Grant Number:	720270	Grant Title:	Human Brain Project SGA1

Deliverable Title:	SP7 D7.6.3 (D42.3, D46) SP7 High-Performance Analytics and Computing Platform - Results for SGA1 Period 2
Contractual Number and type:	SGA1 D7.6.3 (D42.3 D46) - Demonstrator
Dissemination Level:	PUBLIC
Version / Date:	Resubmitted 17 July 2018; Accepted 1 October 2018

Abstract:

This deliverable is the annual compound of HBP deliveries and results (outputs and outcomes) from Sub-Project SP7 – High Performance Analytics and Computing Platform. The live complete catalogue of HBP deliveries is accessible on-line from the HBP portal.

The main deliveries from April 2017 to March 2018 have been:

- Advances in simulation technology for Petascale systems as well as in coupling simulators to enable multi-scale simulations
- Data-intensive supercomputing: enabling the use of new storage technologies
- Development of new and enhancement of existing tools and methods for interactive visualisation and visual data analysis
- Major progress in the development of holistic dynamic resource management mechanisms implemented at different scheduling levels
- Deployment of new services as part of the High Performance Analytics and Computing Platform, in particular to enable the other HBP Platforms to run their services based on the HPAC Platform

Keywords: High Performance Analytics and Computing (HPAC) Platform, High Performance Computing (HPC), Platform, research infrastructure, simulation technology, data-intensive supercomputing, interactive visualisation, dynamic resource management, key results

The High Performance Analytics and Computing Platform achieved major progress in providing the base infrastructure for the Human Brain Project during the last year. New services have been developed and made available in close co-design with the other HBP Subprojects that enable the other Platforms, in particular the Neuroinformatics Platform, the Brain Simulation Platform and the Neurorobotics Platform, as well as the Collaboratory, to run their services on the HPAC infrastructure (see section 2.5 for more details).

Moreover, the visibility of the HPAC Platform, and of the HBP as a whole, has significantly increased in the High-Performance Computing community. One of the highlights of last year was the choice of the HBP as one of the two project examples for the #HPCconnects campaign of the Supercomputing Conference 2017 (SC17). The picture above was taken during a shooting for a movie on the importance of HPC for the HBP and neuroscience, it was shown during the SC17 opening event.

Targeted users/readers	Computational neuroscience community, neuroimaging community, HPC community
Contributing Work-Package(s):	SGA1 WPs 7.1, 7.2, 7.3, 7.4, 7.5, 7.6
Initially Planned Delivery Date:	SGA1 M24 / 31 Mar 2018

Authors:	Thomas LIPPERT, JUELICH (P20), SP Leader Thomas SCHULTHESS, ETHZ (P18), SP Leader Anna LÜHRS, JUELICH (P20), SP Manager
Compiling Editors:	Anna LÜHRS, JUELICH (P20), SP Manager
Contributors:	Julita CORBALÁN, BSC (P5) Benjamin CUMMING, ETHZ (P18) Timo DICKSCHEID, JUELICH (P20) Markus DIESMANN, JUELICH (P20) Cyrille FAVREAU, EPFL (P1) Andreas FROMMER, BUW (P7) Cristian MEZZANOTTE, ETHZ (P18) Roberto MUCCI, CINECA (P13) Ralph NIEDERBERGER, JUELICH (P20) Boris ORTH, JUELICH (P20) Dirk PLEITER, JUELICH (P20) Luis PASTOR, URJC (P69) Hans Ekkehard PLESSER, NMBU (P44) Raül SIRVENT, BSC (P5) Bernd SCHULLER, JUELICH (P20) Dennis TERHORST, JUELICH (P20) David VICENTE, BSC (P5) Thomas VIERJAHN, RWTH (P46) Benjamin WEYERS, RWTH (P46) Gabriel WITTUM, UFRA (P35) Felix WOLF, TUDA (P85)
SciTechCoord Review:	EPFL (P1): Jeff MULLER, Martin TELEFONT, Marie-Elisabeth COLIN UHEI (P47): Martina SCHMALHOLZ, Sabine SCHNEIDER
Editorial Review:	EPFL (P1): Annemieke MICHELS

Table of Contents

List of Figures

List of Tables

1. Introduction

The mission of the High Performance Analytics and Computing (HPAC) Platform is to build, integrate and operate the base infrastructure for the Human Brain Project, together with Fenix, a project that is about to start. The infrastructure comprises hardware and software components, which are required to run large-scale, data-intensive, interactive simulations, to manage large amounts of data and to implement and manage complex workflows comprising concurrent simulation, data analysis and visualisation workloads.

