Grant Agreement:	604102	Project Title:	Human Brain Project

Document Title:	Brain Simulation Platform v1
Document Filename:	SP6 D6.7.4 FINAL.docx
Deliverable Number:	D6.7.4
Deliverable Type:	Prototype
Work Package(s):	WPs 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7 (WPs involved in writing this document)
Dissemination Level:	PU
Planned Delivery Date:	M30 / 31 Mar 2016
Actual Delivery Date:	M30 / 31 Mar 2016

Authors:	Felix SCHÜRMANN, EPFL (P1), Task No. 6.1.5, 6.2.1, 6.2.2, 6.2.4, 6.5.1, 6.6.1, 6.7.1 Jean-Denis COURCOL, EPFL (P1), Task No. 6.5.1
Compiling Editors:	Katrien VAN LOOK, EPFL (P1), Task No. 6.7.1
Contributors:	Henry MARKRAM, EPFL (P1), Task No. 6.1.1, 6.1.2, 6.1.3 Jeanette HELLGREN-KOTALESKI, KTH (P33), Task No. 6.4.1 Julian SHILLCOCK, Juan PALACIOS, Liesbeth VANHERPE, EPFL (P1), Task No. 6.1.1 Idan SEGEV, Guy EYAL, HUJI (P21), Task No. 6.1.3 Marc-Oliver GEWALTIG, Csaba EROE, EPFL (P1), Task No. 6.1.4 Fabien DELALONDRE, EPFL (P1), Task No. 6.1.5, 6.2.1, 6.2.2, 6.2.4 Erik DE SCHUTTER, OIST (P38) Task No. 6.2.1 Michael HINES, YALE (P80) Task No. 6.2.2 Markus DIESMANN, JUELICH (P17), Task No. 6.2.3 Richard LAVERY, Elisa FREZZA, Juliette MARTIN, CNRS (P7), Task No. 6.3.1 Paolo CARLONI, Pietro VIDOSSICH, JUELICH (P17), Task No. 6.3.2 Rebecca WADE, Neil BRUCE, HITS (P24), Task No. 6.3.3 Antoine TRILLER, Leandro ALMEIDA, ENS (P13), Task No. 6.4.2 Pierre MAGISTRETTI, Daniel KELLER, EPFL (P1), Task No. 6.4.3 Alex THOMSON, Audrey MERCER, UCL (P71), Task No. 6.4.4 Eilif MULLER, Armando ROMANI, EPFL (P1), Task No. 6.4.4 Egidio D'ANGELO, UNIPV (P61), Task No. 6.4.5 Sten GRILLNER, Alexander KOZLOV, KI (P31), Task No. 6.4.6 James DYNES, Michael GEVAERT, Luis RIQUELME, Werner VAN GEIT, EPFL (P1), Task No. 6.5.1
Coordinator Review:	EPFL (P1): Jeff MULLER, Martin TELEFONT UHEI (P45): Sabine SCHNEIDER, Martina SCHMALHOLZ

Editorial Review:	EPFL (P1): Guy WILLIS, Lauren ORWIN

Abstract:	The Deliverable accompanies the public release of the Brain Simulation Platform v1 prototype. This document lists the products, software packages and services that are available in this first public release, along with the status of the use cases that were described in the specification document (D6.7.1). There is also information on new features that will be added in future releases, and how to provide user feedback, which is important for the development of the Platform.
Keywords:	Prototype, Platform release, Brain Simulation Platform
Available at:	www.humanbrainproject.eu/ec-deliverables

Table of Contents

List of Figures and Tables

1. The Aim of this Document

This document provides access to the Brain Simulation Platform (BSP) v1 and related information.

2. How to Access the Brain Simulation Platform

The Brain Simulation Platform is one of six ITC Platforms that comprise the HBP Scientific Research Infrastructure. All these Platforms can be accessed via the HBP Collaboratory web interface:

https://collab.humanbrainproject.eu/#/collab/19/nav/403

Direct link to the Brain Simulation Platform on the Collaboratory:

https://collab.humanbrainproject.eu/#/collab/161/nav/1389

3. Platform User Instructions

The Platform documentation constitutes a separate Deliverable (D6.7.5 – Brain Simulation Platform v1 — Documentation), which will include direct links to Technical and User Documentation, such as the Brain Simulation Platform Guidebook. The Guidebook can be found in the Brain Simulation Platform Collaboratory: https://collab.humanbrainproject.eu/#/collab/161/nav/5576

D6.7.5 was also scheduled to include a roadmap describing plans for future Platform development, but this topic is covered in this document - see Annex F: Backlog (remaining bugs and new features to be added).

4. Platform Testing and Quality Strategy

Multiple layers of controls are applied to the Platform to ensure that its quality is maintained.

4.1 Continuous Integration System

Each piece of software delivered on the Platform is managed by a source code management system. Upon each tentative modification of the software, the following workflow is triggered:

- The existing unit test bed is run with the tentative modifications
- A set of code sanity and syntax checks is run
- A code coverage check is computed and cannot be below a certain threshold (90%); for legacy code not reaching this coverage level, should at least increase
- A code review request is triggered to a set of developers.

The outcome of the four checks mentioned should be positive for the modification to be applied to the software.

The modification of the source code does not mean that the code is deployed to the production system. The production manager or a representative responsible for the software is required to trigger the deployment manually.

4.2 Functional Tests

Functional tests are run during a version change of the software to assess the quality of the version being built.

The modification of the source code or the release of a new version does not mean that the code is deployed to the production system. The production manager or a representative responsible for the software is required to trigger the deployment manually.

4.3 System Tests

A set of tests that check high-level use cases involving several pieces of the architecture are executed twice a day.

Upon failure, the developers are notified with a link to the log of the failing tests.

4.4 Monitoring

The BSP services use the Collaboratory platform services, such as the Task Service or Document Service. As such, they benefit from the Collaboratory default monitoring capabilities.

The Collaboratory web servers send their logs to a pool of indexers that extract metadata such as hostname, service, etc. and index them in an Elasticsearch cluster.

Building on this cluster, a web interface named Kibana enables interactive log exploration and dashboard generation.

5. Platform User Adoption Strategy

Annex H describes the BSP platform user adoption strategy and steps taken to implement it.

6. Help and User Feedback

To obtain help in using the Platform, please start by checking the online user documentation here: https://collab.humanbrainproject.eu/#/collab/161/nav/5991

If you are interested in the online developer documentation, you can find it here: https://developer.humanbrainproject.eu/docs/

If you need personal assistance, want to provide feedback or contribute to the on-going development of the Platform, please contact: bsp-support@humanbrainproject.eu

A forum is also available where users can have access to known problem resolutions, get assistance from the developers and the community, and share knowledge: https://forum.humanbrainproject.eu

6.1 User Feedback Received in Month 18-Month 30

The main feedback was collected from Co-Design users and early external adopters:

1) SP6 internal (EPFL/BBP, HUJI, UNIPV, KI, KTH) and HBP external (Allen Institute for Brain Science) users' feedback on the single cell modelling workflow

2) SP6 internal (EPFL/BBP, UNIPV) and HBP external (Hippocampus community) users' feedback on the brain building workflow
3) Community feedback on the simulator engines.

Adopter 1): The single cell modelling workflow comprises various steps and was originally developed for cortical neurons by Partners EPFL/BBP and HUJI. It involves the curation of morphologies, extraction and definition of electrical features, search for optimal parameters, model selection and inspection. When exercising this for neurons from different brain regions, the main feedback was:

- **Flexibility of overall workflow:** sometimes morphologies of good quality are readily available; sometimes they require substantial curation to prep them for electrical simulations. The implication of this is that sometimes certain workflow steps can be skipped and sometimes not, or sometimes even additional steps are required.
  Consequently, it was decided to:
  - Factor out the morphology curation workflow; this has been implemented in use case SP6BSP-UC-007
  - Provide the morphology curation workflow and electrical model building workflow in the form of ipython notebooks; this has been implemented in use case SP6BSP-UC-009.
- **Flexibility of specific workflow steps:** when building neuron models for other brain regions it is sometimes necessary to define novel electrical features (e.g. complex dendritic features) or to constrain the parameters with additional knowledge (e.g. make certain parameters dependent on each other). Consequently, it was decided to:
  - Open-source critical components, such as the eFEL library; the eFEL library has been open-sourced on Github: https://github.com/BlueBrain/eFEL.
  - Develop a more flexible version of the original OptimizerFramework in Python, allowing the user to modify it more easily; this has been developed and BluePyOpt has been open-sourced on Github: https://github.com/BlueBrain/BluePyOpt.

Adopter 2): The brain building workflow comprises various steps and was originally developed for cortical microcircuits by Partner EPFL/BBP. It involves the definition of the volume and cell positions, the assignment of cell types to positions, the selection and placement of appropriate cell morphologies, structural proximity identification and conversion to functional synapses. When exercising this for different brain regions, the main feedback was:

- **Flexibility of overall workflow:** sometimes certain data are not available and parts of the workflow have to be implemented differently or region specific steps have to be introduced. The implication of this is that a more modular building workflow is necessary, making it possible for the users to reuse portions, but overwrite/extend other portions they may require. Consequently, it was decided to:
  - Provide a new brain building workflow framework in Python, which allows users to use the existing components (e.g. touch detector), but also to change certain steps easily while still maintaining the overall structure. This development has been started (see use case SP6BSP-UC-012) and is being tested with the various Co-Design users (especially Hippocampus, Cerebellum); additional work is required.

- **Integration with the Neuroinformatics Platform:** a set of APIs is provided to interoperate with the data format produced by the Neuroinformatics Platform. Every step of the pipeline works with a common volumetric representation of the circuit properties. This implies that every step can use either the Neuroinformatics Platform data (through the provided API) or custom data provided by the user, as long as this can be converted into the common representation (see SP6BSP-UC-012).

Adopter 3): As the three main simulators adopted by SP6 (NEST, NEURON, STEPS) are open-source, there is a wider community out there. In order to have this community profit from the HBP efforts (independently of the BSP), all extensions by SP6 Partners will be made open-source:

- NEST—all extensions made during the Ramp-Up Phase are available in the open-source version 2.10.0 of NEST in the regular NEST repository
- NEURON—the coreNeuron developments and appropriate hooks for its use with NEURON are open-sourced in Month 30 in the regular NEURON repository
- STEPS—the path-finding work for the parallelisation of STEPS will be open-sourced in the regular STEPS repository once the publication has been accepted.

