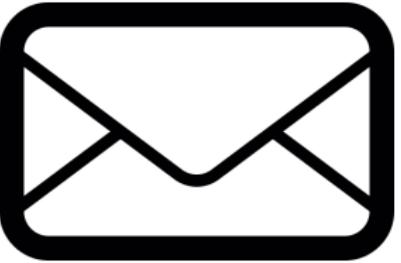
2019-10-04

Journal club

└ Standards: the common tongue

Standards: the common tongue

The Simple™, OSB sponsored way of converting models to
NeuroML2



<mailto:p.gleeson@ucl.ac.uk>

Journal club

└ Standards: the common tongue

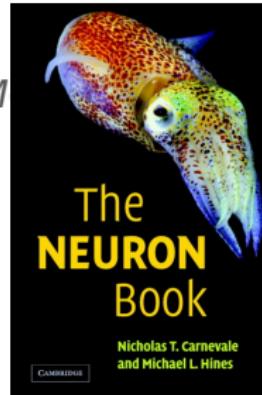
2019-10-04

Standards: the common tongue

Defining models in *NEURON*

<https://www.neuron.yale.edu/neuron/>

- Cells, Networks: *hoc* language (accessible from Python)
 - morphologies
 - synaptic connections
 - *.hoc* files
- Ion Channels (membrane mechanisms): *NM*
 - *.mod* files



2019-10-04

Journal club

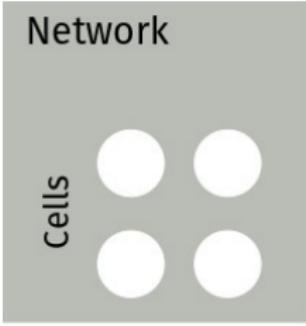
└ Standards: the common tongue

Standards: the common tongue

What is NeuroML, and why should I care?

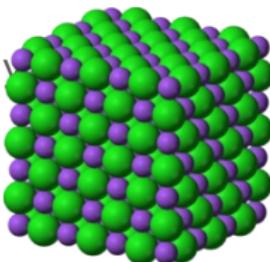
Why can OSB process any NeuroML2 file?

- NML is *structured* (not unlike a *Type System*)



Structure in NeuroML / NMODL

- A *Type System* (composability rules) is what grants NML its superpowers
- nmodl is also powerful, but can be used as a general purpose language
 - VERBATIM blocks
 - many different ways of achieving same goal
 - prone to *unstructuredness*
- OSB could in theory treat nmodl the same way as NML...
 - if only people stuck to "good practices"!



2019-10-04

Journal club

└ Standards: the common tongue

Standards: the common tongue

Levels of Abstraction

NeuroML2

```
<ionChannelHH id="kChan" conductance="10pS" species="k">

    <gateHHrates id="n" instances="4">
        <forwardRate type="HHExpLinearRate" rate="0.1per_ms" midpoint="-55mV" scale="10mV"/>
        <reverseRate type="HHExpRate" rate="0.125per_ms" midpoint="-65mV" scale="-80mV"/>
    </gateHHrates>

</ionChannelHH>
```

NMODL

```
BREAKPOINT {
    SOLVE states METHOD cnexp
    gk = gkbar * n ^ 4
    ik = gk * (v-ek)
}
INITIAL{
    n = alpha(v) / (alpha(v) + beta(v))
}
DERIVATIVE states{
    n' = (1 - n) * alpha(v) - n * beta(v)
}
```

```
FUNCTION alpha(Vm(mV))/(ms){
    LOCAL x
    UNITSOFF
    x = (Vm + 55) / 10
    if(fabs(x) > 1e-6){
        alpha=0.1*x/(1-exp(-x))
    }else{
        alpha=0.1/(1-0.5*x)
    }
    UNITSON
}
```

2019-10-04

Journal club
└ Standards: the common tongue

Standards: the common tongue

Declarative vs Imperative

- NeuroML2 operates (at least syntactically) closer to the level of abstraction employed by electrophysiologists
- The gory details exist, but elsewhere: *LEMS*
 - i.e. what to do with α , β ; the definition of an *ExpRate*; how all of that is converted to conductances/currents...
- But we seldom need (want!) to interact with that level
(look under the hood)

```
<Network ...>
  <Cell ...>
    <Channel ...>
      <Gate ...>
        <Rate ...>
```

```
SOLVE{...} METHOD euler
...
DERIVATIVE {...}
...
FUNCTION trap(v){...}
```