During the last year, major progress has been made in all areas.

The HPAC Platform has been extended with a set of new features, functionalities and services. The most important achievement is that the Neuroinformatics Platform, the Brain Simulation Platform, the Neurorobotics Platform and the Collaboratory can run (part of) their services on HPAC infrastructure, which also ensures a close infrastructural link between these other HBP Platforms, so that these can offer efficient, fast and scalable services to their user communities. Key new elements are the now operational OpenStack service integrated with the HPAC authentication and authorisation infrastructure, a flexible tool for creating on-demand virtual computing infrastructure, the deployment of high-performance data transfer services between HPAC sites, the availability of Object Storage and the ability to mount HPC storage into Jupyter notebooks running in the Collaboratory.

The HPAC Platform experts also develop new technologies that will be integrated into the Platform, once they are mature. To ensure that all developments serve the requirements of the neuroscience community, they are co-developed with potential future users, who get access to the new services early during development.

Data-intensive applications, like the analysis of data from experimental facilities such as brain image scanners or simulations, play an increasingly important role. Ongoing development efforts focus on utilisation and management of hierarchical storage architectures, as well as data stores exploiting novel dense memory-based storage devices for such applications. Furthermore, software for coupling simulations and data processing pipelines, e.g. visualisation pipelines, has been developed.

Supercomputers are typically operated in batch mode, i.e. users submit their jobs to a job scheduler that tries to maximise the overall system usage. To reach a higher level of interactivity, for example to start an *ad hoc* visualisation during a simulation run, to have a look at the evolvement of the simulated network, this visualisation job needs to get scheduled instantaneously. A solution is to have a more dynamic resource management. This has to be implemented at different levels in the HPC system, as well as at the application level. The simulators NEST and Neuron have been prepared to support such resource sharing, and the job schedulers have been enhanced for this new type of resource management.

We also made important progress on the development of application-level software that needs to be closely aligned with the infrastructure evolution. The NEST simulator now supports rate models of neuronal networks, the data structures have been revised to better exploit the architecture of modern and future supercomputer architectures, and critical bottlenecks in the construction of brain-scale networks were eliminated. NESTML, a domain-specific language for the specification of neuron models, has been advanced. MUSIC has been used to couple NEST and UG4 to support multi-scale simulations. The Arbor library has also been enhanced; the focus has been on optimising the kernels for target architectures such as Intel KNL or NVIDIA GPUs, which are more and more often available as part of large-scale HPC systems.

Many developments in the HBP yield data sets that are too large to open and analyse with standard viewers or visual analysis tools, be it ultra-high resolution imaging data or the results of large-scale simulations. Therefore, major efforts have been spent on the development of interactive visualisation and visual analysis tools that are well integrated within the HPAC infrastructure, to make such large data sets usable for visual analysis. Some of the tools can also be coupled to give users different data views at the same time to ease analyses.

2. Results

2.1 Simulation technology for Petascale systems: concepts, numerical algorithms and software technology

Building on over 20 years of experience in neuronal network simulation technology, we have taken crucial steps to prepare NEST, The Neural Simulation Tool, for brain-scale simulations on future exascale high-performance computing systems. We focused on extending NEST capabilities, based on requirements of users inside and outside the HBP, on multi-scale runtime integration with other simulation tools, on facilitating the addition of new neuron models, and, in close collaboration with colleagues from the HBP Brain Simulation Platform, on preparing network creation and communication architecture of the simulation kernel for future exascale systems. Specifically, we

- extended NEST to support rate models of neuronal networks, thus permitting the investigation of network dynamics on the scale of spiking neurons, the classic domain of NEST, and the more abstract scale of rate models;
- integrated the NEST and UG4 simulators via the MUSIC library to support multi-scale simulations coupling network dynamics in NEST with detailed solutions to the 3D cable equations in UG4;
- developed NESTML as a domain-specific language for the specification of neuron models and implemented a tool generating optimized C++ code to add neuron models specified in NESTML to the NEST simulator;
- eliminated critical bottlenecks in the construction of brain-scale networks requiring a very large number of parallel processes;
- analysed user requirements for connectivity generation, drafted a more flexible and powerful user interface for neuron and connection instantiation;
- collaborated with the Brain Simulation Platform on revised data structures for connection representation and communication patterns, to allow NEST to exploit the full power of future exascale architectures;
- systematically reviewed all new code contributions to the NEST simulator in formalised, continuous integration based review processes in collaboration with the NEST developer community.