Annex A: Platform Architectural Diagram

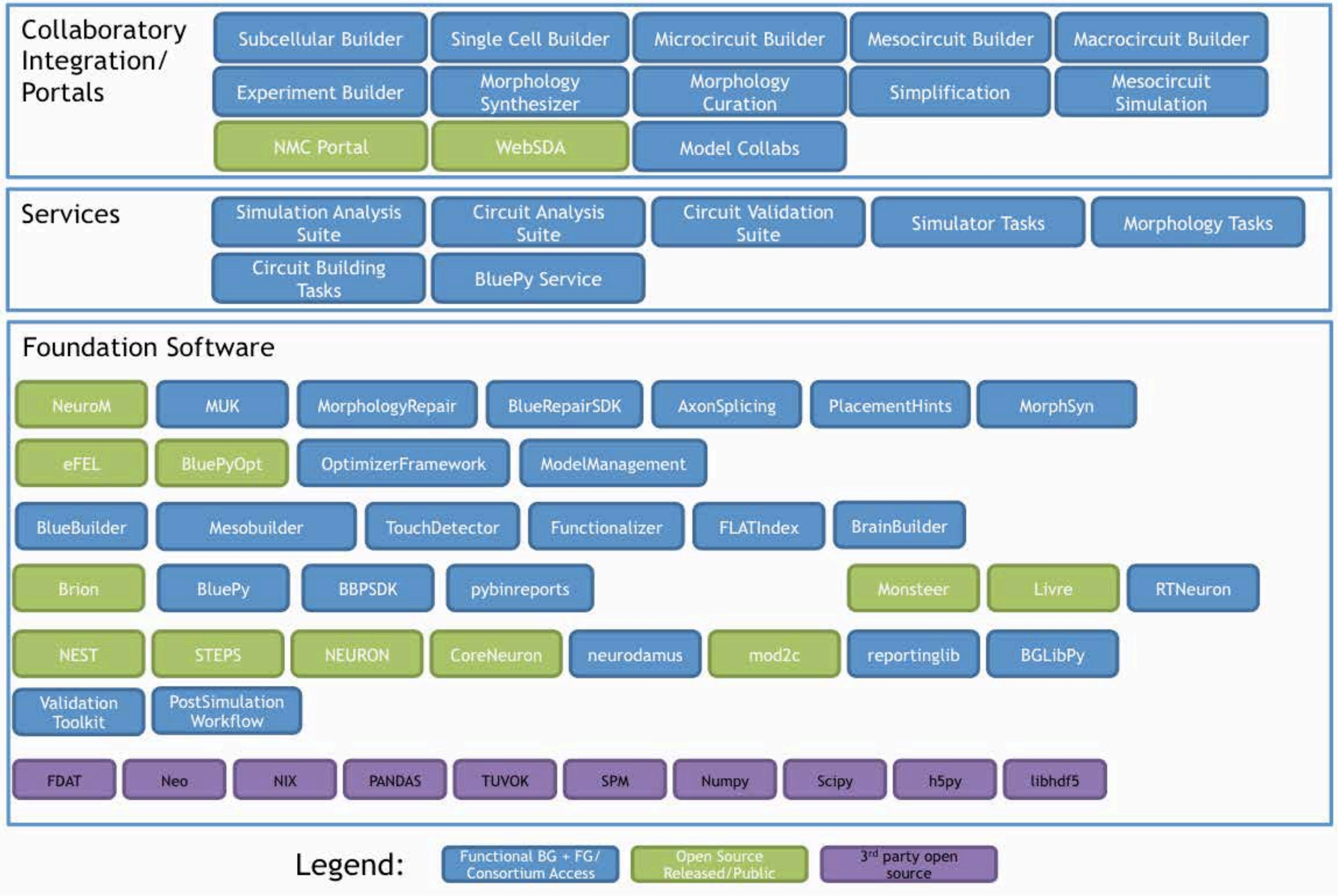

Figure 1: Architectural diagram of the Brain Simulation Platform

Annex B: Software and Services Included in this Platform Release

Table 1: Collaboratory Integration/Portals

Name	TRL	Description	Depends on	URL
Subcellular Builder	3	This Collab exemplifies the current capabilities for subcellular model building and simulation. It illustrates the following use cases of the specification document D6.7.1: • Brain Builder (SP6BSP-UC-001) • Simulation Configure and Launch (SP6BSP-UC-004) • Export a Volume Region of the Cellular Level Model and Add Molecular Level Detail to Produce a Molecular Level Model (SP6BSP-UC-014) • Geometrically Accurate Synapse Model with Molecular Reactions and Diffusion (SPBSP-UC-015) • Molecular Neuron Simulation using MolSim (SPBSP-UC-016)	Tasks • task steps_meshfile_creator • task steps_csv_bundle_creator • task steps_task • task steps_sim_analysis Foundation Software • STEPS	https://collab.humanbrainproject.eu/#/collab/161/nav/4963
Single Cell Builder	5	This Collab exemplifies the current capabilities for building electrical single cell models. It illustrates the following use case of the specification document D6.7.1: • Brain Builder (SP6BSP-UC-001) • Create a Complete Cell Model Using Automated Fitting of Conductance Densities (SP6BSP-UC-009)	Tasks • Run_remote_notebokk • Run_remote_opt_notebook Foundation Software • BluePyOpt • eFEL	https://collab.humanbrainproject.eu/#/collab/161/nav/4960
Microcircuit Builder	4	This Collab exemplifies the current	Tasks	https://collab.humanbrainproject.eu/#/c

		capabilities for building detailed microcircuits. It illustrates the following use case of the specification document D6.7.1: • Brain Builder (SP6BSP-UC-001) • Validations (SP6BSP-UC-002) • Compound Model and Model Component Analysis (SP6BSP-UC-003) • Distribute Cells and use this to Create a Detailed Neuron Model of a Rodent Neuronal Microcircuit (SP6BSP-UC-012)	• task_touch_detector • task _functionalizer • task_cell_densities_per_layer_from_mvd • task_cell_ratios_per_layer • task_inhibitory_synapses_density • task_intrinsic_extrinsic_synapse_densities_per_layer • task_intrinsic_inh_synapse_densities_per_layer • task_plot_layers • task_morphology_collage • task_plot_mosaic • task_somata_volume_fraction • task_spine_length • task_synapse_counts • task_synapses_overall_density Foundation Software • BrainBuilder • TouchDetector • Functionalizer • ValidationFwk • Bluepy	ollab/161/nav/4962
Mesocircuit/Macrocirc	3	This Collab exemplifies the current capabilities for building point neuron models	Tasks	https://collab.humanbrainproject.eu/#/collab/161/nav/4961

uit Builder		of brain regions or whole brainscale models (mouse). It illustrates the following use cases of the specification document D6.7.1: • Brain Builder (SP6BSP-UC-001) • Distribute Cells and Use this to Create a Point Neuron Model of a Brain Region (SP6BSP-UC-0010) • Distribute Cells and Use this to Create a Point Neuron Model of a Whole Rodent Brain (SP6BSP-UC-011)	• get_volumetric_data	
Experiment Builder	3	This Collab exemplifies the current capabilities for configuring and launching simulations. It illustrates the following use cases of the specification document D6.7.1: • Simulation Configure and Launch (SP6BSP-UC-004) • Simulation Analysis Tools (SP6BSP-UC-005) • Collaborative Review Process (SP6BSP-UC-006) • Simulation of a Microcircuit with Biophysically Realistic Neurons (SPBSP-UC-017) • Multi-Parameter Exploration of Medium sized Networks with Biophysically Detailed Neurons (SPBSP-UC-018) • Full-Scale Simulation of an Entire Brain Region with Biophysically Realistic Neurons (SPBSP-UC-019)	Tasks • task_voltage_collage • task_spike_raster • task_plot_psth • task_neuron_simulation Foundation Software • NEURON • Neurodamus • reportingLib Web Service: • BluePyService	https://collab.humanbrainproject.eu/#/collab/161/nav/3180
Morphology Synthesizer	2	This Collab exemplifies the current capabilities for synthesis of neuronal morphologies. It illustrates the following use	Tasks	https://collab.humanbrainproject.eu/#/collab/531/nav/4699

		case of the specification document D6.7.1: • Brain Builder (SP6BSP-UC-001) • Synthesise Full Cell Morphologies (SP6BSP-UC-008)	• task/MorphologyValidation • task/MorphologySynthesizer Foundation Software • NeuroM Other • morphsyn	
Morphology Curation	5	This Collab exemplifies the current capabilities for synthesis of neuronal morphologies. It illustrates the following use case of the specification document D6.7.1: • Repair and Diversification of Reconstructed Morphologies (SP6BSP-UC-007)	Tasks Foundation Software • morphology_repair_all • NeuroM	https://collab.humanbrainproject.eu/#/collab/161/nav/4948
Simplification	2	This Collab exemplifies the current capabilities for circuit simplification. It illustrates the following use case of the specification document D6.7.1: • Simplify the Cellular Level Model to a Network Level Model (SP6BSP-UC-013)	Tasks Foundation Software	https://collab.humanbrainproject.eu/#/collab/161/nav/6537
Mesocircuit Simulation	5	This Collab exemplifies the current capabilities for point neuron simulations (Network simulator). It illustrates the following use case of the specification document D6.7.1: • Simulation Configure and Launch (SP6BSP-UC-004) • Simulation of a Multi-Layered Local Cortical Mesocircuit with Full Scale Connectivity (SPBSP-UC-020)	Tasks • Task_Microcircuit_Task Foundation Software • NEST Other • Elephant	https://collab.humanbrainproject.eu/#/collab/161/nav/5307

		And lists proofpoints for: • Macrocircuit Model Combining Local with Macroscopic Connectivity (SPBSP-UC-021) • Verification of the Biological Relevance of Theoretical Prediction Obtained in the Large N Limit (SPBSP-UC-022) • Robustness of Network Activity with Respect to Neuron and Synapse Model (SPBSP-UC-023) • Verification of Independence of Simulated Network States from Time Discretisation (SPBSP-UC-024) • Flexibility in Specification of New Neuron and Spike-Time Dependent Plasticity (STDP) models (SPBSP-UC-025) • Detailed Investigation of the Correlation Structure of Neuronal Activity (SPBSP-UC-026)		
NMC Portal	9	This is a portal for the dissemination of the Neocortical Microcircuit as per Markram et al. 2015	Other • Liferay • D3.js • Three.js	https://collab.humanbrainproject.eu/#/collab/161/nav/3183
webSDA	8	Webserver for Simulation of Diffusional Association (SDA): making life much easier to simulate macromolecular diffusion.	Other • SDA • Play Framework • BioJava • Biopython	http://mcm.h-its.org/webSDA

Model Collab - “Cerebellum”	N/A	This Collab demonstrates results and work in progress from SP6 scientific drivers for the Brain Simulation Platform.	Foundation Software • Brainbuilder • eFel • OptimizerFramework See also “Getting Started” page in Collab.	https://collab.humanbrainproject.eu/#/collab/161/nav/3441
Model Collab - “Basal Ganglia”	N/A	This Collab demonstrates results and work in progress from SP6 scientific drivers for the Brain Simulation Platform.	Foundation Software • Brainbuilder • eFel • OptimizerFramework See also “Getting Started” page in Collab.	https://collab.humanbrainproject.eu/#/collab/161/nav/3442
Model Collab - “Hippocampus”	N/A	This Collab demonstrates results and work in progress from SP6 scientific drivers for the Brain Simulation Platform.	Foundation Software • Brainbuilder • eFel • OptimizerFramework See also “Getting Started” page in Collab.	https://collab.humanbrainproject.eu/ -/collab/594/nav/5317
Model Collab - “Subcellular Modelling”	N/A	This Collab demonstrates results and work in progress from SP6 scientific drivers for the Brain Simulation Platform.	Foundation Software • NEURON simulation bundle creator • NEURON simulation runner See also “Getting Started” page in Collab.	https://collab.humanbrainproject.eu/#/collab/161/nav/4678
Model Collab - “Human Cells”	N/A	This Collab demonstrates results and work in progress from SP6 scientific drivers for the Brain Simulation Platform. It includes a rare	Foundation Software	https://collab.humanbrainproject.eu/#/collab/161/nav/4683

		database for L2/3 reconstructed and physiologically measured cells models of these cells.	• BluePyOpt See also “Getting Started” page in Collab.	

Table 2: Services

Name	Version	Category	Description	Repository
task_assign_metype	1	task	Assign me-types to positions	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_assign_morphology	1	task	Assign morphologies to positions	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_assign_synapse_class	1	task	Assign ei type to positions	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_axon_splicing_task	1	task	Splice axons from an input set into a current circuit	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_brunel_delta_nest_task	1	task	Brunel delta nest	https://bbpcode.epfl.ch/platform/hbp-platform-tasks
task_build_morphology_bundle	1	task	Creates a bundle of morphologies.	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_cell_densities_per_layer_from_mvd	1	task	Cell densities per layer validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_cell_positioning	1	task	Place cells in regions	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_cell_ratios_per_layer	1	task	Gabaergic cell fractions per layer circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_createAavSynapses	1	task	Generate synapses from cell positions and aav voxel data.	https://bbpcode.epfl.ch/neurorobotics/genbrain
task_createCellPositions	1	task	Generate point positions.	https://bbpcode.epfl.ch/neurorobotics/genbrain

task_functionalize	1	task	Filter and functionalize neurite touches into synapses	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_get_volumetric_data	1	task	From copyinputmorphologies to clonemorphologies	https://bbpcode.epfl.ch/neurorobotics/genbrain
task_inhibitory_synapses_density	1	task	Inhibitory synapses density circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_intrinsic_extrinsic_synapse_densities_per_layer	1	task	Intrinsic+extrinsic synapse densities per layer circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_intrinsic_inh_synapse_densities_per_layer	1	task	Intrinsic inhibitory synapse densities per layer circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_mesobuilder_touch_detection	1	task	Run touch detection	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_microcircuit_hpc_task	1	task	Cortical microcircuit simulation	https://bbpcode.epfl.ch/platform/hbp-platform-tasks
task_microcircuit_task	1	task	Microcircuit model	https://bbpcode.epfl.ch/platform/hbp-platform-tasks
task_morphology_collage	1	task	Plot a collage of random morphology by layer and m-type	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_morphology_repair_all	1	task	From copyinputmorphologies to clonemorphologies	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_morphology_synthesis	1	task	Runs a morphology synthesis job	https://bbpcode.epfl.ch/algorithms/tasks/MorphologySynthesizer
task_morphology_validation	1	task	Neurom morph check on a set of morphologies	https://bbpcode.epfl.ch/algorithms/tasks/MorphologyValidation
task_neuron_bundle_creator	1	task	Create a neuron simulation bundle.	https://bbpcode.epfl.ch/subcellular/MolConfig
task_neuron_task	1	task	Run a neuron simulation	https://bbpcode.epfl.ch/subcellular/MolConfig
task_plot_layers	1	task	Plot layers	https://bbpcode.epfl.ch/platform/tests/SampleTask

task_plot_mosaic	1	task	Produce a plot of the mosaic geometry and minicolumn centers in the merged circuit.	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_plot_psth	1	task	Peristimulus-time histogram plot	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_plot_psth_mpld	1	task	Peristimulus-time histogram plot (mpld version)	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_run_remote_notebook	1	task	Run remote notebook	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_run_remote_opt_notebook	1	task	Run remote opt notebook	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_select_region	1	task	Select brain region	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder
task_simulation_launch_viz	1	task	Run a bbp simulation on the viz cluster	https://bbpcode.epfl.ch:22/platform/tests/SampleTask
task_single_neuron_task	1	task	Single neuron	https://bbpcode.epfl.ch/platform/hbp-platform-tasks
task_somata_volume_fraction	1	task	Somata volume fraction circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_spike_raster	1	task	Spike raster plot	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_spine_length	1	task	Spine length distribution validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_splicing_collage	1	task	Plot a collage of spliced morphologies before and after by layer and m-type	https://bbpcode.epfl.ch/platform/tests/SampleTask
task_steps_sim_analysis	1	task	Run analysis on a steps simulation	https://bbpcode.epfl.ch/subcellular/MolConfig
task_synapse_counts	1	task	Synapse counts circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_synapses_overall_density	1	task	Synapses overall density circuit validation	https://bbpcode.epfl.ch/platform/tasks/CircuitValidations
task_touch_detect	1	task	Detect touches between morphologies	https://bbpcode.epfl.ch/platform/tasks/BrainBuilder

task_tsodyks_depressing_task	1	task	Tsodyks depressing	https://bbpcode.epfl.ch/platform/hbp-platform-tasks
task_voltage_collage	1	task	Create an analysis plot with a collage of voltage reports	https://bbpcode.epfl.ch/platform/tests/SampleTask

