EBRAINS modelling workflows implementation status at M42 (D5.8 - SGA3)

M42 status on EBRAINS implementation and procedures, including open-source workflows, simulation engines for multilevel brain modelling, and outlook for post-HBP phase.

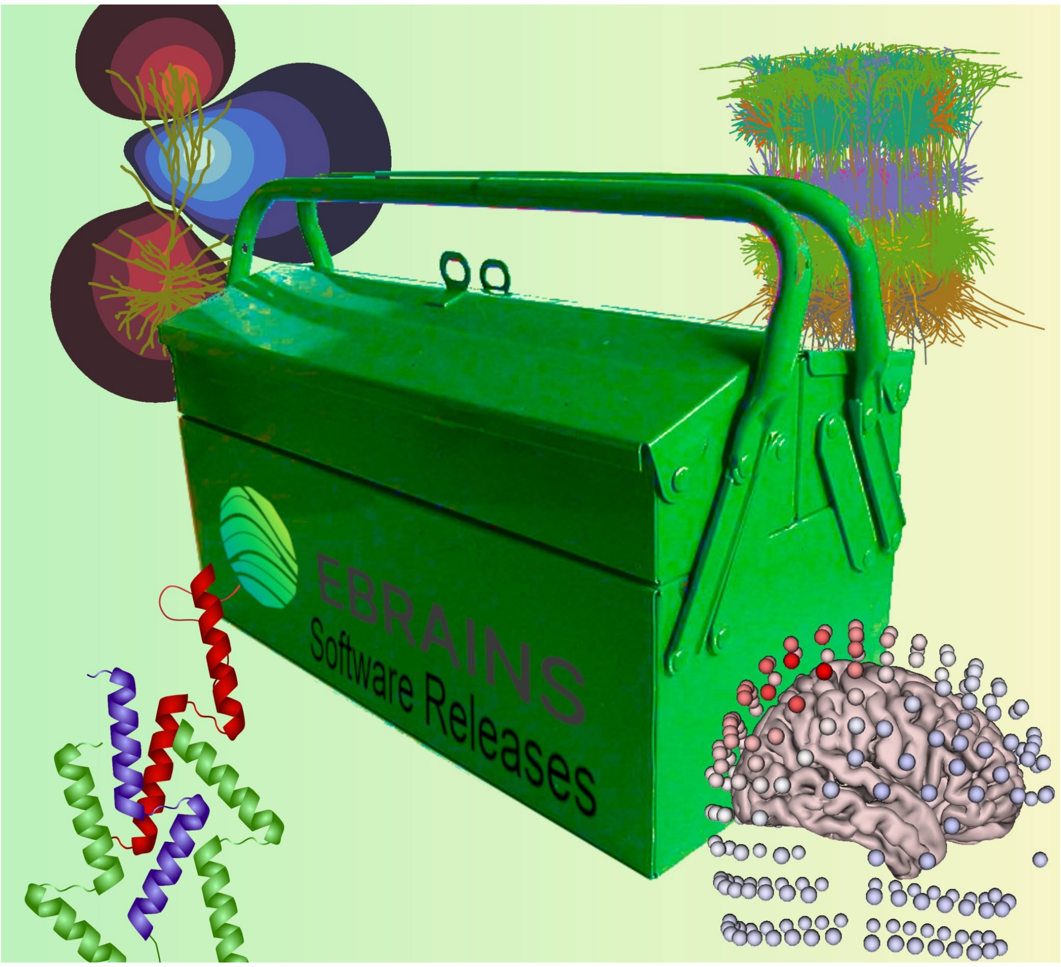


Figure 1: EBRAINS software releases bundle simulation tools at different abstraction levels

EBRAINS software releases provide scientists with an easy way to access the modelling and simulation tools, spanning the molecular to whole-brain levels, in a common, consistent software environment, facilitating the development of complex simulations.[1]

---

[1] Image sources: Næss, S., et al., Biophysically detailed forward modeling of the neural origin of EEG and MEG signals. NeuroImage, 2021. 225: p. 117467 (P2604); http://www.logodesignweb.com/stockphoto, JM.

Project Number:	945539	Project Title:	HBP SGA3

Document Title:	EBRAINS modelling workflows implementation status at M42
Document Filename:	D5.8 (D55) SUBMITTED 230913.docx
Deliverable Number:	SGA3 D5.8 (D55)
Deliverable Type:	Other
Dissemination Level:	PU = Public
Planned Delivery Date:	SGA3 M42 / 31 SEP 2023
Actual Delivery Date:	SGA3 M42 / 13 SEP 2023
Author(s):	Michael DENKER, Dennis TERHORST, Susanne KUNKEL, JUELICH (P20)
Compiled by:	Anne ELFGEN, JUELICH (P20)
Contributor(s):	Jeanette HELLGREN KOTALESKI, KTH (P39) Rebecca WADE, HITS (P26) Michele MIGLIORE, CNR (P12) Stefano ANTONEL, EPFL (P1) Jean-Denis COURCOL, EPFL (P1) Michael GEVAERT, EPFL (P1) Daniel KELLER, EPFL (P1) James KING, EPFL (P1) Werner VAN GEIT, EPFL (P1) Katrien VAN LOOK, EPFL (P1) Robin Gilbert DE SCHEPPER, UNIPV (P70) Markus DIESMANN, JUELICH (P20) Abigail MORRISON, JUELICH (P20) Charl LINSSEN, JUELICH (P20) Sandra DIAZ, JUELICH (P20) Huifang WANG, AMU (P78) Wouter KLIJN, JUELICH (P20) Lena ODEN, JUELICH (P20) Sonja GRÜN, JUELICH (P20) Benjamin WEYERS, UT (P130) Óscar David ROBLES SÁNCHEZ, UPM (P68) Giulia ROSSETTI, JUELICH (P20) Espen HAGEN, NMBU (P44) Torbjørn Vefferstat NESS, NMBU (P44) Brent HUISMAN, JUELICH (P20) Christian TETZLAFF, UGOE (P140) Mario NEGRELLO, EMC (P141)
WP QC Review:	Anne ELFGEN, JUELICH (P20)
WP Leader / Deputy Leader Sign Off:	Susanne KUNKEL, JUELICH (P20)
T7.4 QC Review:	Only formatting by Annemieke MICHELS, EBRAINS (P1)
Description in GA:	M42 status on EBRAINS implementation and procedures, including open-source workflows, simulation engines for multilevel brain modelling, and outlook for post-HBP phase.
Abstract:	This report presents the status of tools and workflows for modelling and simulation in EBRAINS, as of September 2023, the end of the HBP. It contains a brief guide to each of the available and planned tools/workflows with links to further information, recent changes since the M21 Deliverable and an outlook for the post-HBP phase.

Keywords:	Modelling, simulation, data analysis, model validation, visualisation, molecular, subcellular, single neuron, network, whole brain, software libraries, web apps, workflows.
Target Users/Readers:	Computational neuroscience community, computer scientists, Consortium members, funders, general public, HPC community, neuroinformatics specialists, neuroscience community, neuroscientists, platform users, researchers, students.

Table of Contents

Table of Tables

Table of Figures

1. Overview

Over the last decade, the Human Brain Project, one of the flagship projects of the EU, has set out to develop EBRAINS as a digital research infrastructure aimed at providing experimental and computational neuroscientists the tools and resources necessary to create cutting-edge approaches to scientific discovery in fundamental neuroscience research and drive innovation in medical brain research and industry. Challenges arise in a diverse set of scenarios, including integration efforts to bridge the diversity of scales inherent to brain research, in handling large and complex datasets, modern large-scale simulations of brain dynamics in a multi-disciplinary world, and a variety of applications ranging from robotics to progress in treating brain diseases.

The vision of EBRAINS is to provide the scientific community with an open infrastructure that is more than the sum of its parts: a shared platform hosting a rich diversity of tools, enabling interoperability, rigour and compatibility of its components in order to facilitate cross-disciplinary exchange and collaborative work on complex research questions. In this way, EBRAINS acts as an ecosystem of tools, services and resources to leverage, transform and extend knowledge for next generation research. EBRAINS is a testament to Europe's ambition to facilitate and shape the digital transformation in the neurosciences.

In this document, we present the final implementation status at the end of the project (M42) of all software and service components that make up the EBRAINS modelling workflows. The structure of the document closely follows previous updates given in Deliverables D5.2[2] (M9, December 2020) and D5.5[3] (M21, December 2021) and provides recent updates and improvements on a component-by-component basis. The tool and service integration efforts of EBRAINS are highly visible in the recent efforts to bundle many EBRAINS offerings into an integrated set, in which all components are tested for compatibility and interoperability (see D5.7). The descriptions of this document provide information on the integration type and status of tools in terms of the specific release of the global EBRAINS Software Stack hosting a specific software, and whether a specific service is integrated into the EBRAINS cloud resources and available under the ebrains.eu domain.

2. Status Summary of Modelling Workflows in EBRAINS

For all components of the simulation and modelling workflows, efforts in SGA3 have focused on meeting the tool-sided requirements for integration and smooth operation with the central components of the EBRAINS infrastructure. Ease of use (e.g., via documentation) has been further improved, and harmonisation, modularity and interoperability of the components has seen a significant boost. In this M42 deliverable, we detail these efforts, both user facing and hidden, as they build upon the accomplishments already achieved at M21. In addition, we summarize for each component its key improvements in SGA3.

EBRAINS at M42 (September 2023) sees a massively improved process for software delivery that provides mechanisms for ensuring compatibility, interoperability and rigour. By introducing the concept of regular EBRAINS Software Releases consisting of the latest assortment of tested and interoperable EBRAINS tools and workflows, packaged in a single consistent environment, users are now presented with a hassle-free setup process for their software in new projects hosted on the EBRAINS infrastructure (see D5.7). Efforts were made to integrate, where possible, simulation and modelling software offers, such that those tools are readily available on all EBRAINS and FENIX resources. To this end tools and services have been continuously updated to interoperate with the EBRAINS infrastructure. In part, this involves the use of the central EBRAINS identity and access

---

[2] https://sos-ch-dk-2.exo.io/public-website-production/filer_public/bc/40/bc400294-0104-4ec3-b463-72803255b686/d52_d49_sga3_m9_accepted_210303.pdf

[3] https://sos-ch-dk-2.exo.io/public-website-production-2022/filer_public/ec/82/ec8251bf-0ec8-4c9f-9d53-bb992aa56fde/d55_d52_sga3_m21_accepted_220520.pdf

management (IAM) system to provide a unified login using HBP credentials but also moving services to be hosted on EBRAINS infrastructure and be reachable under the ebrains.eu domain.

These harmonisation efforts, taken together with the ability to handle data access and deposition by the EBRAINS Knowledge Graph, access to HPC systems via the UNICORE system, and the availability of Collaboratory features including JupyterHub, Drive and Bucket, ensure that users are confronted with a consistent and uniform access to modelling workflows that is fundamentally different from previous implementations of the infrastructure.

Facilitating this integration work behind the scenes, there were multiple new software releases in the past 21 months, and defined procedures for continuously including these into the evolving EBRAINS Software ecosystem have been developed. These procedures include checks for software quality (SQ) and its development process (SQ-Guidelines, D5.7), ensuring the interoperability of tools and services via automated test procedures (see D5.7), and the definition of security requirements (D5.14). Over the different EBRAINS Software Releases (21.09, 22.07, 22.10, 23.02, 23.06, and 23.09) this has led to a robust, centrally defined workflow that is decoupled from timelines of single tools or deployment sites, respectively leaving responsibility and control in the same hands (decentrally).

Many services were extended with essential new features and/or have been integrated into the deployment on the EBRAINS OpenShift cloud infrastructure. The interconnection and interoperability of these services presents a unique multi-tool platform to users, allowing to switch easily to the tool best suited for the task at hand. Table 1 shows an overview of the current status at M42 of the interdependencies between EBRAINS tools and services. A simple example is to move with a single click directly from documentation into the live Jupyter Notebook (aka “Try it on EBRAINS”) to try a feature (such as implemented here[4]), giving users a direct way to explore EBRAINS tools and workflows. Other highlights of new features include the extended list of tools accepting files in SONATA format as input or producing these as output.