2019-10-04

Journal club

└ Standards: the common tongue

Standards: the common tongue

NetPyNE: structured network specification

i) `popParams['EXC_L2'] = {
 'cellType': 'PYR',
 'cellModel': 'simple',
 'yRange': [100, 400],
 'numCells': 50}`

ii) `popParams['EXC_L5'] = {
 'cellType': 'PYR',
 'cellModel': 'complex',
 'yRange': [700, 1000],
 'density': 80e3}`

iii) `cellParams['PYR_simple'] = {
 'conds': {'cellType': 'PYR',
 'cellModel': 'simple'},
 'secs': {'soma': {
 'geom': {'diam': 18, 'L': 18},
 'mechs': {'hh': {
 'gnabar': 0.12,
 'gkbar': 0.036,
 'gI': 0.003,
 'el': -70}}}},
 'synMechParams': {
 'AMPA': {
 'mod': 'Exp2Syn',
 'taul': 0.8,
 'tau2': 5.3,
 'e': 0}}`

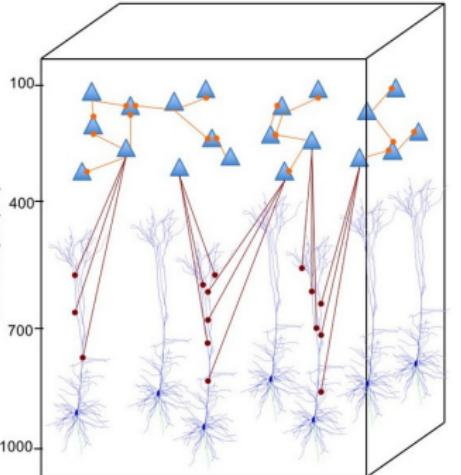
iv) `importCellParams(
label = 'PYR_complex',
conds = {'cellType': 'PYR',
 'cellModel': 'complex'},
fileName = 'L5_pyr_full.hoc',
cellName = 'PYR_L5')`

v) `connParams['L2->E2'] = {
 'preConds': {'y': [100, 400]},
 'postConds': {'pop': 'EXC_L2'},
 'probability': '1*exp(-dist_3D/200)',
 'weight': 0.4,
 'delay': 5,
 'synMech': 'AMPA'}`

vi) `connParams['E2->L5'] = {
 'preConds': {'pop': 'EXC_L2'},
 'postConds': {'y': [700,1100],
 'cellModel': 'complex'},
 'convergence': 25,
 'weight': '0.001 * post_ynorm',
 'delay': 'dist_3D/propVelocity',
 'sec': 'allbend',
 'synMech': 'AMPA',
 'synsPerConn': 3}`