We integrated our activities closely in the NEST user community within and outside the HBP by means of

- systematic monitoring and follow-up of user requests and proposals through the NEST mailing list and issue tracker;
- regular open NEST developer video conferences (every second week);
- personal contact with and support to key NEST users in the HBP, especially in the Human Brain Organization, Systems and Cognitive Neuroscience, and Theoretical Neuroscience Subprojects;
- organisation of an annual NEST Conference (before the 2017 NEST User Workshop), in collaboration with the NEST Initiative, bringing together users and developers for intense exchanges of success stories, challenges and ideas.

2.1.1 Achieved Impact

- NEST is a well-established simulation tool for large neuronal networks, with users from inside and outside the HBP
- Newly introduced abilities, e.g. the ability to simulate rate-based models with NEST, have been received eagerly by the community and have already led to publications currently under review:
  - Jordan *et al*. 2017 https://arxiv.org/pdf/1709.05650.pdf
  - Senk *et al*. 2018, https://arxiv.org/pdf/1801.06046.pdf
  - Senden *et al*., 2018 https://github.com/ReScience/ReScience-submission/pull/46
- Code generation using NESTML has significantly reduced the burden of porting existing network models to NEST, and thus facilitates model porting efforts in Systems and Cognitive Neuroscience, as well as outside the HBP.
- The close interaction between NESTML developers in the High Performance Analytics and Computing Platform and NESTML users in Systems and Cognitive Neuroscience has driven the development of NESTML further and ensured that NESTML addresses the needs of computational neuroscientists.

Figure 1: Participants of the NEST Conference 2017

2.1.2 Component Dependencies

Component ID	Component Name	HBP Internal	Comment
510	Continuous dynamics code in NEST	No	Provides support for rate-based models in NEST

567	Report on numerical techniques for stochastic equations of population dynamics	No	Mathematical and algorithmic foundations of rate-based modelling in NEST
512	Simulator-simulator interfaces	No	Provides support for integrating NEST and UG4 simulations
520	Report on multi-scale challenges in the HBP simulator hierarchy	Yes	Background analysis to prepare simulator-simulator interface
519	Report on neuronal modelling languages and corresponding code generators	No (once paper is out)	Fundamentals of NESTML language and code generator
209	NEST – The Neuronal Simulation Tool	No	Key software component of the HBP for simulation of networks of simplified neurons
518	Community contacts established related to NEST	No	Ensure good contacts and information flow between NEST users and developers, both within and beyond the HBP
622	NEST Requirements Management	No	Provide a single contact point for handling user requirements and forwarding them to developers.
661	NEST Support for Providers	No	Support system operators in installing and providing NEST to users.
660	NEST Support for Modellers	No	Support users in using NEST, especially in porting existing models to NEST.

2.2 Data-intensive supercomputing technology

Data-intensive applications needing HPC resources have been identified as one of the challenges of the HBP, which need to be taken into account for further evolution of the HPAC Platform and the upcoming ICEI/Fenix infrastructure.

Our approach was use case driven. The following use cases have been addressed:

- Deep learning on large images
- Parallel image registration
- Cell detection and feature calculation
- Visualisation of compartment data

The work focussed among others on enabling the use of new storage technologies. New storage devices based on dense and non-volatile memory technologies provide much higher performance, both in terms of bandwidth as well as IOPS (input/output operations per second) rates, while compromising on capacity. To provide access to storage devices integrated into HPC systems, like the HBP PCP Pilot systems, a variety of technologies have been deployed, explored and enhanced, ranging from parallel file systems like BeeGFS, new object store technologies like Ceph, key value stores (KVS) enhanced by powerful indexing capabilities, to technologies like IBM's Distributed Shared Storage-class-memory (DSS).