There is a new workflow listed in this Deliverable, namely BioExcel Building Blocks Workflows, which was not listed in the earlier M9 and M21 Deliverables (D5.2 and D5.5). The ArDOCK web server is no longer online and has been removed from the HBP/EBRAINS list of offered tools/software.

The large majority of tools and services is presented through the single entry point on the ebrains.eu website[5] providing a comprehensive overview outlining the users’ possibilities within EBRAINS.

Since M21, further HBP Facility Hubs[6] for several communities, i.e. the Hippocampus, Cerebellar Modelling, Basal Ganglia, NEST and Virtual Brain Facility Hubs, have been established to serve the EBRAINS research infrastructure with additional services that are aligned with the Fenix supercomputing network to provide user access to computing resources, e.g. for data storage, analysis, and simulation.

3. Outlook Summary

EBRAINS tools and workflows have become an active and living ecosystem with many component owners committed to further sustain the technological heritage of the HBP. Most projects have concrete plans for continued maintenance and advancement of their tools, half of them plan to invest in interfaces or data-format compatibility to further promote interoperability with other tools of the ecosystem. As in every evolving system, tools are in different places in their software lifecycle. Four tools will drop to a maintenance-only status, and two will not be continued after the end of the project. However, about half of the projects plan to establish new tools or workflows for EBRAINS or provide new data sets catering for the needs of ever more complex scientific endeavours.

A positive aspect of the interaction of tool developers in the HBP is the common awareness that solid support structures, such as better research software engineering (RSE) practices and automated integration workflows, are key for sustainable research, where the success of more mature projects

---

[4] https://nest-simulator.readthedocs.io/en/stable/examples/index.html#pynest-examples

[5] https://www.ebrains.eu/modelling-simulation-and-computing/simulation/simulation-tools

[6] https://www.humanbrainproject.eu/en/collaborate-hbp/facility-hubs/

has served as inspiration. Some projects have also started to develop administrative support structures, such as facility hubs or foundations, or they now benefit from an active user community.

4. How to Read this Document

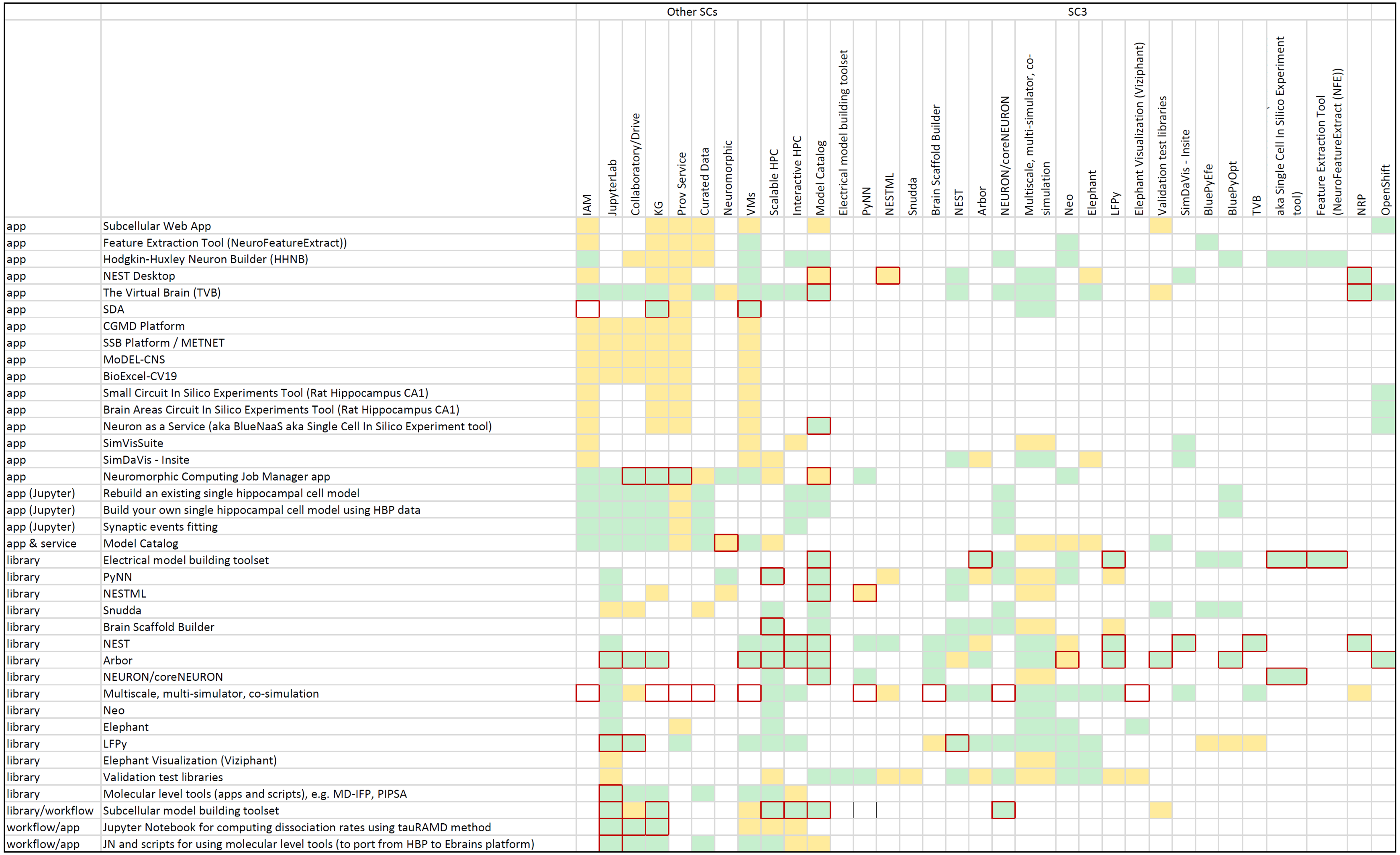
EBRAINS Modelling Workflows can be categorised by type, availability/maturity, scope, modelling and abstraction level. Table 2 explains the icons used in Section 5 to document the categorisation of each tool or workflow and available user engagement channels. Where tools and services have multiple features with different levels of maturity, the icon indicates the most mature features. These icons have also been used in both previous Deliverables D5.2 and D5.5, while the categorisation of the integration states has been revised, and a new icon has been introduced for "integrated in EBRAINS software stack".

General tool and service descriptions have been provided in D5.2 and are thus not being repeated in this Deliverable. Below we give brief updates to the available tools and services, grouping them by activity.

Table 1: Legend for icons for each tool or service that were already used in D5.2 and D5.5.

Type		Availability/ maturity		Scope		Modelling abstraction level		User engagement channels	
web app	💻	mature, available now (TRL 7-9)	✔	molecular	⁙	nanoscale	NS	online training/course/MOOC	☁
command-line tool or library	⚙	prototype available (TRL 5-6)	🔧	subcellular	⇄	systems biology	SB	workshop/summer school (in person)	👥
		in progress / future work	📋	single neurons	⚲	point neuron models	PT	electronic support	🎧
				networks	⅄	biophysical models	BI		
				whole brain	🧠	population models	PP		
						statistical models	ST		

Table 2: Interdependencies between EBRAINS tools and services.



		Other SCs										SC3																					
		IAM	JupyterLab	Collaboratory/Drive	KG	Prov Service	Curated Data	Neuromorphic	VMs	Scalable HPC	Interactive HPC	Model Catalog	Electrical model building toolset	PyNN	NESTML	Snudda	Brain Scaffold Builder	NEST	Arbor	NEURON/coreNEURON	Multiscale, multi-simulator, co-simulation	Neo	Elephant	LFPy	Elephant Visualization (Viziphant)	Validation test libraries	SimDaVis - Insite	BluePyEfe	BluePyOpt	TVB	aka Single Cell In Silico Experiment tool)	Feature Extraction Tool (NeuroFeatureExtract (NFE))	NRP	OpenShift
app	Subcellular Web App	🟨			🟨	🟨	🟨		🟨			🟨														🟨								🟩
app	Feature Extraction Tool (NeuroFeatureExtract))	🟨			🟨	🟨	🟨		🟩													🟩						🟩						
app	Hodgkin-Huxley Neuron Builder (HHNB)	🟩		🟨	🟨	🟨	🟨		🟩		🟩	🟩								🟩		🟩							🟩		🟩	🟩		
app	NEST Desktop	🟨			🟨	🟨			🟩			🟥🟨			🟥🟨			🟩			🟩		🟨				🟩						🟥🟩	
app	The Virtual Brain (TVB)	🟩	🟩	🟩	🟩	🟨	🟩	🟨	🟩	🟩	🟩	🟥🟩						🟩		🟩	🟩		🟩			🟨							🟥🟩	🟩
app	SDA	🟥			🟥🟩	🟨			🟥🟩												🟩													
app	CGMD Platform	🟨	🟨	🟨	🟨	🟨			🟨																									
app	SSB Platform / METNET	🟨	🟨	🟨	🟨	🟨			🟨																									
app	MoDEL-CNS	🟨	🟨	🟨	🟨	🟨			🟨																									
app	BioExcel-CV19	🟨	🟨	🟨	🟨	🟨			🟨																									
app	Small Circuit In Silico Experiments Tool (Rat Hippocampus CA1)	🟨			🟨	🟨			🟨																									🟩
app	Brain Areas Circuit In Silico Experiments Tool (Rat Hippocampus CA1)	🟨			🟨	🟨			🟨																									🟩
app	Neuron as a Service (aka BlueNaaS aka Single Cell In Silico Experiment tool)	🟨			🟨	🟨			🟨			🟥🟩																						🟩
app	SimVisSuite	🟨							🟨		🟨										🟨						🟩							
app	SimDaVis - Insite	🟨							🟨	🟨								🟩	🟨		🟩		🟨				🟩							
app	Neuromorphic Computing Job Manager app	🟩	🟩	🟥🟩	🟥🟩	🟥🟩	🟨	🟩	🟩	🟨		🟥🟨		🟩								🟩												
app (Jupyter)	Rebuild an existing single hippocampal cell model	🟩	🟩	🟩	🟩	🟨	🟩				🟩	🟩								🟩									🟩					
app (Jupyter)	Build your own single hippocampal cell model using HBP data	🟩	🟩	🟩	🟩	🟨	🟩				🟩	🟩								🟩									🟩					
app (Jupyter)	Synaptic events fitting	🟩	🟩	🟩	🟩	🟨	🟩				🟩									🟩														
app & service	Model Catalog	🟩	🟩	🟩	🟩	🟨	🟩	🟥🟨	🟩	🟨											🟨	🟨	🟨			🟩								
library	Electrical model building toolset											🟥🟩							🟥🟩	🟩		🟩		🟥🟩				🟩	🟩		🟥🟩	🟥🟩		
library	PyNN		🟩					🟩		🟥🟩		🟥🟩			🟨			🟩	🟨	🟩	🟨	🟩		🟨										
library	NESTML		🟩		🟨			🟨				🟥🟩		🟥🟨				🟩			🟨													
library	Snudda		🟨	🟨			🟨			🟩		🟩								🟩						🟩		🟩	🟩					
library	Brain Scaffold Builder									🟥🟩		🟩						🟩	🟩	🟩	🟨			🟨										
library	NEST		🟩						🟩	🟩	🟥🟩	🟥🟩		🟩	🟩		🟩	🟩	🟨		🟩	🟨		🟥🟩			🟥🟩			🟥🟩			🟥🟩	
library	Arbor		🟥🟩	🟥🟩	🟥🟩				🟥🟩	🟥🟩	🟥🟩	🟥🟩					🟩	🟨	🟩		🟩	🟥🟨		🟥🟩		🟥🟩			🟥🟩					🟥🟩
library	NEURON/coreNEURON		🟩							🟩		🟥🟩		🟩			🟩				🟨										🟥🟩			
library	Multiscale, multi-simulator, co-simulation	🟥	🟩	🟨	🟥	🟥	🟥		🟥	🟩	🟩			🟥	🟨		🟥	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟥		🟩			🟩			🟨	
library	Neo		🟩							🟩											🟩													
library	Elephant		🟩			🟨				🟩											🟩	🟩			🟩									
library	LFPy		🟥🟩	🟥🟩		🟩			🟩	🟩	🟩						🟨	🟥🟩	🟩	🟩	🟩	🟩	🟩					🟨	🟨	🟨				
library	Elephant Visualization (Viziphant)		🟨																		🟨	🟩	🟩											
library	Validation test libraries		🟨							🟨		🟩	🟩	🟩	🟨	🟨		🟩	🟨	🟩	🟨	🟩	🟩	🟨	🟨									
library	Molecular level tools (apps and scripts), e.g. MD-IFP, PIPSA		🟥🟩	🟩	🟩		🟩		🟩	🟩	🟨																							
library/workflow	Subcellular model building toolset		🟥🟩	🟨	🟥🟩				🟨	🟥🟩	🟥🟩	🟥🟩								🟥🟩						🟨								
workflow/app	Jupyter Notebook for computing dissociation rates using tauRAMD method		🟥🟩	🟥🟩	🟥🟩				🟨	🟨	🟨																							
workflow/app	JN and scripts for using molecular level tools (to port from HBP to Ebrains platform)		🟥🟩	🟩	🟩		🟩		🟩	🟩	🟨	🟨																						

Green indicates that the tool or service in the left-hand column uses or can easily be run on the tool or service listed in the columns. Yellow means that such integration is planned, partial, or in progress. Red frames indicate changes since D5.5 (M21). It should be noted that this table is meant to guide the user as to which use scenarios are possible, i.e., not all combinations are desirable/useful nor are they planned (for example, libraries would typically not integrate with IAM, but the applications using the libraries would).