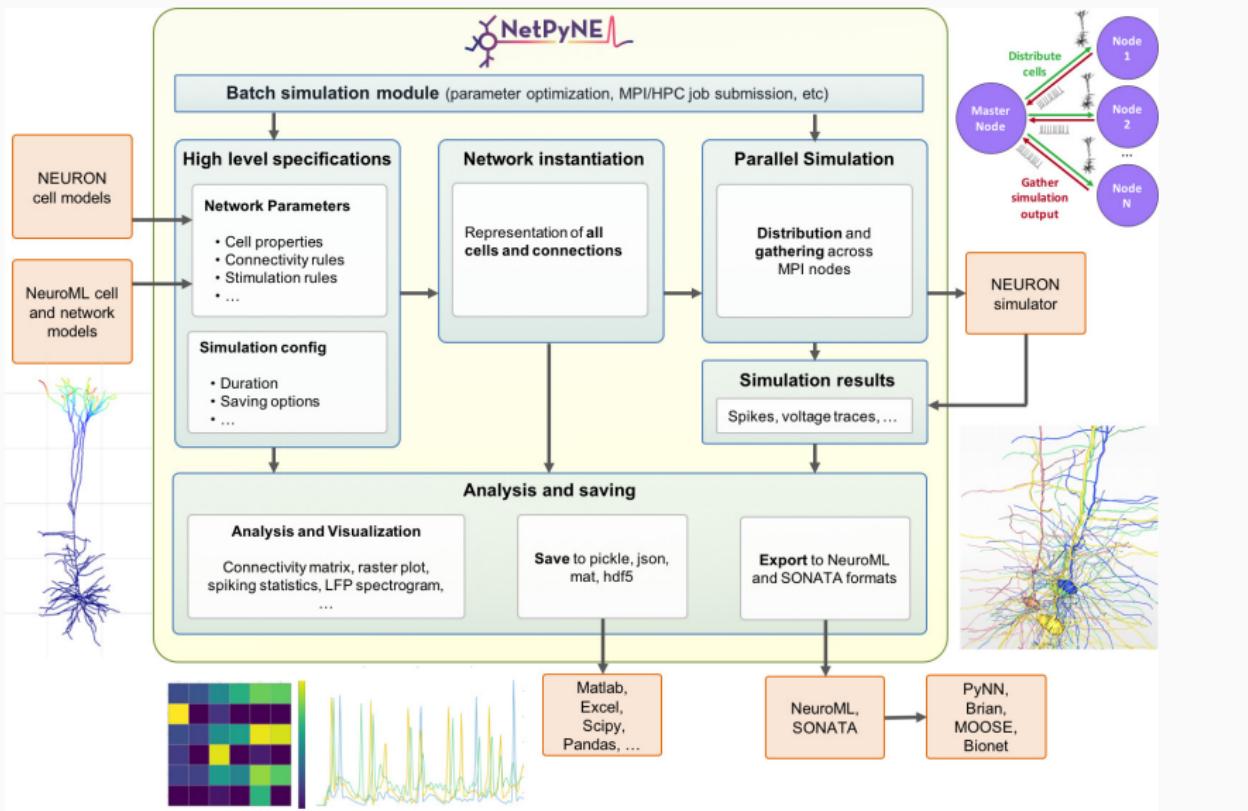
vii) `synMechParams['AMPA'] = {
 'mod': 'Exp2Syn',
 'taul': 0.8,
 'tau2': 5.3,
 'e': 0}`

viii) `synMechParams['GABA'] = {
 'mod': 'Exp2Syn',
 'taul': 0.8,
 'tau2': 5.3,
 'e': -70}`



Dura-Bernal, Salvador, et al. "NetPyNE, a tool for data-driven multiscale modeling of brain circuits." Elife 8