High-performance non-volatile memory is a precious resource, therefore, its co-allocation together with compute resources becomes necessary. Scheduling strategies for specific scenarios based on the needs of use cases have been developed and evaluated using simulators.

Specific data analytics workflows have been considered and implemented in co-design with data analytics experts.

Tests and performance explorations were performed on the pilot systems JULIA and JURON installed at Jülich Supercomputing Centre (JUELICH-JSC). JULIA is a Cray CS-400 system with four DataWarp nodes integrated, each equipped with two Intel P3600 NVMe drives. These and the compute nodes were integrated in an Omnipath network. Ceph was deployed on the DataWarp nodes. The other system, JURON, is based on IBM Power S822LC HPC ("Minsky") servers. Each server comprises a HGST Ultrastar SN100 card. On this system BeeGFS, DSS and different key value stores were deployed.

2.2.1 Achieved Impact

The work related to this key result laid the basis for integration of hierarchical storage architectures and related technologies within the upcoming ICEI infrastructure. Part of this work was performed in collaboration with the HBP PCP partners and ThinkParQ as a further commercial operator.[1]

Many of the results produced have been deployed on the PCP pilot systems to make them available to early HBP users. This includes different storage technologies, interface libraries including the newly developed Brion I/O library and the data analytics workflows for the human brain atlas.

Results from simulations of co-allocation scheduling strategies will be used as basis for implementation efforts foreseen for the next HBP phase.

2.2.2 Component Dependencies

Component ID	Component Name	HBP Internal	Comment
522	Report on hierarchical data store software components design specification	Yes	Collection of guiding design principles for deploying hierarchical storage within the HPAC Platform
523	All tests and performance explorations related to components listed in this section have been performed on the pilot systems JULIA and JURON installed at Jülich Supercomputing Centre (JSC). JULIA is a Cray CS-400 system with four DataWarp nodes integrated, each equipped with two Intel P3600 NVMe drives. These and the compute nodes were integrated in an Omnipath network.	No	Different software components for exploiting hierarchical storage architectures and/or dense memory based storage devices

[1] See, e.g., the following White Paper: "NVMe and OpenPOWER Performance on JURON", (https://www.beegfs.io/docs/whitepapers/JURON_OpenPOWER_NVMe_by_ThinkParQ_FZ-Juelich.pdf).

	Ceph was deployed on the DataWarp nodes. The other system, JURON, is based on IBM Power S822LC HPC ("Minsky") servers. Each server comprises a HGST Ultrastar SN100 card. On this system BeeGFS, DSS and different key value stores (Cassandra and Scylla) were deployed. More information about the pilot systems is available in the HPAC Platform Guidebook (https://hbp-hpc-platform.fz-juelich.de/). Hierarchical data store software components (prototype)		
524	Evaluation of the hierarchical data store prototypes	No	Report on evaluation of different data store technologies
525	Identification of basic data types for coupling data analytics and visualisation to simulation	Yes	Define objects to be exchanged in the pipeline between simulation and visualisation
526	Data store sources for interactive visualisation for selected data types	No	Libraries interfacing between different data store types, data models and visualisation applications
527	Report on data resource allocation schemes and resource co-allocation strategies	Yes	Definition of suitable data resource allocation schemes and identification of possible resource co-allocation strategies
528	Compute and data resource co-allocation	Yes	Description of strategies for co-allocating compute and storage resources
529	Evaluation of co-allocation prototype	Yes	Results from expanded batch scheduler simulator BatSim
530	Data analytics workflows for the human brain atlas	Yes	Full analysis workflow used in the context of the "Multi-level Human Brain Atlas" utilising components developed within this key result

2.3 Interactive visualisation and visual data analysis

In data-intensive neuroscience, interactive data analysis workflows play a central role for the scientific endeavour. Visual analysis (VA) techniques are a key component in many of these workflows. VA applications have to address the specific use cases and requirements of the neuroscientists using them. To this end, user-centric development and close collaboration with the users are essential, already during development. In addition, basic, shared functionalities have to be made available in re-usable frameworks to enable a seamless integration into users' workflows or to directly couple analysis tools to live data sources, such as large-scale simulations. A two-fold approach was followed to meet these challenges. On the one hand, various applications were conceived, which target specific analysis scenarios. On the other hand, a number of software frameworks were implemented for the integration of interactive and visual analysis applications into the hardware infrastructure. This two-fold approach allowed the creation of increasingly sophisticated interactive visual analysis tools, that actually add value for end users.