5. Recent Changes and Future Plans

5.1 Building Models

5.1.1 Molecular Level Toolset

Type: command-line tool or library; web app
Scope: molecular

Abstraction level: nanoscale
Integration Status: open source, EBRAINS software stack (since EBRAINS 22-10), reachable under custom domain

EBRAINS since 22.10

Homepage & more information:

- Multipipsa: https://search.kg.ebrains.eu/instances/34c044c3-4237-4947-b1bf-8ebf653787eb
- τ-RAMD:
  - https://search.kg.ebrains.eu/instances/774914a7-f3fe-4eb0-97ae-723629d0e3da
  - https://github.com/HITS-MCM/tauRAMD
- MD-IFP:
  - https://search.kg.ebrains.eu/instances/61e709fd-45b6-4808-9d7e-030db4f1e574
  - https://github.com/HITS-MCM/MD-IFP/tree/master/examples
- Documentation: https://humanbrainproject.github.io/hbp-sp6-guidebook/online_usecases/molecular_level/molecular_level.html

Recent Changes (M22-M42): As part of the Molecular Level Toolset, the use of τRAMD[7] and MD-IFP[8] for computing dissociation rates and mechanisms was extended to larger scale systems, i.e., protein-protein systems, proving good performance and usability. For this purpose, FENIX resources were used to fine-tune the code and perform extensive testing. The Gromacs RAMD simulation code was compiled on the JUSUF supercomputer for its use by EBRAINS users. Additionally, Jupyter notebooks for computing unbinding rates by post-processing τRAMD simulation data were refined and adapted to be run in the EBRAINS Collaboratory. Scripts with detailed documentation can be found on EBRAINS[9]. A more complete description and documentation of the overall workflow is found on KBbox[10].

Further changes also regard the integration of the apbs and pdb2pqr software on EBRAINS (with use of Spack) that made it possible to use Jupyter notebooks for running multipipsa[11] on the EBRAINS Collaboratory (EBRAINS-22-10). Scripts and improved documentation are found on EBRAINS[12].

The following additional molecular level tools have also been shared in the Knowledge Graph:

- Distance-Fluctuations (DF) Analysis[13]: performs analysis of the results of a MD simulation using the Distance Fluctuation matrices (DF), based on the Coordination Propensity (CP) hypothesis.

---

[7] Kokh et al. (2019) Machine Learning Analysis of τRAMD Trajectories to Decipher Molecular Determinants of Drug-Target Residence Times. *Frontiers in Molecular Biosciences*, 6. DOI: 10.3389/fmolb.2019.00036 (P2008)

[8] Kokh & Wade (2021) G Protein-Coupled Receptor-Ligand Dissociation Rates and Mechanisms from τRAMD Simulations. *Journal of Chemical Theory and Computation*, 17:10. DOI: 10.1021/acs.jctc.1c00641 (P2945)

[9] https://wiki.ebrains.eu/bin/view/Collabs/protein-ligand-unbinding-rates

[10] https://kbbox.h-its.org/toolbox/tutorials/estimation-of-relative-residence-times-of-protein-ligand-complexes-using-random-acceleration-molecular-dynamics-ramd-implementation-in-gromacs/

[11] Bruce et al. (2019) Regulation of adenylyl cyclase 5 in striatal neurons confers the ability to detect coincident neuromodulatory signals. *PLOS Computational Biology*, 15:10. DOI: 10.1371/journal.pcbi.1007382 (P2205)

[12] https://wiki.ebrains.eu/bin/view/Collabs/computation-of-protein-association-rates/

[13] https://search.kg.ebrains.eu/instances/3647a7eb-a233-440a-8b57-87fed1ec4c01

- MEDIUM[14]: uses DF matrices to extract global information on the protein.
- MLCE[15]: predicts the region in a protein most likely to be involved in interactions with external partners.

Key improvements in SGA3: Expanded usability of the tools to more complex biomolecular systems, from soluble protein-small molecule systems to membrane protein-small molecule systems to protein-protein systems. A key improvement was the migration of the tools to enable use via the EBRAINS Collaboratory.

Outlook: Completion of ongoing applications to protein-protein dissociation rate calculations and applications of the tools to new case studies as well as further tools for further facilitating combined use of molecular and subcellular level tools are planned.

5.1.2 *CNS Ligands*

Type: web app Scope: molecular	
Abstraction level: nanoscale Integration Status: reachable under custom domain Homepage & more information: • http://mmb.irbbarcelona.org/webdev/slim/CNS/public/ • https://search.kg.ebrains.eu/instances/696d987f-128d-467b-a91b-1056a4f7dc7e • Documentation: https://wiki.ebrains.eu/bin/view/Collabs/molecular-tools-central-nervous-system	

Recent Changes (M22-M42): A Jupyter Notebook titled "*Extracting relevant information from atomistic Molecular Dynamics trajectories using the Central Nervous System API*" has been integrated in the EBRAINS Collab infrastructure. The notebook shows how to programmatically extract, play, store, and visualize information from the database. It was used in the "*Second HBPMolSim Training Workshop on Tools for Molecular Simulation of Neuronal Signalling Cascades*" training event on 21-23 June 2023.

Key improvements in SGA3: The database has been integrated in the new EBRAINS infrastructure, with a Jupyter Notebook showing examples on how to extract, play, store, and visualize information from the database.

Outlook: New bioactive conformations for ligands binding to neuronal proteins will be progressively added.

5.1.3 *GRALL*

Type: web app Scope: molecular	
Abstraction level: nanoscale Integration Status: reachable under custom domain Homepage & more information: • https://ifm.chimie.unistra.fr/grall	

Recent Changes (M22-M42): GRALL is now represented on the EBRAINS website.

---

[14] https://search.kg.ebrains.eu/instances/7c3dbeb1-cb2c-4fab-910d-9fe0ca8709e9
[15] https://search.kg.ebrains.eu/instances/a164f8e6-bc1a-4c17-9bb0-0dab0014b3d2

Key improvements in SGA3: The Glycine Receptor Allosteric Ligand Library GRALL is the first database of allosteric modulators of a human synaptic receptor with a structural annotation based on their binding site on the receptor. The GRALL library has been made publicly available as a web-accessible database.

Outlook: We continue to encourage researchers to contribute to the database.

5.1.4 Subcellular Web App

Type: web app Scope: subcellular	💻✔⇄ 👥🗣
Abstraction level: systems biology Integration Status: Reachable under apps.hbp.eu domain and custom domain Homepage & more information: • https://subcellular.humanbrainproject.eu • Documentation: https://humanbrainproject.github.io/hbp-sp6-guidebook/online_usecases/subcellular_level/subcellular_app/subcellular_app.html	

Recent Changes (M22-M42): We tested more model files in the workflow, which allowed bugs to be identified. We developed an API to allow the backend to be accessed directly. The code was released on GitHub[16] and we put in place continuous integration. The visualization capability of the app was improved as well.

Key improvements in SGA3: Identified bugs were fixed. More documentation[17] was written. We added a landing page[18] and made the app more navigable.

Outlook: Ongoing efforts to fully migrate the app to the ebrains.eu domain and improve performance.

5.1.5 Subcellular Model Building Toolset

Type: library Scope: subcellular	⚙🔧⇄ 🗣
Abstraction level: system biology Integration Status: EBRAINS software stack (since EBRAINS 23-03) Homepage & more information: • https://github.com/icpm-kth/	EBRAINS since 23.02

Recent Changes (M22-M42): We have re-worked the code to use many closures. This change makes a better distinction between frequently changing function arguments and rarely changing arguments (which require re-building of the closures), with closures function arguments that don't change between calls become implicit and don't need to be specified. This makes the code easier to maintain, it is easier to continue development and the overall package becomes easier to use for the end-user. In addition, we improved the usability of our R interface to the GSL solvers for initial value problems. Now, we create a list of simulation experiments which relate measured data to instructions for the simulator. This also includes scheduled events which we use to model the interventions that lab technicians use to perform the experiment, e.g.: an inhibitor is added to the system (a sudden change in the model's state is performed by the event system, on our side). The events can be specified in the SBtab document and carried through to the simulator through the entire toolchain. We added probabilistic acceptance rules; previously we used strict cut-offs for ABC

---

[16] https://github.com/BlueBrain/BlueNaaS-Subcellular
[17] https://subcellular-bsp-epfl.apps.hbp.eu/static/docs.html
[18] https://subcellular-bsp-epfl.apps.hbp.eu/home/

proposal acceptance. In addition to that, we added a back-end for fully stochastic models (with Gillespie algorithm solvers). Documentation was improved, some more tests added.

Key improvements in SGA3: Package usability and clarity have been improved, more sampling methods and elaborated documentation are available, and missing features were added as planned.

Outlook: We are aiming for web presence for each package (through GitHub pages), continuous integration on EBRAINS GitLab servers to improve reliability, and improved performance.

5.1.6 *Feature Extraction Tool (NeuroFeatureExtract)*

Type: web app Scope: single neurons	💻✔ ⏳ 👥 🎧
Abstraction level: biophysical models Integration Status: Reachable under custom domain Homepage & more information: • https://hbp-bsp-hhnb.cineca.it/efelg/ • https://search.kg.ebrains.eu/instances/Software/348d66fb-8e82-4e1b-b266-db886d6ca1b9 • Documentation: https://ebrains-cls-interactive.github.io/	

Recent Changes (M22-M42): We have improved the stimulus detection and the voltage correction management, both for the electrophysiology traces selected from the list and for those uploaded by the user.

Key improvements in SGA3: The stimulus and voltage correction management.

Outlook: We plan to tighten the interaction of the tool with the EBRAINS KG by including new types of electrophysiological recordings.

5.1.7 *Hodgkin-Huxley Neuron Builder (HHNB)*

Type: web app Scope: single neurons	💻✔ ⏳ 👥 🎧
Abstraction level: biophysical models Integration Status: Reachable under custom domain, supports login via EBRAINS credentials Homepage & more information: • https://hbp-bsp-hhnb.cineca.it/hh-neuron-builder/ • https://kg.ebrains.eu/search/instances/Software/0295cfba-f9d6-4743-be2b-4628f7c5ab58 • Documentation: https://ebrains-cls-interactive.github.io/index.html	

Recent Changes (M22-M42): The "Optimization Settings" and the "Fetch Results" layouts, in case the Service Account utility is chosen, were improved. It is now possible to select the HPC system to be used and from which project the resources should be drawn.

A new Dialog Window was added in case the users are redirected to the HHNB from the Hippocampus Hub. Via this interface, the user can select which "parameters.json" template to use in order to build the neuron model (the choice is currently between "Pyramidal cell" and "Interneuron cell").