Netpyne workflow



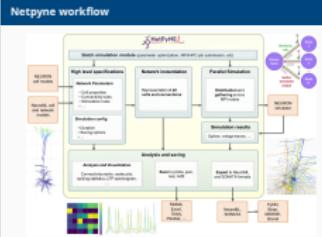
4/8

Journal club

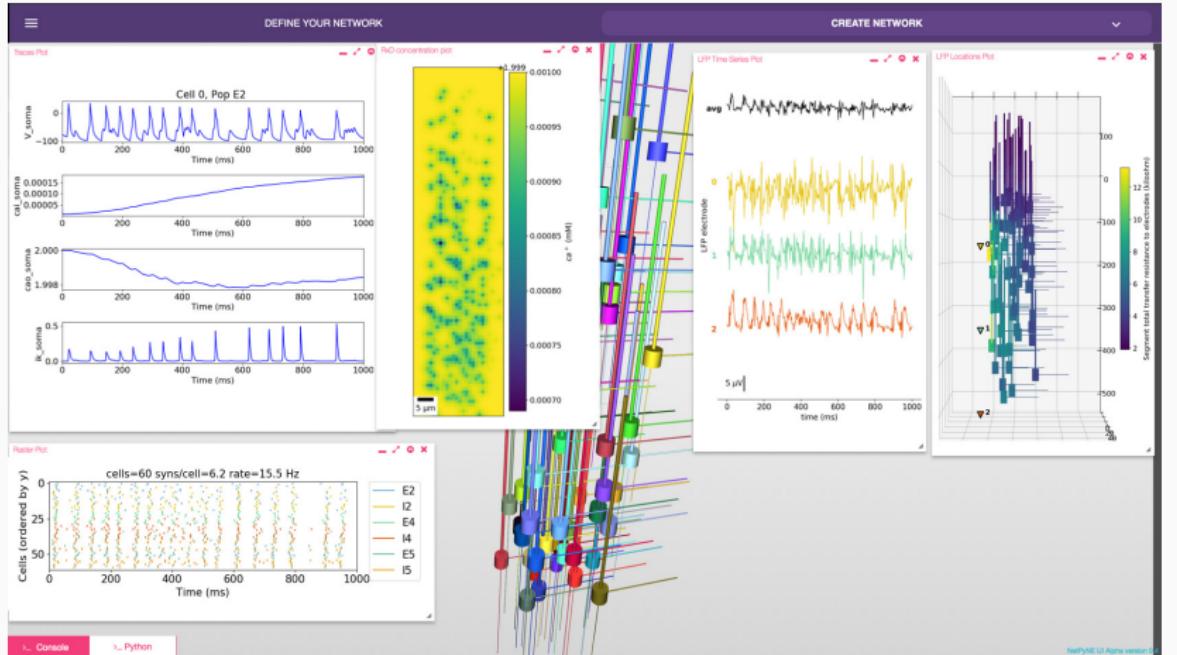
Standards: the common tongue

Netpyne workflow

2019-10-04



Netpyne GUI



Journal club

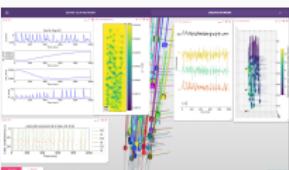
└ Standards: the common tongue

└ Netpyne GUI

2019-10-04

5/8

Netpyne GUI

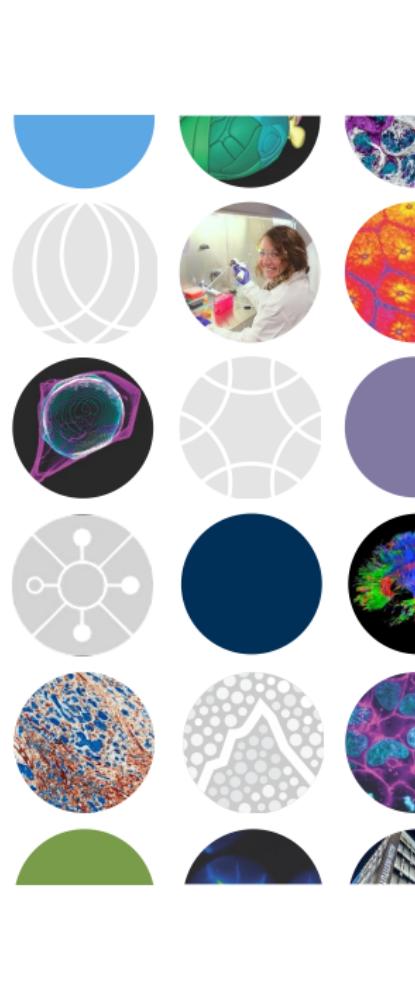




Large-scale Datasets and Modeling Tools from the Allen Institute for Brain Science

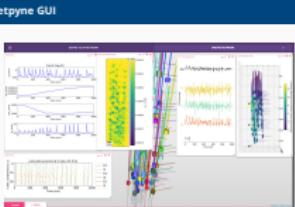
Yazan N. Billeh

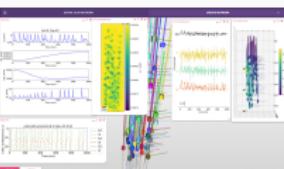
yazanb@alleninstitute.org



2019-10-04

Journal club
└ Standards: the common tongue
 └ Netpyne GUI





Journal club

└ Standards: the common tongue

2019-10-04

└ Netpyne GUI

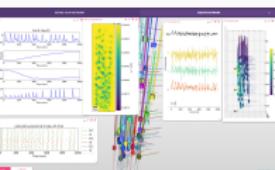
ALLEN INSTITUTE for
BRAIN SCIENCEALLEN INSTITUTE for
CELL SCIENCEALLEN INSTITUTE for
IMMUNOLOGYTHE PAUL G. ALLEN
FRONTIERS GROUP

hard problems
complexity
foundational biology

big science
team science

open science





CORE PRINCIPLES

Team Science

Interdisciplinary teams working towards common goal



Big Science

Large-scale projects with robust, massive data



Open Science

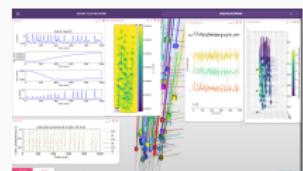
All resources available online at brain-map.org or allencell.org