It is important to note that for each developmental effort within WP7.3 there is a specific, neuroscientific use case that drives its development; we identified each of these use cases through collaborations inside the HBP. The focus lies on the development of visualisation applications for the investigation of computationally intensive neural network simulations on the one hand, and on massive image data created from brain scans on the other hand. Following the architecture presented in Figure 2 below, software development focused on the framework and application layers. We developed visualisation applications that target the analysis of neural networks and simulation models, the activity data originating from those neural simulations, and the aforementioned large neuronal imaging data, such as 3D-Polarized Light Imaging (3D-PLI). To handle the large data sets emerging from running simulations, we are currently developing an *in situ* integration between live simulations and analysis methods (continuing in the next phase of the HBP). An event-driven messaging library is under development for the flexible linking of applications with automatic discovery, and streaming and steering capabilities. Furthermore, an ontology-supported integration system allows users to flexibly mix and merge multiple views into “coordinated multiple view systems” that are custom-tailored for their respective analyses. This enables neuroscientists to adapt their tools to the frequently changing requirements of data analysis workflows in their daily research work. To scale common visualisation algorithms to next-generation HPC systems, we investigated the use of the High Performance ParalleX (HPX) runtime as an execution layer to the widely supported vtk-m visualisation library. This holds the potential to scale vtk-m beyond the confines of a single, shared-memory system.

Below, we give an overview of the various visualisation applications and libraries, which form the major outcome of WP7.3. We focus on components that have been actively developed during the past twelve months. Nevertheless, the development of all tools started in earlier phases of HBP. Some of the activities are based on libraries developed outside this project as in-kind contributions or open source third-party projects. We see this presentation as an overview of major software components for the interactive visual data analysis for simulation and image data emerging from the HBP and beyond.

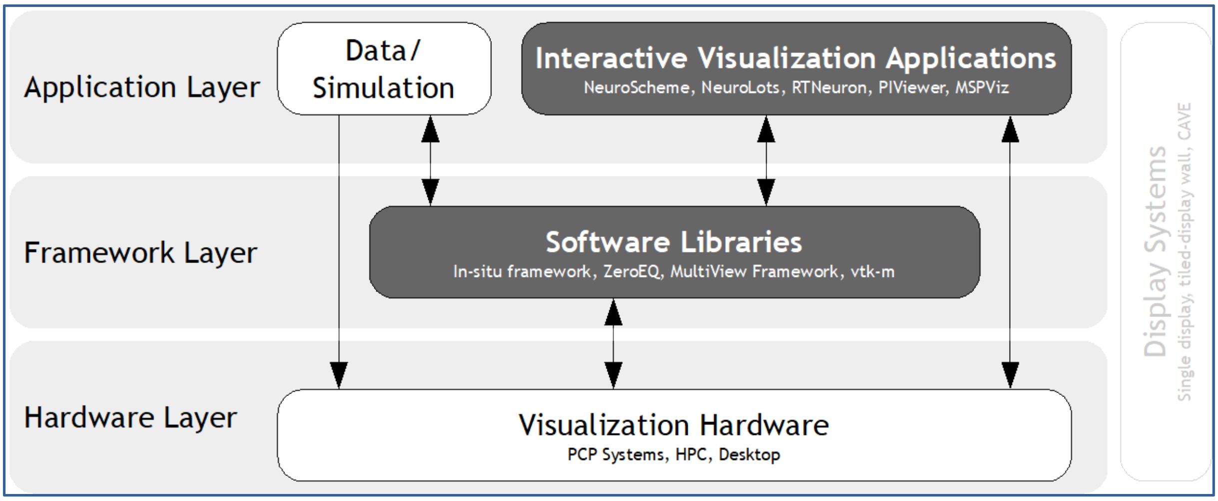

Figure 2: General visualisation software architecture in SP7

Software architecture for the development of interactive and visual data analysis capabilities in SP7

Software developed on Application Layer:

RTNeuron is a tool for the interactive visualisation and media production of simulation results for detailed neuronal network simulations. It allows the visualisation of neurons, synapses and playback of simulation data, as well as arbitrary user given geometry, and it provides some advanced rendering capabilities, such as order-independent transparency and parallel rendering. RTNeuron consists of a C++ library with the rendering back end, a Python wrapper and a Python application called rtneuron-app.py. GUI overlays can be created for specific use cases using PyQt and QML. Some power applications with GUI overlays have already been provided. During this period one application for interactive visualisation of hippocampus models has been developed. Using Brion as data access library, RTNeuron can access simulation data from key-value stores (see 0 and 3.2.5). RTNeuron is available on the pilot system JULIA and on JURECA as environment module.