In the "Single Cell Simulation Run" section, a new button was introduced to visualize the result figures generated from the analysis performed on the optimized model (i.e., no file download is required for the visualization).

Finally, the possibility to resume a previous optimization is now offered to the users. This step is carried out by uploading an already optimized model and, after appropriately changing the parameters of interest, continue the optimization process by running additional iterations of the genetic algorithm.

Key improvements in SGA3: The possibility to resume the optimization process by running additional iterations of the genetic algorithm.

Outlook: We plan to add the CINECA HPC system as an additional choice for the user, both for usage with the user's EBRAINS account and for the service account utility.

5.1.8 *Electrical Model Building Toolset*

Type: command-line tool or library
Scope: single neurons

Abstraction level: biophysical neuron models
Integration Status: open source, EBRAINS Software stack (since EBRAINS 22.07)
Homepage & more information:

BluePyOpt: EBRAINS since 23.06

- https://github.com/BlueBrain/BluePyOpt
- https://kg.ebrains.eu/search/instances/Software/d2918247-8f25-4252-8efb-906e8190d5cd

eFEL: EBRAINS since 22.07

- https://github.com/BlueBrain/eFEL
- https://kg.ebrains.eu/search/instances/Software/42c9ca46-1689-4494-85b5-33f6dffc1ac7

BluePyEfe: EBRAINS since 23.06

- https://github.com/BlueBrain/BluePyEfe
- https://search.kg.ebrains.eu/?category=Software&q=bluepyefe#9af1af71-6842-4a35-8218-4306240376dc

BluePyMM: EBRAINS since 23.06

- https://github.com/BlueBrain/BluePyMM
- https://search.kg.ebrains.eu/?category=Software&q=bluepymm#62e28f34-0080-43b7-9e40-f793193b8cd8

Recent Changes (M22-M42): BluePyOpt has now added support for Arbor, LFPy, and NeuroML, broadening the possibilities for users who seek to employ these platforms for their neuronal modelling needs. Alongside this, the release of BluePyEfe 2.0 represents a significant update, introducing a much-improved interface to enhance user experience and streamline operation. We have also made sure that Zenodo references for all the tools are available[19], providing users with easy access to critical resources. Lastly, we have improved and extended the documentation for eFEL to ensure comprehensive understanding and smooth application of this tool. Overall, these updates are part of our continued commitment to supporting neuroscientists in their research endeavours.

Key improvements in SGA3: The Electrical Model Building Toolset improved interoperability in the community by adding support for more formats and simulators. Based on feedback of users we have redesigned the API of BluePyEfe. By improving the install-ability and documentation of all the tools we made it easier for users to use the software.

Outlook: The tools in the Electrical Model Building Toolset will keep on being maintained and extended by the Blue Brain Project. Some specific plans include the move to more and more electrical features in eFEL being developed in Python instead of C++. We will also soon release an extra package in the toolset: BluePyEModel which will work as a tool that encapsulates the workflow generally used in the toolset.

---

[19] https://zenodo.org/record/8272830, https://zenodo.org/record/8288734, https://zenodo.org/record/6006925, https://zenodo.org/record/8220850

5.1.9 PyNN

Type: command-line tool or library Scope: networks	
Abstraction level: point neuron models, biophysical neuron models Integration Status: open source, EBRAINS stack (since EBRAINS-21.09) Homepage & more information: • http://neuralensemble.org/PyNN/ • https://kg.ebrains.eu/search/instances/Software/bb06ac59-9ada-4b1d-a00a-e2fcafb031f8	EBRAINS since 21.09

Recent Changes (M22-M42): While the PyNN API for point neuron models has remained largely stable, in preparation for the forthcoming v1.0 release, considerable efforts have gone into improving the code base: fixing bugs, adding more test cases, improving the developer experience, for both existing and new developers, and bringing PyNN up to date with current best practices for open-source Python projects. One major new feature has been added to the point neuron API: neuron models can now be composed from a menu of membrane mechanisms and of an arbitrary number of post-synaptic response mechanisms, giving greater flexibility. In addition, where a simulator supports recording with varying sampling intervals (e.g. BrainScaleS, NEURON with CVode), PyNN can now handle such data.

PyNN version 0.10.1 added support for NEST v3.3, and irregular sampling interval support. Version 0.11.0 added support for NEST v3.4 and NEST v3.5, and composed neuron models.

In contrast to the point neuron API, the new API for biophysically and morphologically-detailed models has seen rapid evolution, as adding support for the Arbor simulator required many changes to the original, NEURON-based prototype. A first beta version of PyNN 2.0, with support for both NEURON and Arbor, has now been released.

Key improvements in SGA3: The PyNN point neuron API, and its implementations for NEST, NEURON, Brian, SpiNNaker and BrainScaleS, are mature, well-tested and well-documented. The main foci in SGA3 have been on improving code quality and maintainability, and tracking the evolution of the underlying simulators. The biophysical neuron API has moved from a proof-of-concept with support for only a single simulator, to a beta release with support for both NEURON and Arbor.

Outlook: Release of version 1.0 of the PyNN point neuron API is expected before the end of 2023. The biophysical API will continue to be developed, with support for additional simulators/neuromorphic systems planned (Eden and BrainScaleS-2), and a final v2.0 release in 2024.

5.1.10 NESTML

Type: command-line tool or library Scope: single neurons, networks	
Abstraction level: point neuron models Integration Status: open source, EBRAINS software stack (since EBRAINS-23.02) Homepage & more information: • https://nestml.readthedocs.io • https://kg.ebrains.eu/search/instances/Software/95472c51-772c-42dd-a427-553851f83c2b	EBRAINS since 23.02

Recent Changes (M22-M42): Five versions of NESTML have been released in this reporting period (5.0.0-rc2, 5.0.0, 5.1.0, 5.2.0, 5.3.0[20]). They contain enhancements in software architecture, improved user experience, new models, and documentation updates. NESTML has consistently coordinated with and tracked NEST Simulator releases (with concomitant API changes), ensuring full

---

[20] https://github.com/nest/nestml/releases/

compatibility between the two on a continuous basis; this is supported by modern software development techniques such as continuous integration.

Refactoring of the internal software architecture has increased the versatility in generating code for different target platforms, by making it easier to use a different set of code generation templates. These improvements are the foundation for the addition of SpiNNaker target platform support and NEST-GPU platform support. Interactive tutorials on the adaptive neuron model and the dopamine-modulated STDP synapse model were added. Support was added for delay differential equations to further support advanced synaptic plasticity rules such as the "Clopath" rule that exists in NEST.

Essential components of the NESTML syntax were improved for consistency and a clearly defined semantics, supporting the precision of NESTML as a model specification language while increasing reusability and reproducibility of the models by the NESTML end user community.

Key improvements in SGA3: A SpiNNaker development environment was established, which was used as infrastructure to implement SpiNNaker support in NESTML. This allows all models required for the multi-area model to be generated through NESTML (T3.2). Maturity of the NESTML software component was improved by refactoring, documentation and tutorial improvements and new language features, many of which were based on feedback from (interactive) workshops where NESTML is presented to the neuroscience community.

Outlook: Our coming focus for NESTML will be on advanced plasticity rules and neuromorphic hardware, including GPU support. These plasticity rules link machine learning to biologically realistic mechanisms. Implementing them in NESTML will increase scientific insights by allowing a rapid development cycle, and yield several new, interactive use case demonstrators that would increase the size and diversity of the NESTML user base.

5.1.11 Snudda

<table>
<tr><td>Type: command-line tool or library
Scope: single neurons, networks</td><td></td></tr>
<tr><td colspan="2">Abstraction level: biophysical neurons
Integration Status: open source, EBRAINS Software Stack (since EBRAINS 23.02) EBRAINS since 23.02
Homepage & more information:
• Snudda:
○ https://search.kg.ebrains.eu/instances/e9390ad8-3167-43bd-b04d-fcfdccb59ac1
○ https://github.com/Hjorthmedh/Snudda
○ https://pypi.org/project/snudda/
○ https://snudda.readthedocs.io/en/latest/
• treem:
○ https://pypi.org/project/treem
○ https://github.com/a1eko/treem
○ https://treem.readthedocs.io/en/latest/</td></tr>
</table>

Recent Changes (M22-M42): Snudda generates microcircuits with connectivity constrained by reconstructed morphologies using a touch-detection algorithm. Co-developed Python module `treem` is an open-source tool for processing of the single-cell morphologies for Snudda simulations.

The Snudda generated network can now be exported to SONATA format, and the files have been successfully imported into NEST. Additional improvements have been made in the hdf5 output format, reducing the file size. Snudda can now model the connectivity changes due to neurodegeneration (e.g., Parkinson’s disease). This requires two copies of each morphology, one WT version, and one PD version. From the WT version of the morphology treem can generate the PD version by removing the tips of the dendritic branches. Snudda is a critical component of the basal ganglia studies, originally developed for striatum by Hjorth et al., 2021[21] and recently generalized to other BG nuclei and brain regions by Carannante et al., 2022[22], Trpevski et al., 2023[23], Grillner and Thompson, 2023[24].

Key improvements in SGA3: Performance improvements and ability to export networks in SONATA format. Ability to model network changes due to neurodegeneration (and axonal/dendritic growth).

Functionality and performance improvements in the development version of the Python module `treem` on GitHub.

Outlook: Ongoing testing of Snudda by the user community leading to the production use for upcoming publications and three PhD projects (Carannante, Khodadadi, Thompson).

5.1.12 Brain Scaffold Builder

Type: command-line tool or library Scope: single neurons, networks	
Abstraction level: biophysical neurons, point neurons Integration Status: Open source Homepage & more information: • https://github.com/dbbs-lab/bsb • https://bsb.readthedocs.io • https://search.kg.ebrains.eu/instances/356b837a-0dda-4b05-b94c-829fa3d22a31	

Recent Changes (M22-M42): The Brain Scaffold Builder (BSB) has undergone a radical redesign to decouple it from its origin as a tool to model the cerebellum, to a generalist black-box component framework that offers component-based solutions for every aspect of multiscale brain modelling. Every aspect of the workflow has been captured into a component-type with two key parts: 1) the "invariant" framework function (components have several internal functions for the framework to rely on and execute its core functions) and 2) the "variant" model-specific function (which the user provides and the framework uses to construct the model). Together these invariant parts are now capable of drastically reducing the workload associated with creating brain models, and the invariant parts offer the flexibility each user needs to model their specific brain region. This redesigning is nearing completion with the latest alpha stages finalized. We expect to release our first public beta version until the end of SGA3: the beta marks an increase in stability and maturity, and users can begin building models without requiring the close collaboration with developers that they needed before.

Key improvements in SGA3: BSB has matured into a stable component framework for multiscale brain models and rapid prototyping. New connection strategies, new pipelines to manage the neuron morphologies (orientation, deformation, …), new direct interfaces with atlases, have been introduced. The BSB is an EBRAINS component, available to use on the EBRAINS infrastructure, and can be conveniently installable on any machine or platform where Python is available. Any model can be described in a code-free configuration of components with parameters.

Outlook: Development of BSB will be continued to improve the public API, based on user feedback, to optimize the parallelization and storage, to integrate new modules (e.g., LFP, BOLD simulation, Nitric oxide production and diffusion). Dedicated computer scientists are being involved. The

---

[21] Hjorth et al. (2021) Predicting Synaptic Connectivity for Large-Scale Microcircuit Simulations Using Snudda. *Neuroinformatics*. DOI: 10.1007/s12021-021-09531-w (P2899)

[22] Carannante et al. (2022) Data-Driven Model of Postsynaptic Currents Mediated by NMDA or AMPA Receptors in Striatal Neurons. *Frontiers in Computational Neuroscience*, 16. DOI: 10.3389/fncom.2022.806086 (P3872)

[23] Trpevski et al. (2023) Glutamate spillover provides robust all-or-none behavior of plateau potentials in multicompartment models of striatal projection neurons. DOI: 10.22541/au.168012861.16093883 (P3979)

[24] Grillner and Thompson (2023) Basal ganglia reign through downstream control of motor centers in midbrain and brain stem while updating cortex with efference copy information. *Neuron*, 109:10. DOI: 10.1016/j.neuron.2021.04.015 (P3184)

modelling Hub and the Digital Neuroscience Centre (IRCCS Mondino Foundation, Pavia) will support the tool ecosystem. The main focus during beta development will be on quality assurance and user experience.