Table 3: Foundation Software

Name	Version	TRL	Category	License	Description	Repository	Documentation
AxonSplicing	0.0.0	4	tool		Tool to splice axons into an existing circuit	https://bbpcode.epfl.ch/code/platform/AxonSplicing	https://bbpcode.epfl.ch/browse/code/platform/AxonSplicing/
BBPSDK	0.21.0	6	library		Blue Brain Project Software Development Kit	https://bbpcode.epfl.ch/code/common/BBPSDK	https://bbpcode.epfl.ch/browse/code/common/BBPSDK/
BGLibPy	64a38fd	5	tool		The pythonic interface to BGLib functionality	https://bbpcode.epfl.ch/code/#/admin/projects/sim/BGLibPy	
BlueBuilder	1.2	5	tool		Blue Builder application is used by the Blue Brain Project for the placement of soma during the circuit building process	https://bbpcode.epfl.ch/code/#/admin/projects/building/BlueBuilder	https://bbp.epfl.ch/documentation/code/#BlueBuilder
BluePy	0.5.11	5	library		The pythonic interface to BlueBrain production entities.	https://bbpcode.epfl.ch/code/analysis/BluePy	https://bbpcode.epfl.ch/browse/code/analysis/BluePy/
BluePyOpt	0.2.0	3	library	LGPLv3	The Blue Brain Python Optimisation Library	https://github.co	https://github.c

					(BluePyOpt) is an extensible framework for data-driven model parameter optimisation that wraps and standardises several existing open-source tools. It simplifies the task of creating and sharing these optimisations, and the associated techniques and knowledge. This is achieved by abstracting the optimisation and evaluation tasks into various reusable and flexible discrete elements according to established best-practices. Further, BluePyOpt provides methods for setting up both small- and large-scale optimisations on a variety of platforms, ranging from laptops to Linux clusters and cloud-based compute infrastructures.	m/BlueBrain/BluePyOpt	om/BlueBrain/BluePyOpt
BlueRepairSDK	1.2.0	4	tool		Set of tools to repair morphologies	https://bbpcode.epfl.ch/code/platform/BlueRepairSDK	https://bbpcode.epfl.ch/browse/code/platform/BlueRepairSDK/
BrainBuilder	0.0.3.dev0	4	library		A modularised API to build brain regions based on volumetric data	https://bbpcode.epfl.ch/platform/BrainBuilder	
Brion	1.5.0	6	library		Blue Brain C++ File IO Library	https://github.com/BlueBrain/Brion.git	https://github.com/BlueBrain/Brion
CoreNeuron	0.1	6		"three clause" BSD	CoreNeuron implements the core functionalities of the NEURON simulator targeting high efficiency when using millions of threads.	https://github.com/BlueBrain/CoreNeuron	https://github.com/BlueBrain/CoreNeuron
eFEL	2.7.7	6	library	LGPL	Electrophys Feature Extraction Library	https://github.com/BlueBrain/eFEL.git	https://github.com/BlueBrain/eFEL

FLATIndex	1.7	4	tool		Spatial Indexer (FLATIndex) Tool for the Blue Brain Project	https://bbpcode.epfl.ch/code/#/admin/projects/building/FLATIndex	https://bbp.epfl.ch/documentation/code/#FLATIndex
Functionalizer	3.5	5	tool		Functionalizer is used by the Blue Brain Project to build a functional circuit by applying a series of filters to the structural circuit dating circuit building process	https://bbpcode.epfl.ch/code/#/admin/projects/building/Functionalizer	https://bbp.epfl.ch/documentation/code/#Functionalizer
Livre	0.4.0	6	tool	LGPL	Largescale Interactive Volume Rendering Engine	https://github.com/BlueBrain/Livre	https://github.com/BlueBrain/Livre
Mesobuilder	0.2.5	4	tool		Controls the various phases of building and touch detecting a mosaic of microcircuits	https://bbpcode.epfl.ch/code/#/admin/projects/sim/mesobuilder	
mod2c	1.1.1	5		"three clause" BSD	Mod2C is a converter of .mod files into .c files.	https://bbpcode.epfl.ch/code/#/admin/projects/sim/mod2c	https://bbp.epfl.ch/documentation/code/mod2c-1.1/index.html
ModelManagement	0.0.1	4	tool		A workflow tool to reuse an existing etype and apply it to a series of morphologies	https://bbpcode.epfl.ch/code/#/admin/projects/sim/ModelManagement	https://developer.humanbrainproject.eu/docs/projects/modelmanagement/0.1.0/index.html
Monsteer	0.2.0	6	library	LGPL	Interactive Supercomputing tools and library	https://github.com/BlueBrain/Monsteer	https://github.com/BlueBrain/Monsteer
MorphologyRepair	0.0.1	4	tool		Morphology Repair workflow to curate and fix morphologies.	https://bbpcode.epfl.ch/code/pla	https://bbpcode.epfl.ch/browse

						tform/MorphologyRepair	/code/platform/MorphologyRepair
morphsyn (Morphology Synthesis)	0.02	4	application		Software for synthesising neuronal morphologies embedded in a specific brain region	bbpcode.epfl.ch/algorithms/synthesis/morphsyn	
MUK	3.7.0	4	tool		Morphology Utility Toolkit	https://bbpcode.epfl.ch/code/sim/MUK	https://bbpcode.epfl.ch/browse/code/sim/MUK
NEST (NetSim)	2.8.0	9	tool	GPL2	The NEST simulator	https://github.com/nest/nest-simulator	http://www.nest-simulator.org/documentation/
Neurodamus	1.8.0	5			Neurodamus (formerly BGlib)	https://bbpcode.epfl.ch/code/#/admin/projects/sim/neurodamus/bbp	https://bbp.epfl.ch/documentation/code/Neurodamus-1.8/index.html
NeuroM	0.0.11	5	library	BSD Simplified	NeuroM is a Python-based toolkit for the analysis and processing of neuron morphologies	https://github.com/BlueBrain/NeuroM.git	https://github.com/BlueBrain/NeuroM
NEURON (CellSim)	7.4	9		"three clause" BSD	Simulator supporting empirically-based simulations of neurons and networks of neurons	http://www.neuron.yale.edu/hg/neuron/nrn	https://www.neuron.yale.edu/neuron/
OptimizerFramework	2.5	6	tool		A tool that is able to apply optimization algorithms to the building of neuron electrical models	https://bbpcode.epfl.ch/code/#/admin/projects/sim/OptimizerFramework	https://developer.humanbrainproject.eu/docs/projects/OptimizerFramework/2.3-49/index.html

PlacementHints	0.0.1	3	tool		Tool to improve the placement of neurons in the irrespective layers in a way that satisfies multiple biological constraints	https://bbpcode.epfl.ch/code/#/admin/projects/building/placementAlgorithm	
PostSimulationWorkflow	0.0.10	4	tool		A command-line tool that registers simulation results with the portal and performs common analysis and plots.	https://bbpcode.epfl.ch/browse/code/analysis%2FPostSimulationTasks/	https://developer.humanbrainproject.eu/docs/postsimulationutils/latest/
pybinreports	0.1	4	library		Python bindings for part of ReportingLib	https://bbpteam.epfl.ch/reps/sim/pybinreports/	
ReportingLib	2.4	5	library		Reporting library is used by the Blue Brain Project as an I/O library to Neuron/Neurodamus simulator. It allows output reports in several formats (binary, ASCII, HDF5)	https://bbpcode.epfl.ch/code/#/admin/projects/sim/reportinglib/bbp	https://bbp.epfl.ch/documentation/code/#ReportingLib
RTNeuron	2.8.0	6	tool	Proprietary, LGPL planned	RTNeuron is a tool for the interactive visualisation and media production of cortical column simulation results	https://bbpcode.epfl.ch/code/viz/RTNeuron	https://bbpcode.epfl.ch/browse/code/viz/RTNeuron/
STEPS (MolSim)	2.2.0	8	tool	GPL3	Stochastic Engine for Pathway Simulation	http://steps.sourceforge.net/STEPS/	http://steps.sourceforge.net/STEPS/default.php
TouchDetector	3.2	5	tool		Part of the Simulation tools for the Blue Brain Project	https://bbpcode.epfl.ch/code/#/admin/projects/building/TouchDetector	https://bbp.epfl.ch/documentation/code/#TouchDetector

Validation Toolkit	0.0.2	4	library		Validation Toolkit	https://bbpcode.epfl.ch/code/#/admin/projects/platform/ValidationToolkit	https://bbpcode.epfl.ch/code/#/admin/projects/platform/ValidationToolkit

Annex C: Summary - Platform Use Case Status

The specific developments between the initial internal release of the BSP and this first public release are described in the Status/comments column.

Table 4: Platform Use Case Status Summary

Use Case ID	Success Scenario	Status/comments	ID of Related Functional Requirement	Related Product/Software Package/Service	TRL at end of Ramp-Up Phase	Contributors
SP6BSP-UC-001 Brain Builder	1) Abigail selects an existing COLL Project to work within or creates a new COLL Project. 2) Abigail selects the build for the model type she wants to build. 3) The builder Task guides her through the input selection steps using content typed inputs to filter the list of valid inputs, using previous configurations as initial defaults. 4) Abigail executes the Model Build Task as a Job and is informed when her Job is complete. 5) A minimal set of validation Tasks is run.	For the internal release, brain building functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped with the Task Service and available in Jupyter notebooks as illustrated in several Collabs.	SP6BSP-FR-001, SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/4963 https://collab.humanbrainproject.eu/#/collab/161/nav/4960 https://collab.humanbrainproject.eu/#/collab/161/nav/4962 https://collab.humanbrainproject.eu/#/collab/161/nav/4961	TRL4	WP6.1, WP6.5 Jean-Denis COURCOl (EPFL) Michael GEVAERT (EPFL) Luis RIQUELME (EPFL)
SP6BSP-UC-002 Validations	1) Abigail wants to validate the circuit she has found in the COLL or has built herself. 2) A minimal set of validations has already been run as part of the build process. She reviews these and decides that several more computationally costly validations are required. 3) Abigail selects and runs validations she have been filtered for applicability to the model content type she has selected.	For the internal release, first validations were implemented using the ValidationSuite, which sits on top of the TaskService. For the public release, the intial set has been extended. User can	SP6BSP-FR-002, SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/4962 https://collab.humanbrainproject.eu/#/collab/531/nav/4699 https://collab.humanbrainproject.eu/#/collab/161/nav/4948	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) James DYNES (EPFL)

	4) Validation Tasks output standardised validation reports, which are added to the list of all validations that have been run on the circuit.	add validations using this mechanism.				
SP6BSP-UC-003 Compound Model and Model Component Analysis	1) Abigail finds a released brain or brain component model in the COLL using the integrated search functionality. 2) She would like to review some of the analyses that have been run on this circuit model. 3) She uses the COLL Project Viewer to show her which analyses have been run and their results. 4) Abigail would like to see a pathway connectivity report, but one has not yet been generated. She can select the Pathway Connectivity Report Task and launch it as a Job. She will be notified when the analysis completes.	This functionality is provided by the provenance tracking of the TaskService and Collaboratory.	SP6BSP-FR-002, SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/4962	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) James DYNES (EPFL)
SP6BSP-UC-004 Simulation Configure and Launch	1) Abigail finds a released brain or brain component model in the COLL using the integrated search functionality. 2) Abigail adds the model to a new COLL Project, COLL Project 5. 3) Abigail selects which type of simulation she would like to perform on the model from three options: Molecular, Cellular or Network. In this particular case she chooses the Cellular simulator as her target. 4) The selection causes the Platform to open a Cell Experiment builder that allows Abigail to configure whichever parameters the Experiment builder supports. 5) Abigail then launches the simulation Task as a Job to run an HPC Platform compute resource. During the launch process, she may be asked (depending on the Task definition) to make a decision about which	For the internal release, simulation configure and launch functionality was only available for the Network Simulator. Now this functionality is available for Cellular simulations as well as Molecular Simulations. Job launch and data transfer mostly relies on the TaskService and DocumentService. First prototype for launching and via UNICORE has been	SP6BSP-FR-003, SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/4963 https://collab.humanbrainproject.eu/#/collab/161/nav/3180 https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) Luis RIQUELME (EPFL) Fabien DELALONDRE (EPFL)

	computing resource to run on and where to store final output data. The HPC Platform could provide recommendations based on data locality, data transfer volume and compute resource availability. 6) Once it is complete, the Job registers some or all of its output (as determined by the Task definition) from COLL Project 5 in the COLL. The output may be marshalled to its final storage location by the HPC Platform Data Transfer service.	demonstrated.				
SP6BSP-UC-005 Simulation Analysis Tools	1) Abigail adds Bill to the COLL Project with write permissions to help her analyse the output data using analysis Tasks available for simulations in the COLL. 2) Bill can now run analysis Jobs on the simulation. 3) Analysis output will be associated with the analysed simulation in the COLL Project view.	This functionality is now implemented through the Task Service and available in Jupyter notebooks as illustrated in several Collabs.	SP6BSP-FR-002, SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/3180 https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Luis RIQUELME (EPFL) Michael GEVAERT (EPFL)
SP6BSP-UC-006 Collaborative Review Process	1) Abigail and Bill have a teleconference to discuss the results of the latest model refinement. Abigail and Bill have already run their analysis and both visit the Portal to review the results together. 2) Abigail and Bill notice something in the simulation activity that they can't explain. They trace through the validations that were run on the model components and can't find a clear explanation. 3) They invite Chris into the COLL Project because they think he might have insight on the phenomenon they have observed in silico. 4) Chris joins the COLL Project and runs a couple of additional analyses. He explains that the observed behaviour makes sense in light of a recent paper he read and the results of the analysis. 5) Abigail is happy with the state of the model and releases it for the community to analyse and review	This functionality is provided by the provenance tracking of the TaskService and Collaboratory	SP6BSP-FR-004	https://collab.humanbrainproject.eu/#/collab/161/nav/3180	TRL3	WP6.5 Jean-Denis COURCOL (EPFL)