Journal club

Standards: the common tongue

Netpyne GUI

2019-10-04



Journal club

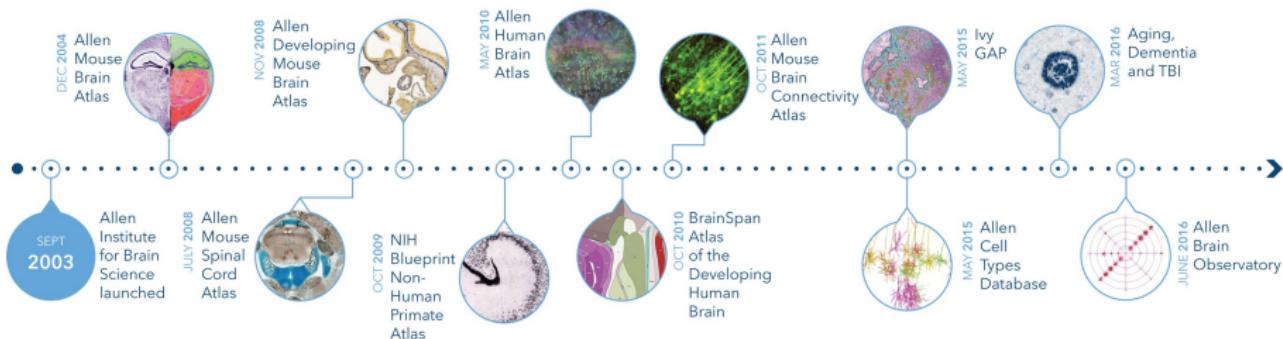
Standards: the common tongue

2019-10-04

Netpyne GUI

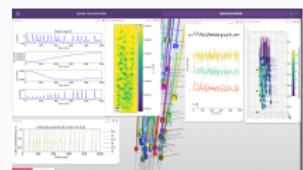
Allen Institute - Online Public Resources

www.brain-map.org



All data are:

- publicly accessible via API as soon as they pass QC
- freely available without any commercial restrictions