5.1.13 The Virtual Brain (TVB)

Type: command-line tool or library & web app
Scope: whole brain

Abstraction level: population models
Integration Status: EBRAINS Software Stack (since EBRAINS-21.09)
EBRAINS since 23.02
Homepage & more information:

- https://www.thevirtualbrain.org/tvb/zwei
- https://kg.ebrains.eu/search/instances/Software/316edfdb-abd7-4451-b54e-21b746838ec0

Recent Changes (M22-M42): During this period, RateML was further refined to support region specific data and complex models like the Zerlaut model. A manuscript about RateML has now been published[25]. TVB and RateML have been further integrated within L2L workflows for model optimization[26].

High resolution (HR) simulations have also been further developed during this period.

A series of widgets and GUIs have been developed with the goal of enhancing the usability of EBRAINS tools from the Collaboratory, with particular focus on the showcases in the scientific work packages.

In order to enable accessing and manipulating data from new sources in TVB, the new data source, i.e., the EBRAINS Knowledge Graph, can now be easily accessed both in a command line environment, using a simple API, and in the TVB Web application. Both methods are used to query the KG in order to obtain structural and functional connectivities. Graphical components have been developed for 1) inspecting unicore jobs from the EBRAINS Collab running on available FENIX HPC systems, 2) visualising structural connectivity, head surfaces and sensors (Surface 3D widget), 3) interactively inspecting Timeseries results (TimeSeries widget), and 4) querying the Drive. Both the extension and the widgets have been integrated into the EBRAINS JupyterLab environment.

Key improvements in SGA3: Testing, integration and formalization into training material of workflows including L2L and TVB for efficient parameter optimization on HPC resources. Refinement of HR simulations with neural fields and inference software. Enhanced usability and integration through Widgets and extensions which provide users new avenues to design and launch scientific workflows using the Collaboratory.

Outlook: We are planning 1) to integrate HR simulations and inference tools with RateML and L2L in support of new use cases in clinical research, cognitive architecture development and multiscale simulations, and 2) for further extensions and Widgets which allow data-centric interoperability between EBRAINS tools.

---

[25] van der Vlag et al. (2022) RateML: A Code Generation Tool for Brain Network Models. *Frontiers in Network Physiology*, 2. DOI: 10.3389/fnetp.2022.826345 (P3790)

[26] Yegenoglu et al. (2022) Exploring Parameter and Hyper-Parameter Spaces of Neuroscience Models on High Performance Computers With Learning to Learn. *Frontiers in Computational Neuroscience*, 16. DOI: 10.3389/fncom.2022.885207 (P3763)

5.1.14 *Single Cell Modelling Workflows*

Type: command-line tool or library Scope: single neurons	
Abstraction level: biophysical models Integration Status: runs on EBRAINS software stack Homepage & more information: • https://ebrains-cls-int eractive.github.io/	

Recent Changes (M22-M42): Maintenance operations were performed in the "Single Cell Modelling Workflows" components, with no major changes implemented.

Key improvements in SGA3: All Jupyter notebooks have been migrated to Python 3 and run on EBRAINS Jupyter.

Outlook: We are planning to improve the user experience with the notebook by enhancing its interactivity and visual appearance.

5.2 Running Simulations

5.2.1 *CGMD Platform*

Type: web app Scope: networks	
Abstraction level: nanoscale Integration Status: open source, reachable under custom domain (iframe in EBRAINS wiki) Homepage & more information: • https://molsim.sci.univr.it/mermaid/begin.php • https://search.kg.ebrains.eu/instances/2bd1f14b-e6a1-45d3-8a80-4ca7bc3d74ae	

Recent Changes (M22-M42): CGMD is now represented on the EBRAINS website.

Key improvements in SGA3: The web server is online, working and available via the EBRAINS Collaboratory[27].

Outlook: We plan to include automatic generation of membrane proteins' oligomeric states.

5.2.2 *SSB Platform*

Type: web app and standalone app Scope: subcellular	
Abstraction level: combined nanoscale and systems biology Integration Status: open source Homepage & more information: • https://ssbtoolkit.readthedocs.io/en/latest/ • https://pypi.org/project/ssbtoolkit/ • https://search.kg.ebrains.eu/instances/073e647e-0836-4075-b88b-ddeb74cbeea0	

Recent Changes (M22-M42): The last release (v1.0.6) of the SSB toolkit stands out for its user-friendly design, extensive documentation, and innovative features. Its intuitive interface and clear

---

[27] https://wiki.ebrains.eu/bin/view/Collabs/molecular-tools-cgmd-platform/

workflows make it accessible to researchers of various backgrounds, while comprehensive documentation ensures a smooth learning experience. The toolkit incorporates innovative methods, such as integrating structural biology data and systems biology simulations, allowing for a deeper understanding of drug-receptor interactions.

Key improvements in SGA3: The SSB Toolkit has evolved into a reliable framework for simulating the signal-transduction kinetics of synaptic G-protein coupled receptors. It is an EBRAINS component, accessible on all EBRAINS infrastructure, and can be easily installed on any machine or platform that supports Python.

5.2.3 MoDEL-CNS

Type: web app Scope: molecular, subcellular	
Abstraction level: nanoscale Integration Status: reachable under custom domain (iframe in EBRAINS wiki) Homepage & more information: • http://mmb.irbbarcelona.org/MoDEL-CNS/ • https://wiki.ebrains.eu/bin/view/Collabs/molecular-tools-model-cns/	

Recent Changes (M22-M42): A Jupyter Notebook titled "*Extracting relevant information from atomistic Molecular Dynamics trajectories of relevant signal transduction proteins using MoDEL_CNS API*" has been integrated in the EBRAINS Collab (via iframe). The notebook shows how to programmatically extract, play, store, and visualize information from the database. It was used in the "*Second HBPMolSim Training Workshop on Tools for Molecular Simulation of Neuronal Signalling Cascades*" training event on 21-23 June 2023.

Key improvements in SGA3: The database has been integrated in the new EBRAINS infrastructure, with a Jupyter Notebook showing examples on how to extract, play, store, and visualize information from the database.

Outlook: A new set of simulations of proteins related to the Parkinson disease are being run and will be uploaded into the database in the coming months. The database will be progressively updated.

5.2.4 BioExcel-CV19

Type: web app, command-line tool or library Scope: molecular, subcellular	
Abstraction level: nanoscale Integration Status: reachable under custom domain (redirect from EBRAINS wiki) Homepage & more information: • https://bioexcel-cv19.bsc.es/#/	

Recent Changes (M22-M42): Three Jupyter Notebooks have been integrated in the EBRAINS Collab:

- Extracting relevant information from atomistic Molecular Dynamics trajectories using the BioExcel-Covid19 API
- ACE2-RBD interaction analysis for different mutations
- Protein-Ligand interaction analysis for different ligands

The notebooks show how to programmatically extract, play, store, and visualize information from the database. They were used in the "*Second HBPMolSim Training Workshop on Tools for Molecular Simulation of Neuronal Signalling Cascades*" training event on 21-23 June 2023.

Key improvements in SGA3: The database has been integrated in the new EBRAINS infrastructure, with Jupyter Notebooks showing examples on how to extract, play, store, and visualize information from the database.

Outlook: New simulations of proteins related to the COVID-19 disease are continuously being uploaded into the database.

5.2.5 BioExcel Building Blocks Workflows

Type: web app, command-line tool or library
Scope: molecular, subcellular

Abstraction level: nanoscale
Integration Status: Open source, EBRAINS software stack (since EBRAINS 23.06)
EBRAINS since 23.06
Homepage & more information:

- https://mmb.irbbarcelona.org/biobb/
- https://wiki.ebrains.eu/bin/view/Collabs/biobb

Recent Changes (M22-M42): A new collab has been generated and integrated into the new EBRAINS infrastructure. It will enclose a set of biomolecular simulation workflows that will help in the analysis of flexibility and dynamics of macromolecules, including signal transduction proteins or molecules related to the Central Nervous System. Molecular dynamics setup for protein and protein-ligand complexes are examples of workflows available as Jupyter Notebooks. The workflows are built using the BioBB software library, developed in the framework of the BioExcel Centre of Excellence. BioBB is a collection of Python wrappers on top of popular biomolecular simulation tools, offering a layer of interoperability between the wrapped tools, which make them compatible and prepared to be directly interconnected to build complex biomolecular workflows. The library has been integrated in the EBRAINS infrastructure through the generation of new Spack packages.

Key improvements in SGA3: The BioBB library has been integrated in the new EBRAINS infrastructure, with Jupyter Notebooks showing examples of biomolecular workflows to explore the flexibility and dynamics of macromolecules, including signal transduction proteins or molecules related to the Central Nervous System.

Outlook: New workflows will be progressively added, adapting the ongoing work of the BioExcel Centre of Excellence to molecules related to the Central Nervous System, and integrating new tools in the EBRAINS infrastructure using new Spack packages.

5.2.6 SDA

Type: web app and standalone app
Scope: molecular

Abstraction level: nanoscale

Integration Status: open source, reachable under ebrains.eu domain, EBRAINS stack since 23.02
EBRAINS since 23.02

Homepage & more information:

- https://mcm.h-its.org/sda
- https://websda.h-its.org/
- https://search.kg.ebrains.eu/instances/a65c722a-71e9-4eba-bce6-fdc7ef736dc3

Recent Changes (M22-M42): SDA 7.3.3 and 7.3.4 were released with support of new gcc compiler versions. With these changes the Spack configuration could be used to integrate SDA into the EBRAINS Jupyter notebook server for use in the EBRAINS Collaboratory (EBRAINS-23.02). Scripts to analyse the results from Brownian dynamics simulations and computing association rates can be found at https://wiki.ebrains.eu/bin/view/Collabs/computation-of-protein-association-rates/ with revised documentation.

Key improvements in SGA3: Migration of the tool to the EBRAINS Collaboratory.

Outlook: Further improvements of the simulation methodology and its application to computing protein-small molecule association rate constants and studying protein-surface adsorption are ongoing.

5.2.7 *Single Cell In Silico Experiment Tool (BlueNaaS)*

Type: web app Scope: single neurons	
Abstraction level: biophysical models Integration Status: Open source Homepage & more information: • https://github.com/BlueBrain/BlueNaaS-SingleCell • https://blue-naas-bsp-epfl.apps.hbp.eu • https://ebrains-cls-interactive.github.io/online-use-cases.html	

Recent Changes (M22-M42): Running single cell *in silico* experiments is a central application on EBRAINS for users to explore existing neuron models. This service is now fully migrated to EBRAINS. Furthermore, the API was modified to support the EBRAINS Live Papers, which make frequent use of this neuron simulation service in a variety of the Live Papers.

Key improvements in SGA3: The code base was open sourced, enabling the community to participate in the development of the application.

Outlook: The application will be maintained, but no more features will be added.

5.2.8 *Small Circuit In Silico Experiments Tool (Rat Hippocampus CA1)*

Type: web app Scope: networks	
Abstraction level: biophysical neurons Integration Status: reachable under apps.hbp.eu domain Homepage & more information: • https://pair-recording-bsp-epfl.apps.hbp.eu/circuits/rat-ca1 • https://ebrains-cls-interactive.github.io/online-use-cases.html	

Recent Changes (M22-M42): An important dependency of an underlying service - BlueCelluLab[28], which is used to set-up and run simulations - has become open sourced, finally allowing the tool's code base to undergo the same procedure. User experience has been improved for touch-based devices, as well as overall performance and reliability.

Key improvements in SGA3: This service has been deployed on a dedicated site, in an EBRAINS-compatible manner, i.e., leveraging the services of the EBRAINS infrastructure, such as the EBRAINS authentication system. The service's functionality has been improved: the possibility to run longer simulations (up to 30 s) has been added, as well as the support for the SONATA format, in view of a Massive Open Online Course[29] that makes use of this service.