	further.					
SP6BSP-UC-007 Repair and Diversification of Reconstructed Morphologies	1) Abigail wants to generate a large collection of morphologies based on repaired and diversified reconstructed morphologies. These morphologies will populate a detailed circuit model in a particular region. 2) She selects the Morphology Release Task and builds her set of input morphologies from those available in the Neuroinformatics Platform. 3) Abigail then launches the Morphology Release Task as a Job. 4) Once the portal notifies Abigail that the Job is complete, Abigail can run the Morphology Validation suite to verify that the morphologies match all expected biological characteristics.	For the internal release, morphology curation and repair functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped with the Task Service and available in Jupyter notebooks as illustrated in a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/4948	TRL5	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL)
SP6BSP-UC-008 Synthesise Full Cell Morphologies	1) Abigail wants to synthesise cell morphologies to test that synthesis is working properly. 2) She selects the cell synthesis configuration tool and searches the Neuroinformatics Platform for synthesis configurations for the appropriate species, brain region, age, etc. 3) Abigail then launches the Cell Synthesis Job. 4) Once the Portal notifies Abigail that cell synthesis is complete, Abigail can run the Morphology Validation suite to verify that the morphologies match all expected biological characteristics.	For the internal release, synthesis functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped with the Task Service and available in Jupyter notebooks as illustrated in a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/531/nav/4699	TRL3	WP6.1, WP6.5 Julian SHILLCOCK (EPFL) Juan PALACIOS (EPFL) Liesbeth VANHERPE (EPFL) Guy ATENEKENG KAHOU (EPFL) Lida KANARI (EPFL) Athanassia CHALIMOURDA (EPFL) Eleftherios ZISIS (EPFL)
SP6BSP-UC-009 Create a	1) Bill wants to create a complete Cell model from a validated morphology.	For the internal release, cell building functionality was only	SP6BSP-FR-005, SP6BSP-FR-006,	https://collab.humanbrainproject.eu/#/collab/161/nav/4960	TRL5	WP6.5 Jean-Denis COURCOI (EPFL)

Complete Cell Model Using Automated Fitting of Conductance Densities	2) Bill selects an exemplar morphology from a list of previously validated neuron morphologies. The exemplar is used as input to the automated conductance density-fitting tool, and Bill configures the tool with the correct optimisation settings. 3) Bill submits the configuration for execution as a Job. 4) When the Job is complete, an electrical-type template is produced. The electrical-type template can be considered validated, if the automated conductance density fitting tool functions. As a result, no further validation of the electrical type template is necessary. 5) Bill applies his electrical type template to candidate morphologies to validate that the morphological and electrical type combinations are valid. This process produces combined morphological and electrophysiological models known as METypes. Validated METypes can now be used standalone or in cellular level model building.	available from the command line using Foundation software on the HBP Development Computer. Now this functionality is much improved due to the newly developed BluePyOpt Foundation software, available in Jupyter notebooks as illustrated in a Collab	SP6BSP-FR-007			Michael GEVAERT (EPFL) Werner VAN GEIT (EPFL)
SP6BSP-UC-010 Distribute Cells and Use this to Create a Point Neuron Model of a Brain Region	1) Bill would like to build a point neuron model of a brain region. 2) Bill selects volumetric inputs for Excitatory-inhibitory ratio, Cell density, M-type ratio and connectivity ratios. 3) Bill executes the point neuron region builder workflow Task as a Job. 4) When the Job completes, Bill can use BSP Tasks to analyse and validate the completed circuit. 5) Bill can also use the BSP Network-level simulation configuration system to configure and launch simulations.	For the internal release, point neuron brain building functionality was only available from the command line using prototypic Foundation software on the HBP Development Computer. Now this functionality is wrapped within a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/4961	TRL3	WP6.1, WP6.5 Csaba EROE (EPFL)

SP6BSP-UC-011 Distribute Cells and Use this to Create a Point Neuron Model of a Whole Rodent Brain	1) Bill would like to build a point neuron model of a whole brain. 2) Bill selects volumetric inputs for Excitatory-inhibitory ratio, Cell density, M-type ratio and connectivity ratios. 3) Bill executes the point neuron whole Brain Builder workflow Task as a Job. 4) When the Job completes, Bill can use BSP Tasks to analyse and validate the completed circuit. 5) Bill can also use the BSP Network Experiment Builder to configure and launch simulations.	For the internal release, point neuron brain building functionality was only available from the command line using prototypic Foundation software on the HBP Development Computer. Now this functionality is wrapped within a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/4961	TRL3	WP6.1, WP6.5 Csaba EROE (EPFL) Marc-Oliver GEWALTIG (EPFL) Till SCHUMANN (JUELICH) Fabien DELALONDRE (EPFL)
SP6BSP-UC-012 Distribute Cells and use this to Create a Detailed Neuron Model of a Rodent Neuronal Microcircuit	1) Bill would like to build a detailed neuron microcircuit model in a particular rodent brain region. The microcircuit would represent a small portion of the neuron population of the full region. 2) Bill selects the species, age, region, etc. where the microcircuit will be built. This defines the ontological model context. In this case he will use a preferred species, but the other pieces of model context could be selected based on availability of data. 3) Bill selects a previous microcircuit recipe for the selected model context. 4) Bill executes the microcircuit builder workflow Task as a Job. 5) When the Job completes, Bill can use BSP Tasks to analyse and validate the completed microcircuit. 6) Bill can also use the BSP Cellular level simulation configuration system to configure and launch simulations using the microcircuit model.	For the internal release, detailed microcircuit brain building functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is integrated with the TaskSerice and demonstrated in a Jupyter notebook within a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/4962	TRL6	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) Luis RIQUELME (EPFL) Fabien DELALONDRE (EPFL) Christian RÖSSERT (EPFL) Christian POZZORINI (EPFL)

SP6BSP-UC-013 Simplify the Cellular-Level Model to a Network-Level Model	1) Bill selects a Cellular-level microcircuit, brain region circuit or multi-region circuit of interest. 2) Bill selects the Network-level model export Task. 3) Bill configures the Network-level model export Task. 4) Bill runs the Network-level model export Task as a Job. 5) Upon completion, the Job deposits the exported Network-level simulation in the same COLL Project as the source Cellular-level model.	Proof of concept/implementation will be achieved and reported in a pre-print manuscript. Deployment in the Platform depends on features not yet available.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/6537	TRL3	WP6.1, WP6.5 Christian RÖSSERT (EPFL) Eilif MULLER (EPFL) Christian POZZORINI (EPFL) Marc-Oliver GEWALTIG (EPFL) Csaba EROE (EPFL) Idan SEGEV (HUJI)
SP6BSP-UC-014 Export a Volume Region of the Cellular-Level Model and Add Molecular-Level Detail to Produce a Molecular-Level Model	1) Bill selects a Cellular-level microcircuit, brain region circuit or multi-region circuit of interest. 2) Bill selects the Molecular-level model export Task. 3) Bill selects a sub-region of space in the Cellular-level model to export. 4) Bill configures the Molecular-level model export Task. 5) Bill runs the Molecular-level model export Task as a Job. 6) Upon completion, the Job deposits the exported Molecular-level simulation in the same COLL Project as the source Molecular-level model.	For the internal release, this functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is demonstrated in a Collab. Since generation of primary geometries takes several hours, users can select from a gallery of pre-generated meshes.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/4963	TRL3	WP6.4, WP6.5 Daniel KELLER (EPFL) Jean-Denis COURCOL (EPFL) James DYNES (EPFL)
SP6BSP-UC-015 Geometrically Accurate Synapse Model	1) Abigail wants to conduct an in silico experiment on a molecular-level synapse between a bi-tufted cell and a pyramidal cell. 2) She chooses the appropriate synapse from the	For the internal release, this functionality was only available from the command line using	SP6BSP-FR-002, SP6BSP-FR-003, SP6BSP-FR-004, SP6BSP-FR-005, SP6BSP-FR-006,	https://collab.humanbrainproject.eu/#/collab/161/nav/4963	TRL3	WP6.4 Daniel KELLER (EPFL) James DYNES (EPFL)

with Molecular Reactions and Diffusion	neuron pair model in the Brain Simulation Portal. 3) Abigail then opens the Cellular Experiment Builder and configures the stimulation protocol for the presynaptic neuron of interest. She furthermore selects the duration of the experiment. 4) She then launches the cellular simulation from the Brain Simulation Portal, selecting the CellSim. The voltage traces of the presynaptic neuron spiking and the post-synaptic neuron's synaptic potentials are recorded. 5) Once the Portal notifies Abigail that the cellular simulation is complete, Abigail uses the Exporter Module to extract the detailed geometry of the chosen synapses together. She is able to curate and adjust the mesh generation properties for the geometry at this stage to match electron microscopic data previously identified in the Neuroinformatics Platform. 6) Abigail configures the model with molecules; selects a subset of proteins, sets initial concentrations for each, defines the distribution of each protein, and sets the reaction kinetics between each protein and another reaction partner. 7) In the BSP, Abigail can now choose the MolSim as the target simulator for her molecular synapse and uses a subset of the presynaptic traces to stimulate the synapse and the cellular-level synaptic response as a target result. 8) A BSP application will analyse the target synaptic responses and use the features extracted (amplitudes, latencies, etc.) as a target in a multi-objective feature optimisation fitting that adjusts only the relative concentrations of the proteins until the target synaptic response is recreated.	Foundation software on the HBP Development Computer. Now this functionality is demonstrated in a Collab. Users can select from a gallery of pre-generated synapses, without direct interaction in the generation process. Task completion notifications are through the platform.	SP6BSP-FR-007, SP6BSP-FR-008, SP6BSP-FR-009, SP6BSP-FR-010, SP6BSP-FR-011, SP6BSP-FR-012, SP6BSP-FR-013, SP6BSP-FR-014, SP6BSP-FR-016			

	9) Abigail now stimulates the synapse with a different subset of voltage traces from the presynaptic neuron to validate the synapse model. 10) She chooses the molecule species she would like to track and launches the MolSim from the Portal; she also chooses how many random seeds she would like to simulate for the diffusion process. 11) The Portal notifies Abigail by email/SMS when the simulations have finished. Abigail now has the opportunity to analyse the concentrations of molecules in the presynaptic terminal, in the cleft and in the synaptic spine. 12) Abigail now invites an expert in synaptic biophysics and physiology to review the results and provide further validation data if necessary. 13) Abigail releases the model for others to use and then publishes a multi-author paper describing the role of the chosen molecule in synaptic transmission.					
SP6BSP-UC-016 Molecular Neuron Simulation using MolSim	1) Bill wants to study a molecular-level model of an entire neuron, stimulated with network activity drawn from a circuit simulation. 2) He chooses an appropriate neuron within a previously built circuit from the BSP. In the Molecular Model Builder he chooses to export the entire neuron volume mesh to MolSim. He has the opportunity to curate and adjust the mesh generation properties at this stage. 3) Bill then opens the Molecular Experiment Builder. He chooses a pre-run Cellular simulation run in NEURON that includes all of the other neurons in the circuit. The simulations available for selection will be only those that were run on the circuit selected in 2). This step sets the synaptic activity for the Molecular	Users can select from a gallery of pre-generated meshes.	SP6BSP-FR-002, SP6BSP-FR-003, SP6BSP-FR-004, SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007, SP6BSP-FR-008, SP6BSP-FR-009, SP6BSP-FR-010, SP6BSP-FR-011, SP6BSP-FR-012, SP6BSP-FR-013, SP6BSP-FR-014	https://collab.humanbrainproject.eu/#/collab/161/nav/4963	TRL3	WP6.4, WP6.5 Daniel KELLER (EPFL) James DYNES (EPFL) Sam YATES (EPFL)

	Experiment. 4) Bill selects the subsets of proteins and synapse configuration parameters to include in the simulation. These are drawn from protein libraries stored in the NIP. 5) Once the simulations are complete, the portal notifies Bill. He can then apply a range of other stimuli to study the resulting molecular interactions in an interactive analysis session on a high-fidelity cockpit.					
SP6BSP-UC-017 Simulation of a Microcircuit with Biophysically Realistic Neurons	1) Abigail wants to conduct an in silico experiment on the cortical microcircuit in order to test the effect of thalamic stimulation; the microcircuit is too big and too detailed to be simulated on a workstation and requires a supercomputer. 2) She chooses the appropriate microcircuit model from the Brain Simulation Portal that was previously built. 3) Abigail opens the Cellular Experiment Builder and configures thalamic inputs and a stimulation protocol using data identified in the NIP. She selects the duration of the experiment and selects a set of neurons from which she wants to have current clamp traces recorded. 4) She then launches the simulation from the Brain Simulation Portal. 5) Once the Portal notifies Abigail that the Job is complete, Abigail runs an analysis on the completed simulation to produce a movie of the voltage activity of the microcircuit.	For the internal release, simulation functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped for execution in a Collab.	SP6BSP-FR-002, SP6BSP-FR-003, SP6BSP-FR-004, SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007, SP6BSP-FR-012, SP6BSP-FR-013, SP6BSP-FR-014, SP6BSP-FR-015, SP6BSP-FR-016	https://collab.humanbrainproject.eu/#/collab/161/nav/3180	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) Fabien DELALONDRE (EPFL)
SP6BSP-UC-018 Multi-Parameter	1) Bill is scheduled to do an experiment in the wetlab the following day; he has a hypothesis of the effect of a particular multi-electrode array stimulation pattern.	For the internal release, simulation functionality was only	SP6BSP-FR-002, SP6BSP-FR-003, SP6BSP-FR-004,	https://collab.humanbrainproject.eu/#/collab/161/nav/3180	TRL3	WP6.5 Jean-Denis COURCOL (EPFL)