Journal club

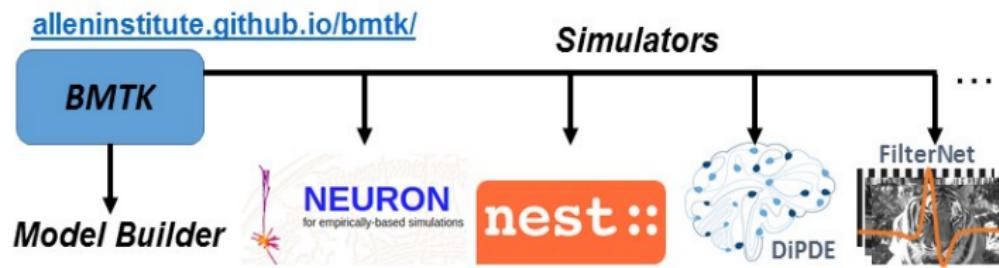
└ Standards: the common tongue

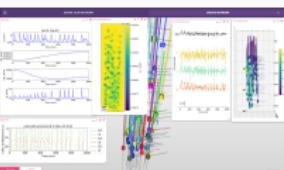
└ Netpyne GUI

2019-10-04

Our Models and Modeling Software Are Freely Available to the Community

Brain Modeling ToolKit (BMTK): <https://alleninstitute.github.io/bmtk/>





Journal club

└ Standards: the common tongue

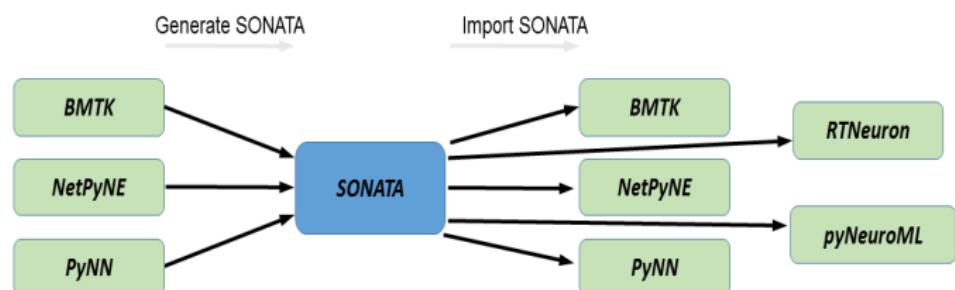
└ Netpyne GUI

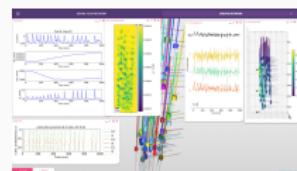
2019-10-04

Our Models and Modeling Software Are Freely Available to the Community

Scalable Open Network Architecture TemplAte (SONATA): <https://github.com/AllenInstitute/sonata>

An interface between SONATA and the NWB format has been developed as well





Journal club

└ Standards: the common tongue

└ Netpyne GUI

2019-10-04



Human Brain Project

How model standardization enables new tools and applications in neuroscientific research

Insights from the HBP

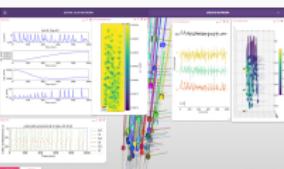
Yann Zerlaut

Neuroinformatics team / group of A. Davison
Centre National de la Recherche Scientifique, France



Open Source Brain Meeting 2019, Alghero





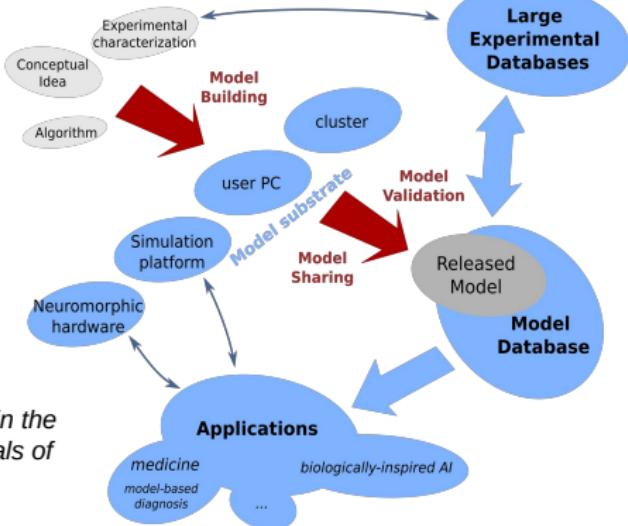
Journal club

└ Standards: the common tongue

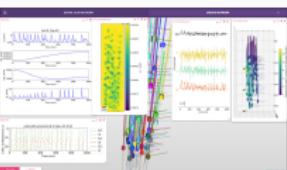
└ Netpyne GUI

2019-10-04

Motivation



Model production pipeline within the infrastructure and research goals of the Human Brain Project



Journal club

Standards: the common tongue

Netpyne GUI

2019-10-04



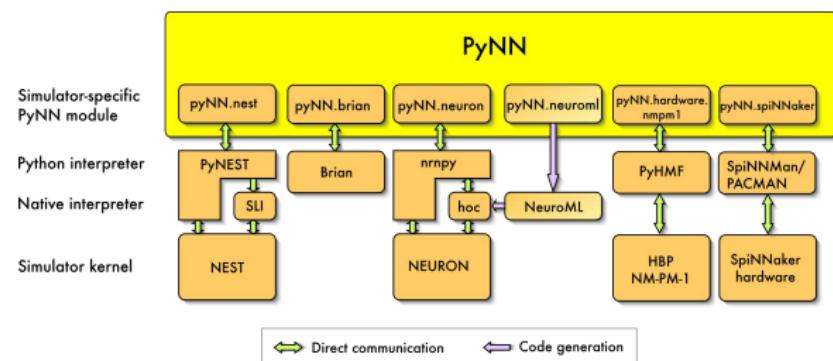
A unified interface for neuronal network simulators

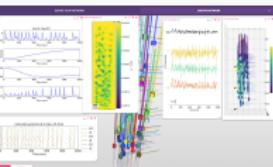
Simulator-independent environments for developing neuroscience models:

- keep the advantages of having multiple simulators or hardware devices
- but remove the translation barrier.