---

[28] https://github.com/BlueBrain/BlueCelluLab

[29] https://portal.bluebrain.epfl.ch/resources/online-learning/simulating-a-hippocampus-microcircuit/

Outlook: An additional functionality to replay spikes from existing simulations will be added. The integration of a simulation campaign concept will be considered, which would allow to chain more complex simulation-analysis workflows.

5.2.9 Brain Areas Circuit In Silico Experiments Tool (Rat Hippocampus CA1)

Type: web app Scope: networks	💻 ✔ 🔀 🎓 🧠
Abstraction level: biophysical neurons Integration Status: Open source Homepage & more information: • https://github.com/BlueBrain/BlueNaaS-BrainAreaExperiment • https://simulation-launcher-bsp-epfl.apps.hbp.eu/index.html#/circuits/hippo_hbp_sa_full_ca1 • https://ebrains-cls-interactive.github.io/online-use-cases.html	

Recent Changes (M22-M42): We have worked on improving the documentation[30] and updating the tutorials with the latest integrations, and the codebase for open sourcing on GitHub[31], and enabling compatibility with the latest Unicore 8 version.

Key improvements in SGA3: This service has been deployed on a dedicated site, in an EBRAINS-compatible manner, i.e., leveraging the services of the EBRAINS infrastructure, such as the EBRAINS authentication system. The service's functionality has been improved: the possibility to run longer simulations (up to 30 s) has been added, as well as the support for the SONATA format. The service is integrated in a MOOC with a dedicated tutorial[32]. The service has been open sourced on GitHub[33].

Outlook: The Blue Brain Project will continue the maintenance of this software.

5.2.10 NEST

Type: command-line tool or library Scope: networks	⚙ ✔ 🔀 🎓 👥 🧠
Abstraction level: point neuron models Integration Status: open source, EBRAINS Software Stack (since EBRAINS 21.09) Homepage & more information: • https://nest-simulator.org • https://kg.ebrains.eu/search/instances/Software/c3d23fd5-4370-487b-9231-e0f8e8a1e9e6	EBRAINS since 23.02

Recent Changes (M22-M42): NEST 3.2 release focuses on improving the Python interface making the syntax more intuitive for users. Highlights of the NEST 3.3 release are the improvements in parallel network construction time if large numbers of devices are present due to accelerated node lookups. In specific models this has been shown to reduce network construction time by a factor up to 20, which can strongly increase user productivity. Additionally, full support for generic compartmental neurons was added, which paves the way for studies investigating the effect of neuron model simplifications from rich multi-compartment models to simple point neurons within a single simulation framework. NEST 3.4 adds features for spatially structured networks, such as queries

---

[30] https://github.com/ebrains-cls-interactive/docs/blob/main/source/online_usecases/brain_area_circuit_in_silico_experiments/hippocampus/configure_run_brainregion_preconf_model_data/configure_run_brainregion_preconf_model_data.rst

[31] https://github.com/BlueBrain/BlueNaaS-BrainAreaExperiment

[32] https://www.edx.org/course/simulating-a-hippocampus-microcircuit

[33] https://github.com/BlueBrain/BlueNaaS-BrainAreaExperiment

from target neuron perspective. Further, NEST 3.5 includes native support for networks represented in SONATA format, and new and enhanced models such as the spike_train_injector and access to calcium concentration parameters.

Key improvements in SGA3: The new features for simulations of multi-compartment models and network descriptions in SONATA format act as bridging technologies towards cellular level simulations done with other simulators and are key to validate models across different scales. Furthermore, the solid deployment via the EBRAINS Software Stack allows users to seamlessly run the same scripts on EBRAINS Jupyter, the HPC sites, or own setups. Documentation now centrally points users towards EBRAINS in all examples[34] to directly gain first-hand experience for models and features (installation-free, low-threshold access).

Outlook: The next steps aim to increase the usage of different EBRAINS services by more and improved inter-tool workflows and support of user groups. Establishing stronger connections within the EBRAINS ecosystem allows to switch between and combine different tools more easily, thus enabling users to harness the full power of EBRAINS for their purpose.

5.2.11 Arbor

Type: command-line tool or library Scope: single neurons, networks	
Abstraction level: biophysical models Integration Status: open source, EBRAINS stack since 21.09 Homepage & more information: • arbor-sim.org • docs.arbor-sim.org • https://kg.ebrains.eu/search/instances/Software/80d205a9-ffb9-4afe-90b8-2f12819950ec	EBRAINS since 23.02

Recent Changes (M22-M42): Users can now write custom mechanisms for gap junctions, improving the ability to model synaptic plasticity. A Mechanism ABI enable a user to easily package, maintain, disseminate and archive mechanisms, within their lab or with the wider world. Arbor can now be used as a backend for local field potential calculations through an LFPyKit integration. Ions can propagate along the dendrite by diffusion and can be received by other synapses, mechanism parameters can be scaled along a cell, and both point and density mechanisms may use white noise as part of state updates. Users now can edit the connection table and morphologies on-line through Arbor interfaces. The Arbor cable cell exporter is now complete and used by the recently released Arbor backend in BluePyOpt 1.14.

Key improvements in SGA3: During SGA3 a community has emerged around Arbor. The FIPPA and Arborio CEoI partners have done significant testing of Arbor and have various publications in preparation. As of Arbor v0.8, we co-released Arbor and Arbor GUI, which is a comprehensive tool for building single cell models using Arbor. It strives to be self-contained, fast, and easy to use, and the ideal on-boarding experience for future Arbor users.

Outlook: In the near future we expect to deliver full NeuroML[35] (a simulator independent file format) integration through Nmlcc[36], as well as compatibility with a significant part of the Sonata format. This will unlock large catalogues of models for Arbor users. A new user community aiming to push Arbor into plasticity research is in the process of obtaining various grants.

---

[34] https://nest-simulator.readthedocs.io/en/stable/examples/index.html#pynest-examples

[35] https://neuroml.org

[36] https://github.com/arbor-sim/nmlcc

5.2.12 NEURON/CoreNEURON

Type: command-line tool or library
Scope: single neurons, networks

Abstraction level: biophysical models
Integration Status: EBRAINS software stack (since EBRAINS-21.09)
EBRAINS since 23.02
Homepage & more information:

- https://github.com/BlueBrain/CoreNeuron
- https://github.com/neuronsimulator/nrn
- https://search.kg.ebrains.eu/instances/09873c44-bf70-4e01-a8d6-7fc6ab303070
- https://search.kg.ebrains.eu/instances/16c86cde-4796-418b-a358-ef33667912b4

Recent Changes (M22-M42): We have made significant improvements to the software, focusing on enhancing its quality. This includes refactoring, increasing code coverage, fixing bugs, adding support for new compilers and improving the user and developer documentation. Additionally, we have consolidated and simplified our CI/CD processes, resulting in better management and reduced total costs of ownership.

Key improvements in SGA3: For upcoming NEURON 9.0, we have performed a thorough refactoring of the data structures for the NEURON model, in order to move from Array-of-Structures (AoS) to Structure-of-Arrays (SoA), resulting in better CPU performance. This refactoring also sets the stage for tighter cohesion between NEURON and CoreNEURON. We have also undertaken a major refactoring of the Python extensions handling through setup tools, streamlining the build complexity (CMake and wheels) and eliminating technical debt. Furthermore, CoreNEURON now utilizes our modern NMODL framework, while mod2C has been discontinued.

Outlook: Moving forward, we aim to achieve even tighter integration between NEURON and CoreNEURON, leading to increased performance and optimization. This involves introducing common SOA data structures and implementing modern solutions in some NEURON code from the CoreNEURON implementations. Moreover, we are adopting MorphIO for morphology loading, which promises a significant speed increase in simulations.

5.2.13 Neuromorphic Computing Job Manager App

Type: web app, command-line app and library
Scope: networks

Abstraction level: point neurons
Integration Status: available as a Collaboratory app, supports login via EBRAINS credentials, reachable under a custom domain, EBRAINS software stack (since EBRAINS-22.07)
EBRAINS since 22.07
Homepage & more information:

- https://electronicvisions.github.io/hbp-sp9-guidebook/using_the_platform.html
- https://wiki.ebrains.eu/bin/view/Apps/?text=Job+Manager&category=Neuromorphic

Recent Changes (M22-M42): The web app, command-line app and Python client have all been updated to work with version 3 of the neuromorphic computing remote access API, which notably adds support for tracking quota usage of interactive sessions with SpiNNaker and BrainScaleS.

In the web app, it is now possible to select the model/simulation code from the Collaboratory Drive and from the Data Proxy service ("Buckets"), and to copy simulation outputs to the same locations with a single click. It is also possible to export provenance metadata to the EBRAINS Knowledge Graph, facilitating workflows that combine HPC and neuromorphic computing.

Key improvements in SGA3: The neuromorphic computing job queue system has been fully migrated to Collaboratory v2, including using the latest version of the EBRAINS authentication system, reading and writing from Collaboratory storage, and availability both as a stand-alone web app and as a Collaboratory app. The web app was entirely rewritten using a more modern Javascript framework, to improve maintainability and performance.

Outlook: The neuromorphic computing remote access system will be maintained as needed to support neuromorphic computing services in EBRAINS. Integration into the common EBRAINS workflow system as a CWL component is underway.

5.2.14 *Multiscale Co-Simulation Framework*

Type: command-line tool or library Scope: networks, whole brain	
Abstraction level: point models, biophysical models, population models Integration Status: depends on use case Homepage & more information: • https://github.com/multiscale-cosim	

Recent Changes (M22-M42): 1. Ongoing development of communication and deployment infrastructure for launching and coordinating complex multi-scale simulation workloads. 2. the development of tooling for monitoring of basic simulator state and resource utilization both on HPC and virtualized hardware. 3. The creation of a template-based model formalization and deployment configuration to support reproducibility. 4. Integration of the co-simulation use-cases: a. TVB-NEST two-way coupling, b. NEST-Arbor co-simulation, c. Coupling with insitu visualization infrastructure, and d. LFPy NEST coupling. e. Neurorobotics.

Key improvements in SGA3: The framework is fully developed in SGA3 and enables configuration, launching and monitoring of complex multi-scale workloads on HPC systems and local laptops. The tooling included full environment build scripts and example use-cases integrating NEST, TVB, Arbor, Elephant, the Insite pipeline, NEST Desktop and Neurorobotics use-cases.

Outlook: Support and development will be reduced in scale pending acquisition of additional funding.

5.2.15 *The Virtual Brain (TVB)*

See section 5.1.13.

5.2.16 *MMCG Platform*

Type: web app Scope: networks	
Abstraction level: nanoscale Integration Status: open source, reachable under custom domain (iframe in EBRAINS wiki) Homepage & more information: • https://mmcg.grs.kfa-juelich.de/ • Documentation: https:/mmcg.grs.kfa-juelich.de/documentation • https://search.kg.ebrains.eu/instances/5f66611e-49c0-489f-840b-a991f051f505	

Recent Changes (M22-M42): Improved documentation to make it accessible to non-experts.

Key improvements in SGA3: The web server is online, working and already available via the EBRAINS Collaboratory[37].

Outlook: The tool will be integrated within the SSB platform.

5.3 Data Analysis

5.3.1 Neo

Type: command-line tool or library Scope: all	
Abstraction level: all Integration Status: open source, EBRAINS software stack (since 21.09) Homepage & more information: • http://neuralensemble.org/neo/ • https://neo.readthedocs.io/ • https://kg.ebrains.eu/search/instances/Software/b0d73040-6547-447d-a1be-6761c209973f	EBRAINS since 23.02

Recent Changes (M22-M42): Improvements in Neo since v0.10.0 mostly fall into two categories: (1) improvements to file format support (bug fixes, performance improvements, better handling of edge cases), adding IO handlers for additional file formats, and improved file format detection; (2) quality improvements: an update of tooling for building and installing Neo to follow the latest recommendations of the Python community; major improvements to the IO test suite, with a substantial speed-up; internal code changes to improve maintainability and facilitate contributions from new developers.

Also in the category of quality improvements, the documentation has been completely rewritten and refreshed, including a more modern appearance, and provision of multiple entry points for different types of users: people wishing to analyse their own data; people looking to share data in an open format; people using Neo as a component in their own library or application; and people contributing to Neo development, for example by adding support for a new file format.