Exploration of Medium-Sized Networks with Biophysically Detailed Neurons	What he doesn't know is the exact pattern and stimulus strength. 2) Abigail has a mesocircuit model of the area of interest in the Brain Simulation Platform. 3) Bill asks Abigail whether she can run a parameter exploration on a mesocircuit model in order to help him reduce the amount of experiment time; however, they only have one day to come up with the computational answer because of the lab schedule. 4) She chooses the appropriate brain region model from the Brain Simulation Portal that was previously built. 5) Abigail opens the Cellular Experiment Builder and configures a parameter sweep on for the multi-electrode array stimulation. She selects the duration of the experiment and the recording channels. 6) She queries the Brain Simulation Portal for the duration of the simulation and realises that the sweep will not finish in time with the default CellSim. 7) Chris has previously worked on the performance characterisation of the CellSim code and has developed an optimised version of it; the Portal indicates to Abigail that a number of different CellSim versions are available and shows what changes have been made in those versions. Chris' version is described as optimised, so Abigail decides to directly contact Chris. 8) Abigail opens a chat with Chris and discusses her simulations; Chris informs Abigail that the simulation she is about to run is covered by his optimised version of CellSim. 9) Abigail chooses to go ahead with the optimised version of CellSim and launches the parameter	available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped for execution in a Collab.	SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007, SP6BSP-FR-012, SP6BSP-FR-013, SP6BSP-FR-014, SP6BSP-FR-015, SP6BSP-FR-016			Michael GEVAERT (EPFL) Fabien DELALONDRE (EPFL)

	sweep; she subscribes Bill and Chris to the notification 10) Once the job is complete, Abigail, Bill and Chris are informed; Abigail performs further analysis; Bill goes about launching his experiment; Chris analyses the performance data of the parameter sweep.					
SP6BSP-UC-019 Full-Scale Simulation of an Entire Brain Region with Biophysically Realistic Neurons	1) Abigail wants to conduct an in silico experiment on an entire brain region model of 10 million detailed neurons; due to all the biophysical detail, the brain model is too big to be simulated on her regular supercomputer allocation. 2) She chooses the appropriate brain region model from the Brain Simulation Portal that was previously built. 3) Abigail then opens the Experiment Builder and configures a bath experiment protocol. She selects the duration of the experiment, and then selects the option to have Local Field Potential recorded. 4) She then launches the simulation from the Brain Simulation Portal. 5) The portal informs Abigail that the brain model of her choice cannot be simulated on the currently available resources. It informs Abigail that she can a) wait about two days before a bigger partition frees up, or b) use a memory-optimised version of the CellSim, which is possible since the chosen model ingredients are supported by this simulator. 6) Abigail chooses to go ahead with the memory-optimised version of CellSim. 7) The Portal invokes a special process of writing a cache-efficient memory configuration for the memory-optimised CellSim and launches the simulation; it sets up the data and post-processing	For the internal release, simulation functionality was only available from the command line using Foundation software on the HBP Development Computer. Now this functionality is wrapped for execution in a Collab.	SP6BSP-FR-002, SP6BSP-FR-003, SP6BSP-FR-004, SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007, SP6BSP-FR-012, SP6BSP-FR-013, SP6BSP-FR-014, SP6BSP-FR-015, SP6BSP-FR-016	https://collab.humanbrainproject.eu/#/collab/161/nav/3180	TRL3	WP6.5 Jean-Denis COURCOL (EPFL) Michael GEVAERT (EPFL) Fabien DELALONDRE (EPFL)

	pipeline to calculate the Local Field Potential on the fly. 8) Once the Portal notifies Abigail that the Job is complete, Abigail can invoke frequency analysis on the recorded local field potential. 9) She shares the local field potential recordings with Bill, who has recorded LFP from the same brain region in the wetlab. Together they discuss the results.					
SP6BSP-UC-020 Simulation of a Multi-Layered Local Cortical Mesocircuit with Full Scale Connectivity	1) Alice wants to investigate the statistics of the spiking activity emerging in a network with realistic layer-specific connectivity. 2) To this end, Alice needs to simulate a point neuron microcircuit with full-scale connectivity. This is a point neuron network model with a number of neurons and synapses in the right order of magnitude for a small brain region (100,000 neurons, 1 billion synapses). 3) Alice specifies her network for NetSim by writing a compact Python script, making use of powerful high-level neuron and connectivity construction routines, which support randomisation of the connectivity itself as well as of connection parameters such as synaptic weights and delays. 4) She stimulates the network with stationary input using stimulation devices provided by NetSim. 5) Alice also connects a subset of neurons in the network to spike detectors, which will record spike trains to file during simulation. 6) Alice runs a downscaled version of her network on a local cluster to ensure that her simulation script works as expected. 7) Simulating the network at full scale requires HPC	For the internal release mesocircuit simulation functionality using NEST was integrated in an initial Collab. This Collab and integration has been improved and now includes launching of simulation using SP7's UNICORE interface.	SP6BSP-FR-001, SP6BSP-FR-003, SP6BSP-FR-005, SP6BSP-FR-007, SP6BSP-FR-008	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)

	resources due to memory footprint and simulation time. Alice thus decides to use the API of the Brain Simulation Platform to execute her simulation. 8) The BSP runs Alice's simulation on available HPC resources and registers resulting output data together with the simulation script and NetSim configuration information. 9) Alice retrieves the results and analyses them using the tools provided by the HBP-COLL. 10) To explore the effect of spatial structure on network dynamics, Alice slightly modifies her original script to make connectivity and propagation delays distance-dependent, and re-runs her simulations. 11) Using HBP-COLL analysis tools, she easily compares results from homogeneous and spatially structured networks.					
SP6BSP-UC-021 Macrocircuit Model Combining Local with Macroscopic Connectivity	1) Abigail is interested in emerging activity when combining a model of the local cortical circuit with the recurrent network exhibited between cortical areas. 2) Using the search functions in the HBP-COLL, Abigail locates the network construction code for the local network from Use Case 1. She slightly modifies the code so that it can be used as building blocks for a larger model. 3) Abigail writes a new script connecting the building blocks using the macroscopic connectivity obtained from anatomical data. 4) Abigail instruments her network with stimulation and recording devices. 5) Using tools provided with NetSim, she specifies tests for the correctness of the network connectivity generated by NetSim and executes these on a	Referenced in listed Collab.	SP6BSP-FR-001, SP6BSP-FR-003, SP6BSP-FR-005, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU)

	downscaled version of the network. 6) She submits her simulation Task to the Brain Simulation Platform for execution on high-end HPC resources. 7) Once the simulation is complete, she uses HBP-COLL tools to investigate how the mean activities and power spectra in different areas depend on random spiking activity representing extrinsic input.					Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)
SP6BSP-UC-022 Verification of the Biological Relevance of Theoretical Prediction Obtained in the Large N Limit	1) Abigail has derived an expression for the correlation coefficient between the activities of pairs of neurons in a random network of binary neuron models in the limit of an infinite number of neurons. The analytical expression is nice and the relation between the quantities explains the observed small magnitude of the correlations. 2) Bill now wonders whether the terms surviving in the large N limit are really the dominating ones for networks of biologically relevant size. 3) Bill and Abigail team up to carry out simulations with varying N to obtain a graph showing the correlation coefficient as a function of network size. 4) Bill implements Abigail's model in a compact script using NetSim's high-level connectivity functions, making network size N an easily scaled parameter. 5) Bill runs simulations for a range of N values on a local cluster computer. 6) Bill and Abigail analyse the results using HBP-COLL facilities and find that in this case, the experimentally observed correlation coefficient indeed converges towards the theoretical result in the large-N limit.	Referenced in listed Collab.	SP6BSP-FR-001, SP6BSP-FR-003, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)
SP6BSP-UC-023 Robustness of	1) Abigail has a full-scale model of the cell-type specific in-vivo spiking activity in the local cortical	Referenced in listed Collab.	SP6BSP-FR-002, SP6BSP-FR-003,	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2

Network Activity with Respect to Neuron and Synapse Model	circuit and wants to port a downscaled version of the model to specific neuromorphic hardware. 2) Bill tells her that unfortunately the hardware implements different neuron and synapse models than those Abigail used in her original work; the hardware also imposes constraints on parameter values. 3) Therefore, Abigail needs to carry out simulations of her circuit with a neuron and a synapse model that are mathematically identical to those available in the hardware. 4) Abigail extends NetSim with a module providing a neuron and a synapse model implementing the models available in neuromorphic hardware. This requires some C++ programming, but thanks to the modular nature of NetSim, Abigail can add these models through a dynamically linked library. No modifications to NetSim proper are required. 5) Abigail now substitutes the original neuron and synapse models with those implementing the hardware models in her local circuit model and tunes parameter values by running simulations repeatedly, until the hardware-like model shows the same dynamics as the original. 6) Bill uses the parameters to configure and run network simulations on the neuromorphic hardware. 7) Abigail and Bill then compare spike trains obtained from the original NetSim model simulations, the hardware-like NetSim model simulations and the hardware using HBP-COLL tools. Where discrepancies occur, they clearly distinguish between those caused by differences between the original and the modified model, and those caused by differences between the real neuromorphic hardware and its mathematical		SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007			Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)

	description. 8) To understand some observed discrepancies better, Abigail re-runs NetSim simulations, recording membrane potentials from a small number of neurons in addition to spike trains.					
SP6BSP-UC-024 Verification of Independence of Simulated Network States from Time Discretisation	1) Abigail has a network model exhibiting some degree of synchrony in the activity of the neurons. She wonders whether this really is a property of the dynamics or an artefact of the numerical methods the network simulator employs constraining all spike times to an equidistant grid. 2) In addition to the neuron model Abigail used in her work, NetSim also provides a variant implementation that determines the exact time of threshold crossings in continuous time. 3) Abigail modifies her network model to use this implementation, and configures NetSim to handle all spike times in continuous time. 4) She re-runs her simulations using HPC facilities, as continuous-time simulations are more computationally demanding, and because she needs numerous independent realisations to obtain good statistics. 5) Abigail compares results from grid-based and continuous-time simulations using HBP-COLL tools.	Referenced in listed Collab.	SP6BSP-FR-001, SP6BSP-FR-003, SP6BSP-FR-005, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)
SP6BSP-UC-025 Flexibility in Specification of New Neuron and Spike-Time Dependent Plasticity (STDP)	1) Abigail finds an interesting new neuron model or STDP model in the literature and wants find out how her favourite network model behaves if her own neuron or STDP model is replaced by the newly published one. 2) Unfortunately the network simulator does not yet provide implementations of the new models. Abigail has programmed in C++ before but has no idea of	Referenced in listed Collab.	SP6BSP-FR-001, SP6BSP-FR-003, SP6BSP-FR-005, SP6BSP-FR-006, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH)

Models	parallelisation and the details of the simulation engine. Luckily the specification of a neuron model is well isolated from the parallelisation and communication parts of the software and basically just requires statements of the dynamical equations and the parameters involved. Thus Abigail takes an existing model as a template and manages the implementation of the new model well. 3) Later she finds that the author of the paper describing the new neuron model states that the reported results can be achieved reliably only with a specific numerical solver. She therefore creates a variant of her implementation of the new model using the specified solver. 4) Running simulations using her own models as well as the new model with both solvers, she can discern the effects of the model as well as the numerical solvers.					Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER (NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)
SP6BSP-UC-026 Detailed Investigation of the Correlation Structure of Neuronal Activity	1) Bill wants to investigate the detailed shapes of the time resolved cross-correlations exhibited by the spiking activity of pairs of neurons of different cell types in a large-scale network simulation. 2) Bill estimates that an incredible amount of data would have to be generated to obtain the histograms at the required accuracy. 3) He thus decides not to write out individual spikes but to compute the histograms already while the simulation is running, using the cross-correlation recording device provided by NetSim. 4) Bill runs his simulations, which write only compact correlation data to file, making data management and analysis feasible.	Referenced in listed Collab.	SP6BSP-FR-001, SP6BSP-FR-004, SP6BSP-FR-005, SP6BSP-FR-007	https://collab.humanbrainproject.eu/#/collab/161/nav/5307	TRL5	WP6.2 Inga BLUNDELL (JÜLICH) Hannah BOS (JÜLICH) Markus DIESMANN (JÜLICH) Jochen Martin EPPLER (JÜLICH) Espen HAGEN (JÜLICH) Moritz HELIAS (JÜLICH) Tammo IPPEN (JÜLICH) Susanne KUNKEL (JÜLICH) Abigail MORRISON (JÜLICH) Hans Ekkehard PLESSER

						(NMBU) Jannis SCHUECKER (JÜLICH) Johanna SENK (JÜLICH)