Three (complementary) approaches:

- GUI (e.g. neuroConstruct)
- XML-based language (e.g. NeuroML, NineML)
- interpreted language (e.g. Python)





Journal club

Standards: the common tongue

Netpyne GUI

2019-10-04

Sonata (pyNN support)

Large-scale simulation of biophysically-detailed neuronal circuits
→ sets specific constraints

the SONATA Data Format emerges as the standard
optimized for performance for simulation, analysis and
visualization of large-scale circuits
(joint initiative of Blue Brain Project and the
Allen Institute for Brain Science)

Export to Sonata format

```
from pyNN.network import Network
from pyNN.serialization import export_to_sonata

sim.setup()
...
# create populations, projections, etc.
...

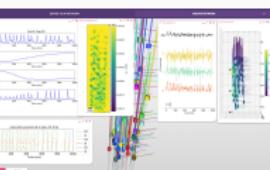
# add populations and projections to a Network
net = Network(pop1, pop2, ..., prj1, prj2, ...)

export_to_sonata(net, "sonata_output_dir")
```

Import from Sonata format

```
from pyNN.serialization import import_from_sonata, load_sonata_simulation_plan
import pyNN.neuron as sim

simulation_plan = load_sonata_simulation_plan("simulation_config.json")
simulation_plan.setup(sim)
net = import_from_sonata("circuit_config.json", sim)
simulation_plan.execute(net)
```



Journal club

Standards: the common tongue

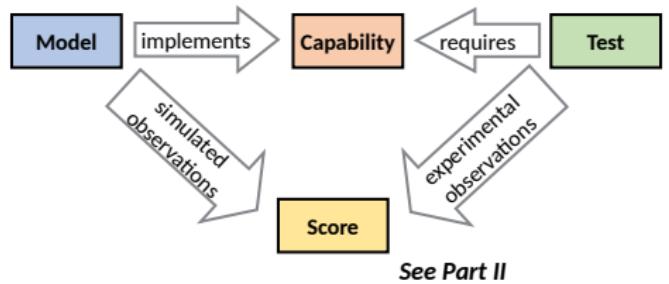
Netpyne GUI

2019-10-04

SciUnit

<https://github.com/scidash/sciunit>

Include a validation framework in model development



✓ What is SciUnit?

A **Test-driven framework** for formally validating scientific models against data.

It employs the concept of **Capabilities**.

✓ What are Tests?

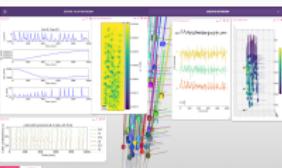
A procedure intended to establish the quality, performance, or reliability of a model

Requires the participation of modellers:

- ✓ support to wrap your models for SciUnit
- ✓ add/request new tests to the library
- ✓ critique existing tests
- ✓ suggest new features

✓ What are Capabilities?

- interfaces through which the model and the validation framework communicate
- implemented as methods (functions) within the model



Journal club

Standards: the common tongue

Netpyne GUI

2019-10-04

Test Packages

The overall test suite has been divided into a number of components, some containing validation tests specific to particular brain regions, others more generic. All validation tests are written in Python, using the SciUnit framework. Some of these are listed below:

Test suites for specific brain regions

- ❑ **HippoUnit:** <https://github.com/KaliLab/hippounit>
- ❑ **HippoNetworkUnit:** <https://github.com/pedroernesto/HippoNetworkUnit>
- ❑ **CerebUnit:** <https://github.com/lungsi/cerebellum-unit>
- ❑ **BasalUnit:** <https://github.com/appukuttan-shailesh/basalunit>

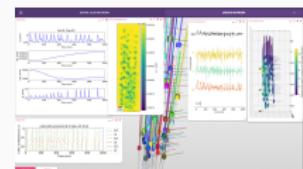
Test suites for model features, independent of cell type or brain region

- ❑ **MorphoUnit:** <https://github.com/appukuttan-shailesh/morphounit>
- ❑ **NetworkUnit:** https://github.com/mvonpapen/simrest_validation
- ❑ **eFELUnit:** <https://github.com/appukuttan-shailesh/eFELunit>