There have been 5 releases of Neo in the time period covered by this report: v0.10.1, v0.10.2, v0.11.0, v0.12.0 and v0.13.0.

Key improvements in SGA3: The main achievements in SGA3 have been: Refinement and stabilisation of the API, making it more intuitive for new users; Continued rapid expansion of file format support: Neo now supports 52 file formats compared to 45 in January 2022 and 40 at the start of SGA3; Greatly improved documentation; Internal changes to simplify the code base, making it more maintainable and easier for new developers to contribute; a larger, more robust and more performant test suite.

Outlook: The release of Neo v1.0 is planned for late 2023. After this, the development team will commit to maintaining backwards compatibility for all v1.x releases. We will continue to add support for new file formats and will focus on improving support for fluorescence imaging data ("optophysiology").

---

[37] https://wiki.ebrains.eu/bin/view/Collabs/hybrid-mmcg-webserver/Webserver

5.3.2 Elephant

Type: command-line tool or library
Scope: single neurons, networks, whole brain

Abstraction level: point neuron models, biophysical models, population models, statistical models
Integration status: open source, EBRAINS software stack (since 22.10) EBRAINS since 22.10
Homepage & more information:

- https://python-elephant.org
- Documentation: https://elephant.readthedocs.io
- https://kg.ebrains.eu/search/instances/Software/f3685e37-cb34-4010-a528-468564482257

Recent Changes (M22-M42): The Electrophysiology Analysis Toolkit Elephant was further developed receiving releases[38] 0.11.0, 0.12.0, 0.13.0 and a first major release 1.0.0. These releases featured several new analysis capabilities that were added to Elephant (e.g., spectral methods, causality estimators, and phase estimation). In terms of the library design, a new data model was introduced to represent the concept of a trial in electrophysiology, i.e., repetitions of a task protocol. This abstraction improves the usability of the library by accepting trial data formatted in multiple ways. In addition, the rewrite of the documentation was finalized with improved tutorials that are now directly executable on the Collaboratory infrastructure. The consistency of documentation profits from the inclusion of documentation tests. Additional validation tests against published results were added and made publicly available. Provenance tracking allows scientists to track the analysis performed throughout a project and can now be implemented in Elephant by integrating the Alpaca[39] tool.

Key improvements in SGA3: Elephant has matured into a stable platform to make available analysis methods for spiking data and mesoscopic population activity. This includes performance improvements, an inviting and defined online documentation, and a full interoperability with the Neo data model. The newly added ability to automate the tracking of provenance in Elephant forms the prerequisite to share electrophysiology analysis results in a searchable, self-documenting manner via the EBRAINS search.

Outlook: Development of Elephant will be continued and will receive contributions as part of ongoing scientific projects. Certain aspects of Elephant, such as development of additional data models and ontologies, have been proposed for funding as parts of national and international grant proposals.

5.3.3 LFPy

Type: command-line tool or library
Scope: single neurons, networks

Abstraction level: biophysical neuron models, point neuron models, population models
Integration Status: open source, EBRAINS software stack (since 22.07) EBRAINS since 22.07
Homepage & more information:

- Documentation LFPy: https://lfpy.readthedocs.io/
- Documentation LFPykit : https://lfpykit.readthedocs.io/
- https://search.kg.ebrains.eu/instances/2b3dba3b-7398-4cd8-bc30-1245ae7caf4f

Recent Changes (M22-M42): The integration between LFPy and Arbor has been improved, and is now documented with examples as part of the documentation of both Arbor and LFPykit. Further, a new version of LFPykernels has been released[40], to enable LFPykernels to work with a broader range

---

[38] https://elephant.readthedocs.io/en/latest/release_notes.html
[39] https://alpaca-prov.readthedocs.io/en/latest/
[40] https://github.com/LFPy/LFPykernels/releases/tag/v0.2.0

of point-neuron network models, like the Potjans-Diesmann network model. This framework enables calculation of brain signals like LFPs and EEGs from point neuron network models or firing rate models.

Key improvements in SGA3: The kernels approach for calculating brain signals like LFPs/EEGs/MEGs directly from spikes or firing rates was developed and found to be both accurate and efficient which makes this a promising approach for improved brain signal predictions from NEST and/or TVB. The results have been published in 2022[41].

Outlook: We will finish the integration with the multi-scale co-simulation framework, where LFP/EEG signals are calculated in real time, based on spikes that are streamed from a NEST simulation (with the Potjans-Diesmann model as a use case). We are planning to use the Insite pipeline for visualisation.

5.3.4 The Virtual Brain (TVB)

See section 5.1.13.

5.4 Visualisation

5.4.1 SimVisSuite

Type: desktop app
Scope: single neurons, networks

Abstraction level: point neuron models, biophysical models
Integration Status: open source
Homepage & more information:
Visimpl:
- https://vg-lab.es/visimpl/
- https://kg.ebrains.eu/search/instances/Software/6f8b4c7d-31f6-49b5-a843-2e0b4421be6f

NeuroTessMesh:
- https://github.com/vg-lab/NeuroTessMesh
- https://kg.ebrains.eu/search/instances/Software/4aa7fc48-101c-4f8f-bd6e-559486c133a3

Neuroscheme:
- https://github.com/vg-lab/NeuroScheme
- https://kg.ebrains.eu/search/instances/Software/a5f69883-eb34-4dd5-bc44-30e5d4718f27

Recent Changes (M22-M42): SimVisSuite tools have been further developing until ViSimpl release 1.8.8, NeuroScheme release 0.7.5 and NeuroTessMesh release 0.4.3. These releases featured different new capabilities, such as a video recorder; the resolution of technical requirements for the deployment of the tools, the use of an updated particle system and the adaptation needed to work with the Insite, apart from continuous integration. Consistent documentation has been also continuously updated. Additionally, ViSimpl has been integrated with NEST Desktop, and can also be used with the Insite.

Key improvements in SGA3: Key improvements of SimVisSuite are the changes in the GUI, the addition of a videorecorder and an updated particle system for the visualization of the spikes. ViSimpl has been integrated with NEST Desktop and can also be used with the Insite. Documentation has been extended and improved.

---

[41] Hagen et al. (2022) Brain signal predictions from multi-scale networks using a linearized framework. PloS Computational Biology. doi: https://doi.org/10.1371/journal.pcbi.1010353 (P3229)

Outlook: SimVisSuite tools are planned to keep evolving and will receive different developments, such as a new graphical engine that improves performance. The tools have been proposed for funding as part of national and international grant proposals.

5.4.2 *NEST Desktop*

Type: web app Scope: networks	
Abstraction level: point neuron models Integration Status: reachable under ebrains.eu domain, supports login via EBRAINS credentials Homepage & more information: • https://nest-desktop.apps.hbp.eu/#/ • Documentation: https://nest-desktop.readthedocs.io • https://github.com/nest-desktop/nest-desktop • https://search.kg.ebrains.eu/instances/0d964b14-92f9-47d0-9841-888ac4e32ef4	

Recent Changes (M22-M42): Release 3.1.0 introduced a reworked side navigation, NEST 3.1 support, code highlighting for the pyNEST script code, improved visual representation for UI elements and new UI functions, an import/export function for models on GitHub, a powerful model manager together with new model types and analysis methods as well as first workflow automatizations (CI integration for automatic deployment with first preparations for automatic tests) for an efficient workflow that is as error-free as possible. Release 3.2.0 introduced a new system bar providing an intuitive overview of the running backends together with usability/workflow improvements, a new model manager, the usage of a code templating engine for better maintainability and extendibility, compartmental neuron models, weight recorders for synapses, a significantly improved documentation integration as well as new export options for network graphs and internal preparations for the usage together with external tools like Neurorobotics Platform (NRP) and CoSim. Furthermore, NEST Desktop has been integrated with ViSimpl. The work also includes an intuitive user documentation, which is contained in the documentation pages of both tools[42]. The feature releases mentioned above were accompanied by four patch releases[43] which improved the stability and fixed some bugs. Finally, NEST Desktop has been formally tested for good usability.

Key improvements in SGA3: Major improvements of NEST Desktop in SGA3 can be summarized due to significant extension of the functionality, the introduction of a comprehensive software architecture and an agile development process. Furthermore, NEST Desktop has been widely integrated in various tools and use cases such as the NRP, CoSim as well as a use integration ViSimpl, Insite and NEST Desktop. Finally, documentation has been significantly extended and improved.

Outlook: It is planned to continue development of NEST Desktop as educational tool and visual interface for NEST. It is planned to broaden the scope towards scientific use cases and application scenarios as well as extend the number of support simulators.

5.4.3 *SimDaVis - Insite*

Type: command-line tool or library Scope: single neurons, networks, whole brain	
Abstraction level: point neuron models, biophysical models, population models, statistical models Integration Status: open source Homepage & more information:	

---

[42] https://nest-desktop.readthedocs.io/en/latest/user/usage-external/usage-with-visimpl.html

[43] https://nest-desktop.readthedocs.io/en/latest/

- https://vrgrouprwth.github.io/insite
- https://github.com/VRGroupRWTH/insite
- https://kg.ebrains.eu/search/instances/Software/eb0b86a5-eb49-4a87-9128-6ca592293218

Recent Changes (M22-M42): The existing NEST API was improved and updated to be more stable. Additionally, new fields indicating important simulator properties were added to make the API easier to use. To improve the performance of Insite, the software was reimplemented in C++, which allows to more efficiently process the data. The simulator support was supplemented by adding APIs for Arbor and TVB, thus, allowing to receive data while simulating from multiple scales at the same time. To show this a reference implementation together with the 25Multiscale Co-Simulation Framework was developed. For all three simulators, tests were developed to ensure the correct functioning. The user documentation was extended to reflect functionality for all three simulators. New Docker containers were created and are available via the EBRAINS container registry for easy deployment of Insite-enabled NEST, Arbor and TVB simulators. Example files are provided which allow to run all three simulators and the access node via docker-compose, an easy-to-use container scripting application. Insite was integrated into an end-to-end workflow with ViSimpl and NEST Desktop to allow network design via NEST Desktop and in-transit visualization with ViSimpl via Insite.

Key improvements in SGA3: The Insite pipeline was completely redesigned to be easy to use and easy to integrate for developers of other tools. This included new and more efficient implementations, simplified APIs and better documentation. This resulted in a pipeline that let users receive in-transit data from NEST, Arbor and TVB, thus, representing multi-scale functionality from the level of individual neurons to the simulation of whole brain regions and is integrated by multiple tools.

Outlook: There is currently no further active development planned.

5.4.4 *Elephant Visualization (Viziphant)*

Type: command-line tool or library
Scope: single neurons, networks, whole brain

Abstraction level: point neuron models, biophysical models, population models, statistical models
Integration Status: EBRAINS Software Stack (since EBRAINS-22.07) EBRAINS since 22.07
Homepage & more information:

- https://github.com/INM-6/viziphant
- https://search.kg.ebrains.eu/instances/4c2fc8b4-8e39-4e6d-ab46-9a979183b60e
- Documentation: https://viziphant.readthedocs.io

Recent Changes (M22-M42): Viziphant has received updates[44] to versions 0.2.0 and 0.3.0 that features continuous updates and API improvements to various functions that visualize analysis results obtained by the Elephant toolkit.

Key improvements in SGA3: Viziphant was continuously developed to better match the data types and return types of the Elephant analysis library.

Outlook: Beside the continuous development of Viziphant as a companion library to Elephant, is currently used as a basis for an experimental Jupyter Lab extension that will visualize analysis results online in the spirit of a graphical variable inspector.