Annex D: Summary – Service IT Resource Planning

Table 5: Collaboratory Integration

Product/ Software Package/ Service	TRL	Data Storage Capacity used by this Product	Data Storage Capacity Allocated for this Product	Location(s) of Data Storage	Data Access Protocol(s)*	Compute Resource(s) Allocated	Location(s) of Compute Resource(s) Allocated	Compute Access Protocol(s)**
Subcellular Builder	3	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Single Cell Builder	5	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service/Amazon
Microcircuit Builder	4	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Mesocircuit/Macrocircuit Builder	3	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service

Experiment Builder	3	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Morphology Synthesizer	2	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Morphology Curation	5	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Simplification	2	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service
Mesocircuit Simulation	5	see respective tasks/foundation software (for dependencies see Annex B)	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Document Service/UNICORE	see respective tasks + respective HPC allocation	see respective tasks + respective HPC allocation	Task Service/UNICORE

WebSDA	8	>200 Gb	1840 Gb	HITS	Webserver	532 cores (shared 50% with other non-HBP-related services)	HITS	Webserver
NMC Portal	9	50GB	100GB	biotech datacenter@geneva	HTTP	N/A	N/A	N/A

Table 6: Services

Product/Software Package/Service	TRL	Data Storage Capacity used by this Product	Data Storage Capacity Allocated for this Product	Location(s) of Data Storage	Data Access Protocol(s)*	Compute Resource(s) Allocated	Location(s) of Compute Resource(s) Allocated	Compute Access Protocol (s)**
task_assign_metype	4	see brainbuilder	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see brainbuilder	1VM + respective HPC allocation	SLURM
task_assign_morphology	4	see brainbuilder	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see brainbuilder	1VM + respective HPC allocation	SLURM
task_assign_synapse_class	4	see brainbuilder	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see brainbuilder	1VM + respective HPC allocation	SLURM
task_axon_splicing_task	4	see AxonSplicing	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see AxonSplicing	1VM + respective HPC allocation	SLURM
task_brunel_delta_nest_task	4	see NEST	1TB (shared) + respective HPC	CEPH@Lugano + respective HPC	Document Service	see NEST	1VM + respective HPC	SLURM

			allocation	storage			allocation	
task_build_morphology_bundle	4	~ 1MB	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	serial	1VM + respective HPC allocation	SLURM
task_cell_densities_per_layer_from_mvd	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_cell_positioning	4	see brainbuilder	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see brainbuilder	1VM + respective HPC allocation	SLURM
task_cell_ratios_per_layer	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_createAavSynapses	4	see genbrain	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see genbrain	1VM + respective HPC allocation	SLURM
task_createCellPositions	4	see genbrain	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see genbrain	1VM + respective HPC allocation	SLURM
task_functionalize	4	see functionalizer	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see functionalizer	1VM + respective HPC allocation	SLURM
task_get_volumetric_data	4	see genbrain	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see genbrain	1VM + respective HPC allocation	SLURM
task_inhibitory_synapses_density	4	see Bluepy Validation	1TB (shared) + respective HPC	CEPH@Lugano + respective HPC	Document Service	see Bluepy Validation	1VM + respective HPC	SLURM

		toolkit	allocation	storage		toolkit	allocation	
task_intrinsic_extrinsic_synapse_densities_per_layer	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_intrinsic_inh_synapse_densities_per_layer	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_mesobuilder_touch_detection	4	see touchdetector	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see touchdetector	1VM + respective HPC allocation	SLURM
task_microcircuit_hpc_task	4	see NEST	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see NEST	1VM + respective HPC allocation	SLURM
task_microcircuit_task	4	see NEST	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see NEST	1VM + respective HPC allocation	SLURM
task_morphology_collage	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_morphology_repair_all	4	see morphology repair	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see morphology repair	1VM + respective HPC allocation	SLURM
task_morphology_synthesis	4	see morphsyn	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see morphsyn	1VM + respective HPC allocation	SLURM
task_morphology_validation	4	see neuroM Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see neuroM Validation toolkit	1VM + respective HPC allocation	SLURM

task_neuron_bundle_creator	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_neuron_task	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_plot_layers	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_plot_mosaic	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_plot_psth	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_plot_psth_mpld	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_run_remote_notebook	4	~1MB	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	serial	1VM + respective HPC allocation	SLURM
task_run_remote_opt_notebook	4	~1MB	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	serial	1VM + respective HPC allocation	SLURM
task_select_region	4	see genbrain	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see genbrain	1VM + respective HPC allocation	SLURM
task_simulation_launch_viz	4	see neuron Validation	1TB (shared) + respective HPC	CEPH@Lugano + respective HPC	Document Service	see neuron Validation	1VM + respective HPC	SLURM

		toolkit	allocation	storage		toolkit	allocation	
task_single_neuron_task	4	see NEST	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see NEST	1VM + respective HPC allocation	SLURM
task_somata_volume_fraction	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_spike_raster	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_spine_length	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_splicing_collage	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM
task_steps_sim_analysis	4	see steps Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see steps Validation toolkit	1VM + respective HPC allocation	SLURM
task_synapse_counts	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_synapses_overall_density	4	see Bluepy Validation toolkit	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy Validation toolkit	1VM + respective HPC allocation	SLURM
task_touch_detect	4	see touch detector Validation	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see touch detector Validation	1VM + respective HPC allocation	SLURM

		toolkit				toolkit		
task_tsodyks_depressing_task	4	see NEST	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see NEST	1VM + respective HPC allocation	SLURM
task_voltage_collage	4	see Bluepy	1TB (shared) + respective HPC allocation	CEPH@Lugano + respective HPC storage	Document Service	see Bluepy	1VM + respective HPC allocation	SLURM

Table 7: Foundation Software

Product/Software Package/Service	TRL	Data Storage Capacity used by this Product S = small (<1GB), M = medium (<10GB), L = large (<100GB), XL = extra large (<1TB), XXL (<10TB), XXXL (<100TB)	Data Storage Capacity Allocated for this Product***	Location(s) of Data Storage	Data Access Protocol(s) *	Compute Resource(s) Allocated (2)	Location(s) of Compute Resource(s) Allocated	Compute Access Protocol(s)**
AxonSplicing	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm
BBPSDK	6	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	Slurm/local
BGLibPy	5	source data	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	Slurm/local
BlueBuilder	5	L	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	BBIV@Lugano	Slurm
BluePy	5	source data	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramewrl/Slurm/local
BluePyOpt	3	S	<1PB	BBIV@Lugano	GPFS	1k-2k cores to cluster queue	cluster@ Lugano	TaskFramework/Slurm
BluePyOpt	3	S	<1GB	Amazon	outsourced to user	outsourced to user	Amazon	EC2
BlueRepairS	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to	cluster@	indirect

DK						cluster queue/local	Lugano	
BrainBuilder	4	S-M	<1PB	BBIV@Lugano	GPFS	1 core	cluster@ Lugano	TaskFramework/local
Brion	6	source data	<1PB	BBIV@Lugano	GPFS	indirect	cluster@ Lugano	indirect
CoreNeuron	6	S-XXXL	<1PB	BBIV@Lugano	GPFS	1k-64k core job to supercomputer queue	BBIV@ Lugano	TaskFramework/Slurm
eFEL	6	S	<1PB	BBIV@Lugano	GPFS	indirect	cluster@ Lugano	TaskFramework/Slurm/local
FLATIndex	4	M-L	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	Slurm
Functionalizer	5	L-XXL	<1PB	BBIV@Lugano	GPFS	1k-64k core job to supercomputer queue?	BBIV@Lugano	TaskFramework/Slurm
Livre	6	M-L	<1PB	BBIV@Lugano	GPFS	10-100 cores/1-10GPUs job to cluster queue	cluster@ Lugano	Slurm
Mesobuilder	4	L-XXL	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	Slurm
mod2c	5	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm

ModelManagement	4	S	<1PB	BBIV@Lugano	GPFS	1k-16k core job to supercomputer	cluster@ Lugano	TaskFramework/Slurm
Monsteer	6	N/A	<1PB	BBIV@Lugano	GPFS	indirect	cluster@ Lugano	indirect
Morphology Repair	4	S-M	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm
Morphsyn	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm
MUK	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	indirect
NEST	9	S-XXXL	<1PB	BBIV@Lugano	GPFS	1k-64k core job to supercomputer queue	BBIV@Lugano	TaskFramework/Slurm
NEST	9	S-XXXL	Requires separate allocation	JUQUEEN@JSC	GPFS	Requires separate allocation	JUQUEEN@JSC	UNICORE/Slurm
Neurodamus	5	S-XXXL	<1PB	BBIV@Lugano	GPFS	indirect	BBIV@Lugano	indirect
NeuroM	5	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm/local
NEURON	9	S-XXXL	<1PB	BBIV@Lugano	GPFS	1k-64k core job to supercomputer queue	BBIV@Lugano	TaskFramework/Slurm

NEURON	8	S-XXXL	Requires separate allocation	JUQUEEN@JSC	GPFS	Requires separate allocation	JUQUEEN@JSC	UNICORE/Slurm
NEURON	8	S-XXXL	Requires separate allocation	FERMI@CINECA	GPFS	Requires separate allocation	FERMI@ CINECA	UNICORE/Slurm
OptimizerFramework	6	S	<1PB	BBIV@Lugano	GPFS	1k-2k core job to supercomputer queue	BBIV@Lugano	slurm
PlacementHints	3	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	local
PostSimulationWorkflow	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm
pybinreports	4	source data	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	cluster@ Lugano	TaskFramework/Slurm
ReportingLib	5	S-XXXL	<1PB	BBIV@Lugano	GPFS	indirect	BBIV@Lugano	indirect
RTNeuron	6	M-L	<1PB	BBIV@Lugano	GPFS	10-100 cores/1-10GPUs job to cluster queue	cluster@ Lugano	Slurm
STEPS	8	S-L	<1PB	BBIV@Lugano	GPFS	1-100 core job to supercomputing queue	BBIV@Lugano	TaskFramework/Slurm

						(embarrassingly parallel)		
STEPS (parallel)	3	S-L	<1PB	BBIV@Lugano	GPFS	1-100 core job to supercomputing queue	Cluster@ Lugano?	TaskFramework/Slurm
TouchDetector	5	L-XXXL	<1PB	BBIV@Lugano	GPFS	8k-64k core job to supercomputer queue	BBIV@Lugano	TaskFramework/Slurm
Validation Toolkit	4	S	<1PB	BBIV@Lugano	GPFS	Serial job to cluster queue/local	Cluster@ Lugano?	indirect

* Data Access Protocols such as GPFS, N.FS, S3, Collab storage, etc.
** Compute Access Protocols such as EC2, Task Framework, Unicore, OCCI, Slurm, ssh, gLite, Condor, etc.
***Resources are shared between all users and all services unless otherwise indicated.

Annex E: Summary – Service Technology Readiness Levels (TRLs) Metrics

Documentation URL - User, Developer and/or Administrator documentation is available at this URL. Strong preference should be given for publicly available documentation services.

Target User Count (TRL6+) – Target user counts (concurrent service users).

SLA Defined – The software documentation defines some Quality of Service metrics in the service documentation. These metrics may or may not be enforced by the service itself. The service has not been tested to adhere to the documented QoS metrics.

SLA Monitored – The Quality of Service metrics are monitored by a monitoring service.

SLA Enforced – The Quality of Service metrics are enforced by implementing service. If the SLA Definition indicates on 3 API/request/sec/user, there are suitable mechanisms implemented in the service to ensure these limits are not exceeded.

Table 8: Technology Readiness Levels

Product/Software/ Package/service	Technology Readiness Level (TRL1-9)	Documentation URL	Target User Count (TRL6+)	SLA Defined (TRL7+)	SLA Monitored (TRL7+)	SLA Enforced (TRL7+)	Comments
WebSDA	8	http://mcm.h-its.org/webSDA	>250	http://mcm.h-its.org/webSDA/terms_and_conditions	Regular monitoring of server availablilty. Planned implementation of automated monitoring service provided by HITS sysadmins. Software updates checked against a set of test cases. Email address provided to users to report any problems they identify.		Target users count defined as number of unique IPs with data stored on server
NMC Portal	9	No SLA documentation	5000	Not available to end user	http://uptime.statuscake.com	No	System has been tested with more than 5000 users with Siege software
Bluepy service	5	No SLA documentation	N/A	N/A	N/A	N/A	
TaskManager	5	No SLA documentation	N/A	N/A	N/A	N/A	

Provenance Service	5	No SLA documentation	N/A	N/A	N/A	N/A	
Document Service	5	No SLA documentation	N/A	N/A	N/A	N/A	
Slurm	9	No SLA documentation					

Annex F: Backlog (Remaining bugs and new features to be added)

Single cell builder

Table 9: Remaining Bugs in the Single Cell Builder

Bug ID	Related Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
BSP-334	SPBSP-UC-008	Requested parameters should be optional.	https://bbpteam.epfl.ch/project/issues/browse/BSP-334	Some parameters like apical_point and circuit recipe should be optional.

Microircuit Builder

Table 10: Remaining Bugs in the Microcircuit Builder

Bug ID	Related Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
BSP-335	SPBSP-UC-012	Random seed should be managed for every steps	https://bbpteam.epfl.ch/project/issues/browse/BSP-335	Some steps do not take into account random seeds.

Table 11: Features to be added to the Microcircuit Builder

Feature ID	Necessary for Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
6.1.1	SPBSP-UC-011, 012	Code not packaged as a gnu-module.		Code is not packaged a gnu-module for standard installation on SP7 facilities
6.2.1	SPBSP-UC-014	A limited set of data are exportable		Only a predefined number of regions are exportable.