Journal club

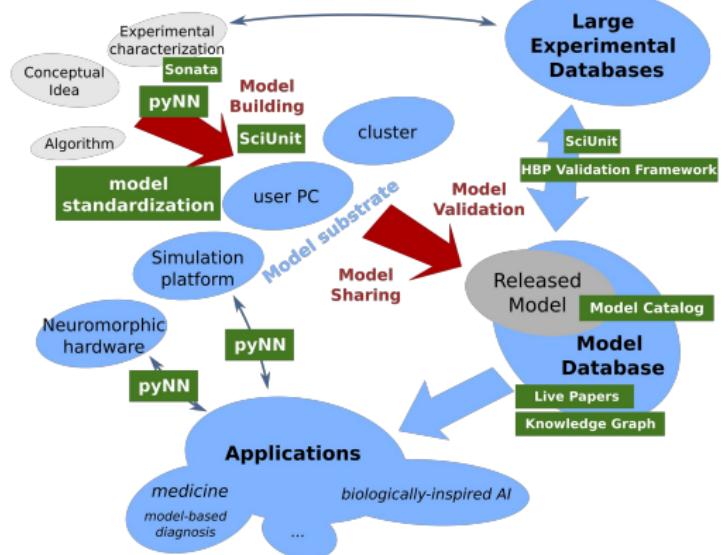
└ Standards: the common tongue

└ Netpyne GUI



2019-10-04

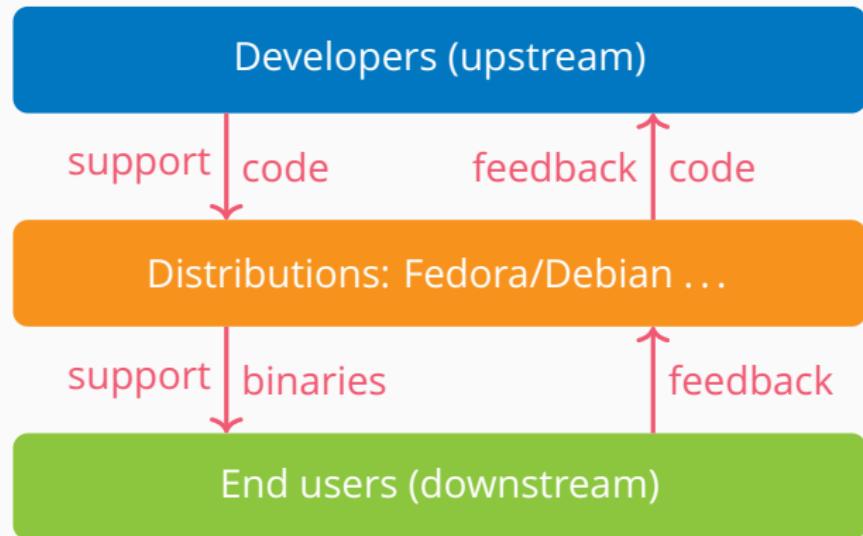
Summary



2019-10-04

NeuroFedora: marketing pitch

Liaison between developers and users



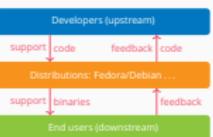
Journal club

└ NeuroFedora: marketing pitch

└ Liaison between developers and users

2019-10-04

Liaison between developers and users



Search: "NeuroFedora"



"Live" ISO now ready to download (demo)

Mailing list: neuro-sig@lists.fedoraproject.org

IRC: #fedora-neuro on Freenode

Telegram: t.me/NeuroFedora

Documentation neuro.fedoraproject.org

Blog: neuroblog.fedoraproject.org

Pagure.io (FOSS Git forge): [neuro-sig/NeuroFedora](#)

Journal club

└ NeuroFedora: marketing pitch

└ Search: "NeuroFedora"

2019-10-04



"Live" ISO now ready to download (demo)
Mailing list: neuro-sig@lists.fedoraproject.org
IRC: #fedora-neuro on Freenode
Telegram: t.me/NeuroFedora
Documentation neuro.fedoraproject.org
Blog: neuroblog.fedoraproject.org
[Pagure.io](#) (FOSS Git forge): [neuro-sig/NeuroFedora](#)



This presentation is made available under a
Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) license.