---

[44] https://viziphant.readthedocs.io/en/latest/release_notes.html

5.5 Validating Models Against Experimental Data

5.5.1 Validation Test Libraries

Type: library
Scope: all

Abstraction level: all
Integration Status: open source, EBRAINS stack (since 22.07, HippoUnit)
Homepage & more information:
NetworkUnit:
- https://github.com/INM-6/NetworkUnit
- https://kg.ebrains.eu/search/instances/Software/e3566d6b-fed9-4f5d-b704-04e6eda76c99

MorphoUnit:
- https://github.com/appukuttan-shailesh/morphounit
- https://kg.ebrains.eu/search/instances/Software/860c230e-56cf-4f49-bdb7-cd438a3de0e2

HippoUnit: EBRAINS since 22.07
- https://github.com/KaliLab/hippounit
- https://kg.ebrains.eu/search/instances/Software/fd369ab8-c788-4c7b-89ec-594aa3e46c24

BasalUnit:
- https://github.com/appukuttan-shailesh/basalunit
- https://search.kg.ebrains.eu/instances/46004f38-8d2b-4c68-81ba-8a85a2271eeb

CerebTests:
- https://cerebtests.readthedocs.io/
- https://search.kg.ebrains.eu/instances/9bc9f88a-8bdd-4bf3-b134-6e0ce2dc7e6b

eFELUnit:
- https://github.com/appukuttan-shailesh/eFELunit
- https://search.kg.ebrains.eu/instances/426af237-1f35-4c24-a8a3-88c454682d9a

Validation Framework - Python Client: EBRAINS since 23.06
- https://hbp-validation-client.readthedocs.io/en/master/#
- https://kg.ebrains.eu/search/instances/Software/c053931f-ddd1-4416-8b3e-5506677c4bda

Recent Changes (M22-M42): Several of the validation tests libraries have received incremental updates. In detail, 1) HippoUnit has had one new release during this time period, adding a pathway interaction test, 2) NetworkUnit has been released as version 0.2 featuring test parallelisation and an extended repertoire of statistical tests, 3) BasalUnit has had three releases including new functionality, in particular support for Snudda, 4) CerebTests has had its first release, including the additional of measures of sensitivity, specificity and accuracy of the validation tests, and 5) eFELUnit has had four releases, including new features and bug fixes.

The Python client for the validation framework has been updated in line with changes to the model validation REST API (see entry on Model Catalog, below) with seven releases in the time period.

Key improvements in SGA3: New collaborations with modelling groups have been developed to push forward development of certain libraries, in particular with the Snudda developers in the case of BasalUnit, building on the successful example of HippoUnit, for which a study demonstrating its functionality and scientific value was published in Sáray et al., 2021[45]. NetworkUnit has received new capabilities for parallelisation of tests, as well as an extended repertoire of statistical tests.

---

[45] Sáray et al. (2021) HippoUnit: A software tool for the automated testing and systematic comparison of detailed models of hippocampal neurons based on electrophysiological data. PLoS Computational Biology. DOI: https://doi.org/10.1371/journal.pcbi.1008114 (P2993)

Outlook: The remaining libraries are in the process of integration into the EBRAINS software stack. An active collaboration is underway on basal ganglia model validation, and additional test libraries for network models of hippocampus and for whole brain models are under development.

5.5.2 *Model Catalog*

Type: web app, REST API Scope: all	💻 ⚙ ✔ ⁙ ⇄ ⚗ ⅄ 🧠 🎓 👥 🧠
Abstraction level: all Integration Status: reachable under a custom domain, supports login via EBRAINS credentials, available as a Collaboratory app Homepage & more information: • https://model-catalog.brainsimulation.eu/docs/ • https://validation.brainsimulation.eu/docs • https://search.kg.ebrains.eu/instances/de2f6456-46a3-40f0-8a51-93833d00b729	

Recent Changes (M22-M42): The Model Validation Web Service, providing a model-centric view of the Knowledge Graph focused on visualisation of model validation results, has been refined and improved. The Model Catalog app has been maintained and polished, with improved unit tests. The REST API has been extensively re-written to adapt to the release of Knowledge Graph version 3, and the associated migration of model metadata to the openMINDS schemas. The deployment pipeline has been streamlined, making it easier and faster to deploy both bug fixes and new features.

Key improvements in SGA3: The Model Catalog is now available as a “community app” in the EBRAINS Collaboratory (as well as being a standalone web app), which allows the app’s configuration (filtering by brain region, cell type, model scope, etc.) to be saved on a per-Collab basis. The landing page of the app now highlights featured models and shows statistics about the models. The Catalog app and API are now fully driven by the Knowledge Graph (v3) using openMINDS schemas.

Outlook: We are planning to improve the usability of the Model Catalog, adding the results of simulations that have been performed of each model, and highlighting models that have been validated against experimental data and/or that have been demonstrated to be reproducible.

Annex - Summary Tables of EBRAINS Tools, Workflows and Services

Building Models

Table 3: Tools and services part of EBRAINS for building models, across the diverse scales.

Tool or service	Characteristics	Modelling abstraction level	Purpose
Molecular Level Toolset	⚙ 💻 🔧 ▦ 🎓 👥 🧠	NS	Set-up, run and analyse molecular level calculations, e.g., tauRAMD calculations to compute dissociation constants.
CNS Ligands	💻 ✔ ▦ 🧠	NS	Efficiently generate and parameterise bioactive conformers of ligands binding to neuronal proteins for molecular dynamics simulations.
GRALL	💻 ✔ ▦ 🧠	NS	Search for allosteric modulators of the human glycine receptor with a structural annotation of their ligand-binding site.
Subcellular Web App	💻 ✔ ⇄ 👥 🧠	SB	Create and simulate brain molecular networks.
Subcellular Model Building Toolset	⚙ 🔧 ⇄ 🧠	SB	Construct, simulate and validate subcellular models.
Feature Extraction Tool (NeuroFeatureExtract)	💻 ✔ ⧗ 👥 🧠	BI	Extract a wide variety of electrophysiological features from experimental recordings.
Hodgkin-Huxley Neuron Builder (HHNB)	💻 ✔ ⧗ 👥 🧠	BI	Optimise a single cell model against experimental data on HPC resources.
Electrical Model Building Toolset	⚙ ✔ ⧗ 👥 🎓 🧠	BI	Build and validate biophysically detailed electrical neuron models.
PyNN	⚙ ✔ ⅄ 🧠 🎓 👥	PT BI	Build simulator-independent spiking neuronal networks and simulation experiments.
NESTML	⚙ ✔ ⧗ ⅄ 🎓 👥 🧠	PT	Build simulator-independent point neuron and synaptic plasticity models.
Snudda	⚙ 🔧 ⅄ 👥 🧠	BI	Build networks of biophysically detailed neurons where the connectivity is based on the neuron morphologies.
Brain Scaffold Builder	⚙ ✔ ⧗ ⅄ 👥 🧠	PT, BI	Rapidly construct biologically realistic network models.
The Virtual Brain (TVB)	💻 ⚙ ✔ 💡 🎓 👥 🧠	PP	Construct and simulate personalised brain network models.
Single Cell Modelling Workflows	⚙ ✔ ⧗ 👥 🧠	BI	Build biologically detailed single cell models.

Running Simulations

Table 4: Tools and services part of EBRAINS for running simulations, for the diverse scales.

Tool or service	Characteristics	Modelling abstraction level	Purpose
CGMD Platform	💻✔⅄ 👤	NS	Prepare, run and analyse coarse-grained molecular dynamics simulations.
SSB Platform	💻📋⇄ 👥👤	NS SB	Predict the downstream effects upon ligand binding/disease associated variants, such as protein/metabolite concentrations/dose response curves, and explore metabolic networks using automated bio-mathematical models.
MoDEL-CNS	💻🔧⇄▦ 🎓👥👤	NS	Access atomistic-MD trajectories for relevant signal transduction proteins in a web browser.
BioExcel-CV19	💻⚙🔧⇄▦ 🎓👥👤	NS	Web server resource for providing molecular dynamics simulation data. Currently, set-up to provide simulation trajectories for Covid-19 studies but is also applicable to brain-related (and other) studies.
BioExcel Building Blocks Workflows	💻⚙🔧⇄▦ 🎓👥👤	NS	Workflows to prepare and run atomistic Molecular Dynamics simulations to signal transduction and Central Nervous System related proteins.
SDA	💻✔▦ 🎓👥👤	NS	Simulate diffusional association of proteins. Compute bimolecular association rate constants.
Single Cell *In Silico* Experiment Tool (BlueNaaS)	💻✔⧗ 🎓👥👤	BI	Neuron as a Service: Simulate and visualise detailed single neuron models in a web browser.
Small Circuit *In Silico* Experiments Tool (Rat Hippocampus CA1)	💻✔⅄ 🎓👥👤	BI	Run simulation experiments with a small hippocampal neural circuit.
Brain Areas Circuit *In Silico* Experiments Tool (Rat Hippocampus CA1)	💻✔⅄ 🎓👤	BI	Run simulation experiments with a full-scale model of the hippocampus CA1 region.
NEST	⚙✔⅄ 🎓👥👤	PT	Run highly scalable simulations of point neuron networks.
Arbor	⚙✔⧗⅄ 🎓👥👤	BI	Run scalable, high-performance simulations of multi-compartment neurons in large networks.
NEURON/CoreNEURON	⚙✔⧗⅄ 🎓👥👤	BI	Run scalable, high-performance simulations of multi-compartment neurons in large networks in the well-established NEURON simulation environment.

Neuromorphic Computing Job Manager App	💻 ⚙ ✔ ⅄ 🎓 👥 👤	PT	Run simulation experiments with point neuron networks on the BrainScaleS and SpiNNaker neuromorphic computing systems.
Multiscale Co-Simulation Framework	⚙ 🔧 ⅄ 💡 👤	PT BI PP	Connect simulation tools at runtime to enable multiscale co-simulations.
The Virtual Brain (TVB)	💻 ⚙ ✔ 💡 🎓 👥 👤	PP	Simulate personalised brain network models locally, on the EBRAINS Collaboratory, or directly on HPC for large parameter explorations.
MMCG Platform	💻 ✔ ⅄ 👤	NS	Set-up hybrid molecular mechanics/coarse grain simulations.

Data Analysis

Table 5: Tools and services part of EBRAINS for data analysis, for the diverse scales.

Tool or service	Characteristics	Modelling abstraction level	Purpose
Neo	⚙ ✔ ▦ ⇄ ⌛ ⅄ 💡 🎓 👥 👤	SB PT BI PP ST	Load, save, annotate and manipulate electrophysiology and imaging data from experiment and simulation in a standardised framework.
Elephant	⚙ ✔ ⌛ ⅄ 💡 🎓 👥 👤	PT BI PP ST	Analyse neuronal activity data from experiment and simulation using high-performance, well-tested methods.
LFPy	⚙ ✔ ⌛ ⅄ 👥 👤	BI PT PP	Calculate electric and magnetic brain signals from multi-compartment neuron models and networks.
The Virtual Brain (TVB)	💻 ⚙ ✔ 💡 🎓 👥 👤	PP	Load, analyse, visualise and explore experimental and simulated brain activity time series.

Visualisation

Table 6: Tools and services part of EBRAINS for visualisation, for the diverse scales.

Tool or service	Characteristics	Modelling abstraction level	Purpose
SimVisSuite	💻 🔧 ⌛ ⅄ 👥 👤	PT BI	Interactively visualise and analyse network and neuron-level simulations.
NEST Desktop	💻 ✔ ⅄ 👥 👤	PT	Build, run, visualise and analyse simulations with the NEST simulator in a web-based GUI.
SimDaVis - Insite	⚙ ✔ ⌛ ⅄ 💡 👤	PT BI PP ST	Continuously access simulation data at runtime.
Elephant Visualization (Viziphant)	⚙ 📋 ⌛ ⅄ 🎓 👥 👤	PT BI PP ST	Visualise analysis results of the Elephant library, e.g., in Jupyter notebooks.

Validating Models Against Experimental Data

Table 7: Tools and services part of EBRAINS for validating models, for the diverse scales.

Tool or service	Characteristics	Modelling abstraction level	Purpose
Validation Test Libraries	⚙ ✔ 🔧 ▦ ⇄ ⚗ ⅄ 💡 ☁🎓 👥 🎧	SB PT BI PP ST	Develop and run structured, quantitative model validation tests.
Model Catalog	💻 ⚙ ✔ ▦ ⇄ ⚗ ⅄ 💡 ☁🎓 👥 🎧	SB PT BI PP ST	Explore and visualise computational models, validation tests and validation results.