Subcellular builder

Table 12: Features to be added to the Subcellular Builder

Feature ID	Necessary for Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
6.2.1	SPBSP-UC-014	A limited set of data are exportable		Only a predefined number of regions are exportable.

Mesocircuit and macrocircuit builder

Table 13: Features to be added to the Mesocircuit and Macrocircuit Builder

Feature ID	Necessary for Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
6.1.1	SPBSP-UC-011, 012	Code not packaged as a gnu-module.		Code is not packaged a gnu-module for standard installation on SP7 facilities

CoreNeuron, NEURON, Neurodamus, ReportingLib

Table 14: Features to be added to CoreNeuron, Neurodamus, and ReportingLib

Feature ID	Necessary for Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
6.2.2	SPBSP-UC- 017	CoreNeuron is installed on BBPIV as a module. Its integration to HBP collab has not been tested by the collab/SP7 team.		Item not on development team but rather on the collab/SP7 team

6.2.3	SPBSP-UC-019	CoreNeuron is installed on BBPIV as a module. Its integration to HBP collab has not been tested by the collab/SP7 team.		Item not on development team but rather on the collab/SP7 team

STEPS & Parallel STEPS

Table 15: Features to be added to STEPS & Parallel STEPS

Feature ID	Necessary for Use Case(s) or FR(s)	Short Description	Bug tracker URL (if available)	Comments
5.4.3		Accurate measurement of memory footprint requirements with scaling up to 10k cores		Due to large code changes, memory footprint changed quite a lot. Since further modifications is required, this feature is not yet applicable.

Annex G: SP6 User Engagement Strategy

Introduction

Experimental knowledge and modelling in neuroscience are highly fragmented. For the former, a broad range of levels are studied and within a given level, classifications schemes, methodological heterogeneity, data quality, curation and standard representations are some of the common sources of controversy. For the latter, the level of detail to be modelled, and simplifying and ansatz assumptions are some of the common sources of controversy; a great many more models are produced than are actively maintained and refined due to, for example, a lack of widespread adoption of model development best-practices such as automated validation strategies and revision control.

A strategic objective of the HBP Ramp-Up Phase is to establish an end-to-end approach, from experiment to application, which integrates the community broadly, and will have a lasting impact on addressing the fragmentation of experimental knowledge and modelling. The development of the Brain Simulation Platform in SP6 is embedded in an integrated network of disciplines, domain specific contributions, and their associated platforms. The integration starts with experimental data (from SPs 1&2, and also the community), which is curated, organised and made accessible in neuroinformatics databases (SP5 Neuroinformatics Platform). While such databases are extensive, there are invariably gaps in our knowledge—unknowns which remain unexplored or experimentally inaccessible using current techniques. The SP6 Brain Simulation Platform provides model building pipelines to integrate the available sparse data into dense data-driven model representations using algorithms and predictive approaches, and frameworks for automated validation against experimental data which are extensive and community extensible. The development of a multiplicity of principled simplifications for *in silico* experimentation, exploratory and hypothesis-driven science, with validation against the data-driven reference is a joint task between SPs 4&6. These “derived but constrained” models are a key output of the Brain Simulation Platform, to drive a vibrant user community in the application domains of neuromorphic applications, neurorobotics, and cognitive neuroscience. Taken together, this end-to-end network is a “whole community model building” approach, where each domain of expertise contributes constraints and insights towards an integrated picture, and strong links through shared conceptual frameworks greatly facilitate the propagation of insights across the domains. Such an approach is able to bridge and unify experimental and modelling communities broadly and incrementally.

It is instructive to contrast this integrated network of interactions with the deceptively similar alternative: “whole community building models”. To a large extent, the latter is currently what we have in the field. Platform development in the HBP must focus on implementing the former and driving its adoption, if HBP platforms are to achieve their strategic objective of having a lasting integrative effect on neuroscience.

An important aspect of SP6 user engagement is the fostering of communities which engage in the data-integration, building, refinement, and validation process, and also the communities which adopt such models for *in silico* experimentation, exploratory and hypothesis-driven science. We will refer to these as model development and model adoption communities respectively.

The proposed whole community model building approach can be viewed as a “disruptive technology” and therefore the technology adoption curve applies (see Figure H1). HBP Platforms are currently in the “pre-chasm” phase, whereby prevalent wisdom suggests development should prioritise implementing for early adopters, success stories and delivering value propositions, rather than focusing on convenience and responding to the needs of critics and sceptics.

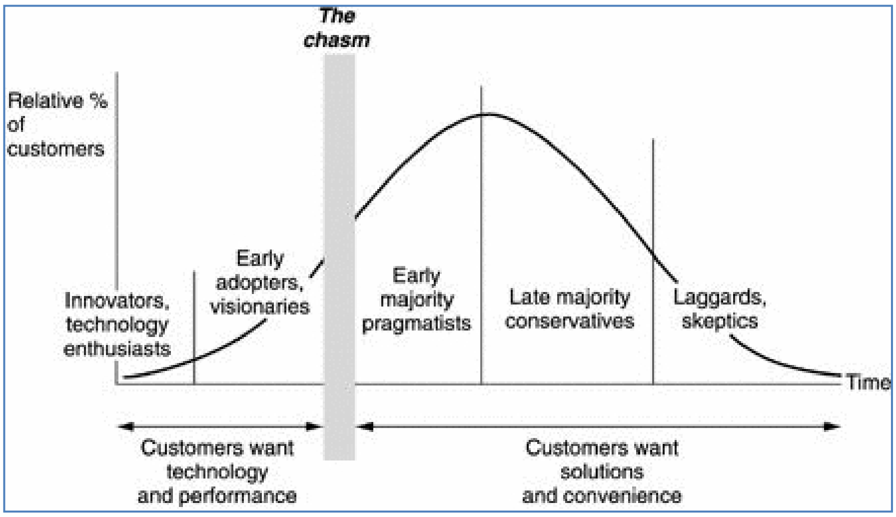

Figure 2: The Technology Adoption Curve [1].

With this in mind, the SP6 user engagement strategy during the Ramp-Up Phase focused on attracting and catering to early adopters, finding synergies and fostering early success stories, both inside and outside the HBP. SP6 invested heavily in establishing and fostering model development communities focused through a co-design process which paired software development with scientist drivers for key representative use cases. As these communities ramp-up to produce compelling data-driven reference models, the availability of these models and their simplified derivatives will attract model adoption communities.

Initiating model development communities

The SP6 Platform development team adopted a co-design approach for model development use cases on multiple scales: molecular level, sub-cellular level, single neuron level, microcircuit level, and brain region and whole brain levels.

Development of these model use cases began as a *scaffold model*. By analogy to building construction, scaffold models provide a structure and foundation to bootstrap a comprehensive data integration and model building pipeline while individual components are incomplete. Described in detail elsewhere (see Deliverable D6.7.1), the general architecture of the scaffold model is to decouple:

- Data collection, curation and integration
- Development of modelling pipelines at various levels
- Validation.

Since a model is never proven right, it is only proven wrong, a model is never “done”. Scaffold models should employ a middle-out approach from a given level of data, establish an iterative process for refinement, and integrate with models at other levels, on decoupled refinement cycles. The existence of bootstrapped scaffold models allows the development of community contributed validation suites to establish and continuously assess the domain of validity of the model, and simplifications thereof. They resulting transparency and existence of an ecosystem of simplifications will drive adoption communities.

As scaffold models evolve and mature, and vibrant model development and adoption communities emerge, model development will transition from the *scaffold model* to the *community model* phase, where model development and refinement cycles are driven by the community.

The first of such transitions will need to be supported by newly-developed Platform functionalities supporting collaborative planning and community-based decision making for regular model releases and refinement roadmaps, history of validations to track regression and provide transparency of evolution of reference models and simplifications, ticket/task systems, wikis, agile estimation tools, discussion forums, comment streams, etc. Such features will be prioritised in SGA1 in a co-design process with drivers of the Hippocampus model, which will pilot the *community model* phase.

Co-Design Use Cases

Specific co-design use cases undertaken in the SP6 Ramp-Up Phase are described in the following.

Scaffold models development - Cellular level

Basal Ganglia (KI)

The data-driven building of the striatal microcircuit was initiated in the SP6 Ramp-Up. The suggested workflow, from single neuron reconstructions to local neural network with central afferents, originally developed for the Neocortex, proved to be fully applicable in this case, requiring only a few simplifications related to the lack of columnar organisation of the striatal network. Difficulties on the way were largely caused by missing experimental data, and not by specific aspects of the workflow. The KI research team learned a lot from the collaboration, starting with pyNEURON scripting and proceeding with eFEL for feature extraction from experimental data, NeuroM for morphometrics, and OptimizerFramework (and newly BluePyOpt) for single-cell fitting and sensitivity analysis. Building a structurally constrained local network is in progress hindered partly by the lack of information about synaptic densities. Available statistical data, which was traditionally used to establish connections, is no longer helpful in the adopted workflow, but will be used for validation instead. Many insights are expected because the striatal circuit has never been built on such a detailed level before.

Cerebellum (UNIPV)

The data-driven building of the cerebellar microcircuit has undergone remarkable progress during the SP6 Ramp-Up Phase. Most of the cerebellar neurons have been modelled at a high level of detail and many of them are already used by the scientific community at large. An initial cerebellar cortical network model has been extended and updated to account for the most recent discoveries. The neuron and network cerebellar models have been extended, simplified and incorporated into closed-loop robotic simulations. The simplified models have been translated into accelerated versions running in real-time in physical robots. All these models are now being adapted to the HBP workflow. This procedure includes single neuron reconstruction from realistic morphologies, morphology cloning, automatic parameter optimisation and local neural network reconstruction following morphology/density/proximity rules. Adaptations to the original modelling schemes developed for the neocortex are needed, for example to account for the extremely complex electrotonic structure and electroresponsive properties of Purkinje cells and for the quasi crystalline geometrical organisation of the cerebellar cortical network, which differs from the neocortical microcolumn. The UNIPV research team is closely collaborating with other HBP teams for pyNEURON implementation and development, for eFEL scripting of feature extraction from experimental data, on NeuroM for morphometrics, and on OptimizerFramework (and newly BluePyOpt) for single-cell fitting and sensitivity analysis. Implementing codes suitable for specific properties of the cerebellar network is key to generalsing the aforementioned procedures and software applications. Strategic missing information on neuronal morphological and excitable properties, on synaptic transmission and plasticity, and on network connectivity will be searched for through specific experiments in SP1 and through worldwide collaborations. This will require the creation of a collaborative cerebellar team through Co-Design Project 2.

Hippocampus (UCL and EPFL lead)

Work on the hippocampus scaffold model proceeded relatively rapidly in the Ramp-Up Phase due to the immediate applicability of the neocortical workflow for hippocampus, interactions with a vibrant hippocampal modelling community, and the availability of relatively high-quality data. During the Ramp-Up Phase, the hippocampus modelling effort already began the process of bootstrapping a *community model* phase (see below), which will become the focus of the hippocampus work in SGA1. The availability of collaborative development hippocampus models, transparent and automated validation and benchmarking suites, and simplification pipelines were identified as key incentives to drive community involvement in both model development and model adoption.

Scaffold model development - Sub-cellular level

We have constructed a molecular database that seeks to integrate knowledge about reactions and concentrations for the purpose of constructing simulations. Other databases, such as the Biomodels database, serve as repositories for hundreds of individual standalone models. We seek to combine these models into a unified network. The models present in the Biomodels database can be used to populate the simulation database, but the process requires user involvement to establish the mapping between protein species in individual models and the actual proteins present in Uniprot and other databases. We have pioneered an approach in which students at external universities take a semester-long project class under our tutelage. Each student picks a particular Biomodels model and maps it to a more universal format. In doing so, the student learns more about the biological signalling pathways and the simulation database is augmented in the process, a net gain for all concerned. Weekly sessions are held remotely via Skype. As there are many models in the Biomodels database, we hope to expand the programme to encompass more students. This expansion will be carried out through notices on the HBP website and with the assistance of the HBP Education outreach staff.

Scaffold models development - Molecular

During the Ramp-Up Phase WP6.3 worked closely with Task 6.4.1 on using different molecular modelling approaches to aid the building of subcellular kinetic models of receptor induced cascades, by both qualitatively and quantitatively predicting network parameters. As a test case, we used a model framework where the cAMP – PKA cascade plays an important role. This model framework has investigated interactions via Golf and Gi/o coupled GPCRs onto adenylyl cyclase (AC) type 5, which are not sufficiently characterised by experimental means. Specifically, we estimated the association rate constants using Brownian dynamics and the allosteric effects of G-protein binding on the mechanics of AC5 using coarse-grained modelling. Furthermore, hybrid QM/MM simulations investigated the catalytic activity of the AC5 enzyme. This research demonstrated the potential of the synergistic effort of the molecular modelling and systems biology communities in advancing the understanding of signalling networks, and constitutes a first step towards the inclusion of the effects of chemical factors (such the role of disease-linked mutations and the presence of drugs) for neuromodulation.

Gathering wider community input

During the Ramp-Up Phase, a significant effort was devoted to gathering community input and feedback on the proposed strategy and roadmap. Many of these activities were undertaken in collaboration with other SPs, including SP4, SP9, SP1, SP12, and SP7.

- EITN workshop: "Are we building the right thing? Requirements from theory for simulation environments and neuromorphic computing", 2–4 March 2015, http://eitnconf-020315.sciencesconf.org/ ; Organisers: A.P. DAVISION, J. MULLER, J. EPPLER, D. LESTER, A. MORRISON, M. DIESMANN.
- "HBP Hippocamp CA1: Collaborative and Integrative Modeling of Hippocampal Area CA1", 31 March–1 April 2015, UCL, London, UK http://neuralensemble.org/meetings/HippocampCA1/; Organisers: J. FALCK, S. KALI, A. MERCER, E. MULLER, A. ROMANI, A. THOMSON.
  - SfN-2015 follow-up informal meeting with Hippocampus participants, and other hippocampus modellers at SfN.
  - HBP Community forum (https://forum.humanbrainproject.eu/) was launched as a result of need identified in the Hippocamp meeting.
- HBP CodeJam workshop #7, 11–14 January 2016, Manchester, UK, http://neuralensemble.org/meetings/CodeJam7/; Organisers: A.P. DAVISON, D. LESTER, A. MORRISON, E. MULLER, J. MULLER.
  - An "early adopters and technology drivers workshop",
  - Included participation from the non-HBP community, such as four scientists from the Allen Institute
  - Discussions with the Blue Brain Project, the Allen Institute, and NeuroML for complementary, comprehensive, open performance data model: D. FENG, S. GRATIY, P. GLEESON, W. VAN GEIT, A. DAVISON, E. MULLER, C. EROE, and others.
- NEST User Workshop, April 2015, at HBP headquarters, Geneva, CH, http://www.nest-initiative.org/nestactivity/first-nest-user-workshop/
- "Building a Neuroscience Community: community modelling and data repositories", Fondation Brocher, Geneva, 11–13 June 2015, http://www.kcl.ac.uk/sspp/departments/sshm/newsrecords/Foresight-Lab-Workshop-at-Brocher.aspx ; Organisers: N. ROSE, C. AICARDI, M. REINSBOROUGH, T. MAHFOUD, P. BELLO (SP12), A.P. DAVISON (SP9), J. MULLER (SP6).
- NESTML Community Workshop, 7-8 December 2015, Juelich, DE, http://www.fz-juelich.de/ias/jsc/EN/Expertise/SimLab/slns/news_events/2015/nestml-ws/Overview/_node.html
- OpenSourceBrain2015 meeting participation, 12–14 May, Alghero, IT, http://opensourcebrain.org/docs/Help/Meetings#OSB_2015 ; Organisers: P. GLEESON, A. SILVER, B. MARIN, S. CROOK, S. SOLINAS, P. ENRICO.
  - Talk: Eilif MULLER, "Approaches for reducing fragmentation in the computational modelling community"
  - Talk: Armando ROMANI, "Data-driven Hippocampus CA1 Modelling in the Human Brain Project"
  - Talk: Michele MIGLIORE (SP1 & SP6-SGA1), "Computational properties of CA1 pyramidal neurons"
  - Talk: Oren AMSALEM (HUJI, SP6), "Incorporating Gap Junctions in the Human Brain Project *in silico* Microcircuitry"
  - Talk: Giogio ASCOLI (Hippocamp community participant), "Hippocampome.org - Neuron classification for real-scale hippocampal modelling"

- CNS*2015 Workshop: Open collaboration in computational neuroscience, July 23 2015, Prague, Czech Republic, http://opensourcebrain.org/docs/Help/Meetings#CNS_2015 ; Organizers: P. GLEESON (UCL), A. LAZAR (Columbia)
  - Talk: Eilif MULLER, "Human Brain Project resources for the integrative modelling community"
  - Talk: Victor JIRSA (SP4), "The Virtual Brain"

Training and dissemination activities

- Paper: MARKRAM *et al.* [2]
- Partner EPFL-BBP released the neural microcircuit model collaboration portal (NMC portal), https://bbp.epfl.ch/nmc-portal , disseminating models, tools of MARKRAM *et al.* [1] in an interactive manner.
- webSDA Brownian dynamics webserver release (http://mcm.h-its.org/webSDA/documentation) and Yu et al. [3] and Martinez et al. [4].
- SfN presence @ HBP booth
  - 2013 - Alex THOMSON
  - 2014 - Eilif MULLER, Jeff MULLER, Srikanth RAMASWAMY
  - 2015 - Jeff MULLER, Srikanth RAMASWAMY
  - Booth presentation of NEST and elephant in the HBP Collaboratory; Michael DENKER explained how working in integrative loops unlocks the potential of the HBP approach.
- 2nd Human Brain Project School – Future Computing. Tom TETZLAFF gave two talks: "Intro neuron models" and "Simulation and numerics of neuron models". Markus DIESMANN gave a lecture about the "Simulation of networks". Johanna SENK gave a tutorial about "Simulating large-scale spiking neuronal networks with NEST".
- CNS*2015: NEST Tutorials and Workshops:
  - Interfaces in Computational Neuroscience Software: Combined use of the tools NEST, CSA and MUSIC. Jochen EPPLER, Jan MORÉN, Mikael DJURFELDT http://www.cnsorg.org/cns-2015-tutorials#t6
  - Modelling and analysis of extracellular potentials. Gaute EINEVOLL, Szymon ŁĘSKI, Espen HAGEN. http://www.cnsorg.org/cns-2015-tutorials#t2
- CNS*2015 Workshop: High-performance computing in neuroscience – from physiologically realistic neurons to full-scale brain models. Wolfram SCHENCK, Alex PEYSER. http://www.fz-juelich.de/ias/jsc/EN/Expertise/SimLab/slns/news_events/2015/HPCN_workshop_CNS2015/_node.html
- Okinawa Computational Neuroscience Course 2015. NEST is one of the software packages used for the student projects at the Okinawa Computational Neuroscience Course 2015. Sacha van Albada participates as a tutor.
- BCCN Conference Dynamics of Neuronal Systems. Susanne KUNKEL talks about "NEST: A highly scalable tool for simulations of spiking neuronal networks with synaptic plasticity".

- Alex THOMSON, Lecture to science students at Haberdasher's School, UK, March 2015.
- Alex THOMSON, Invited lecture about HBP, British Psychological Society, April 2014.

Upcoming dissemination activities

- HBP Young Researchers Event, Budapest, will include a short course on neuron modelling using BluePyOpt (W. VAN GEIT), and present the neocortical microcircuit modelling pipeline and NMC portal (S. RAMASWAMY).
- FENS 2016 technical workshop "Introduction to the HBP Collaboratory" with a focus on neuroscience users and use cases.
- Allen-HBP FENS 2016 satellite workshop will include a short course on neuron modelling using BluePyOpt (W. VAN GEIT).
- Janelia open data-driven modeling workshop, September 2016 (E. MULLER and A.P. DAVISON will be speaking and participating).

Documentation

https://developer.humanbrainproject.eu/docs/

http://mcm.h-its.org/webSDA/documentation

Note: For OSS software, much of the documentation is included in the tools themselves (see below).

Open-sourcing, or contributions to open software in HBP

Activities in SP6 have developed or contributed significantly to the development of open source software for neuroscience, including:

- STEPS stochastic engine for pathway simulation http://steps.sourceforge.net/STEPS/default.php
- BluePyOpt optimization of neuroscience models to data https://github.com/BlueBrain/BluePyOpt, preprint: http://arxiv.org/abs/1603.00500
- NEURON https://www.neuron.yale.edu
- coreNEURON http://www.fz-juelich.de/ias/jsc/EN/Expertise/High-Q-Club/CoreNeuron/_node.html
- NEST www.nest-initiative.org
- eFEL – electrphysiological feature extraction library https://github.com/BlueBrain/eFEL
- NeuroM - used to collect morphometric data for selection of reconstructions and validation of synthetic morphologies (Basal Ganglia), http://neurom.readthedocs.org/en/latest/index.html
- http://mcm.h-its.org/webSDA - Webserver for Simulation of Diffusional Association (SDA): making life much easier to simulate macromolecular diffusion.

References

[1] G. Moore. *Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers*. New York: HarperBusiness, 1991. Print.

[2] H. Markram, E. Muller, S. Ramaswamy, Michael W. Reimann, M. Abdellah, Carlos A. Sanchez, A. Ailamaki, L. Alonso-Nanclares, N. Antille, S. Arsever, Guy Antoine A. Kahou, Thomas K. Berger, A. Bilgili, N. Buncic, A. Chalimourda, G. Chindemi, J.-D. Courcol, F. Delalondre, V. Delattre, S. Druckmann, R. Dumusc, J. Dynes, S. Eilemann, E. Gal, Michael E. Gevaert, J.-P. Ghobril, A. Gidon, Joe W. Graham, A. Gupta, V. Haenel, E. Hay, T. Heinis, Juan B. Hernando, M. Hines, L. Kanari, D. Keller, J. Kenyon, G. Khazen, Y. Kim, James G. King, Z. Kisvarday, P. Kumbhar, S. Lasserre, J.-V. Le Bé, Bruno R.C. Magalhães, A. Merchán-Pérez, J. Meystre, Benjamin R. Morrice, J. Muller, A. Muñoz-Céspedes, S. Muralidhar, K. Muthurasa, D. Nachbaur, Taylor H. Newton, M. Nolte, A. Ovcharenko, J. Palacios, L. Pastor, R. Perin, R. Ranjan, I. Riachi, J.-R. Rodríguez, Juan L. Riquelme, C. Rössert, K. Sfyrakis, Y. Shi, Julian C. Shillcock, G. Silberberg, R. Silva, F. Tauheed, M. Telefont, M. Toledo-Rodriguez, T. Tränkler, W. Van Geit, Jafet V. Díaz, R. Walker, Y. Wang, Stefano M. Zaninetta, J. DeFelipe, Sean L. Hill, I. Segev, and F. Schürmann. *Reconstruction and Simulation of Neocortical Microcircuitry*. **Cell 163, 2015, 456-492.**

[3] X. Yu, M. Martinez, A.L. Gable, J.C. Fuller, N.J. Bruce, S. Richter, W.C. Wade. *webSDA: a web server to simulate macromolecular diffusional association*. **Nucleic acids research. 2015 Jul 1;43(W1):W220-4.**

[4] M. Martinez, N.J. Bruce, J. Romanowska, D.B. Kokh, M. Ozboyaci, X. Yu, M.A. Öztürk, S. Richter, R.C. Wade. **SDA 7: A modular and parallel implementation of the simulation of diffusional association software.** Journal of Computational Chemistry. 2015 Aug 5;36(21):1631-45.

Annex H: IPR Status, Ownership and Innovation Potential

Table 16: IPR Status, Ownership and Innovation Potential

Product / Software Package / Service	IPR Status*	Owner(s)	Non-HPP users**	Innovation Potential***
AxonSplicing	Copyright	EPFL/BBP	-	
BBPSDK	Copyright	EPFL/BBP	1)	
BGLibPy	Copyright	EPFL/BBP	-	
BlueBuilder	Copyright	EPFL/BBP	-	
BluePy	Copyright	EPFL/BBP	Allen Institute for Brain Science	
BluePyOpt	Open source	(EPFL/BBP)	N/A	
BlueRepairSDK	Copyright	EPFL/BBP	-	
BrainBuilder	Copyright	EPFL/BBP	-	
Brion	Open source	(EPFL/BBP)	N/A	
CoreNeuron	Open source	(EPFL/BBP, Michael Hines)	CRAY, IBM/Nvidia (as part of HBP PCP)	
eFEL	Open source	(EPFL/BBP)	N/A	
FLATIndex	Copyright	EPFL/BBP & EPFL/DIAS	-	
Functionalizer	Copyright, patent application pending (pre HBP)	EPFL/BBP	-	
Livre	Open source	(multiple - see repository)	N/A	
Mesobuilder	Copyright	EPFL/BBP	-	
mod2c	Open source	(EPFL/BBP, Michael Hines)	N/A	
ModelManagement	Copyright	EPFL/BBP	-	
Monsteer	Open source	(EPFL/BBP)	N/A	
MorphologyRepair	Copyright	EPFL/BBP	-	
MorphSyn	Copyright	EPFL/BBP	-	

MUK	Copyright	EPFL/BBP	-	
NEST	Open source	(Nest Initiative)	N/A	
neurodamus	Copyright	EPFL/BBP	1)	
NeuroM	Open source	(EPFL/BBP)	N/A	
NEURON	Open source	(Michael Hines)	N/A	
OptimizerFramework	Copyright	EPFL/BBP	Allen Institute for Brain Science	
PlacementHints	Copyright	EPFL/BBP	-	
PostSimulationWorkflow	Copyright	EPFL/BBP	-	
pybinreports	Copyright	EPFL/BBP	-	
ReportingLib	Copyright	EPFL/BBP	-	
RTNeuron	Joint copyright	EPFL/BBP & UPM	1)	
STEPS	Open source	(OIST, Uni Antwerp, EPFL/BBP)	N/A	
STEPS (parallel)	Open source	(OIST, Uni Antwerp, EPFL/BBP)	N/A	
TouchDetector	Copyright	EPFL/BBP and other	2)	
Validation Toolkit	Copyright	EPFL/BBP	-	
webSDA	Copyright	HITS gGmbh	Available for non-profit use	Biomolecular simulation

* IPR Status: Open Source, Copyright, Patent, Trade Secret, pre-IPR (i.e. you intend to obtain some form of IPR in the future)

** If this product/software package/service is currently being used outside HBP (e.g. donated, loaned, licensed, sold), please specify by whom.

*** Innovation Potential: Potential practical applications beyond HBP, commercial and/or non-commercial.

1) license agreements exist independent from and predating HBP

2) can be disclosed upon request