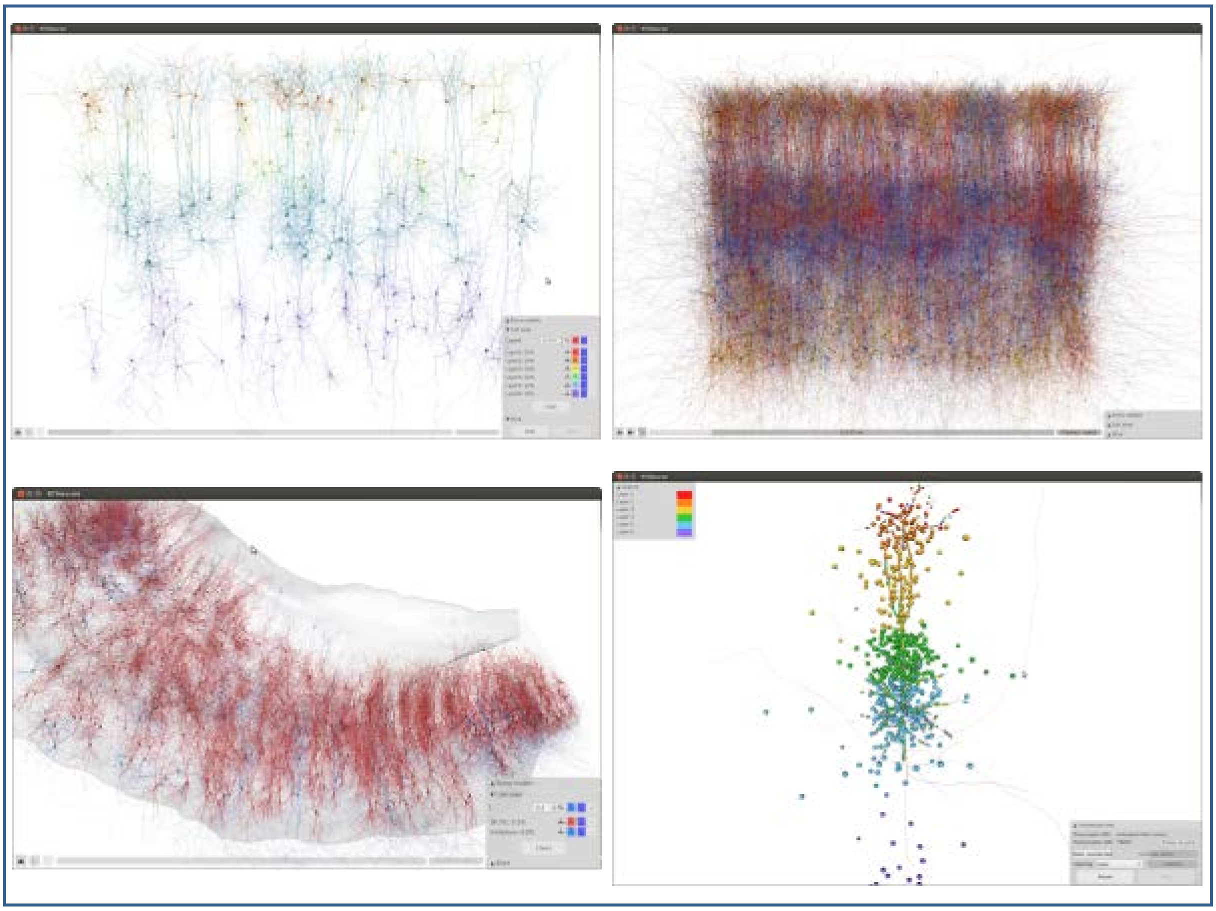

Figure 3: Visualisations provided by RTNeuron

Visual representation of cell dyes (top left), simulation playback (top right), interactive circuit slicing (bottom left), connection browsing (bottom right).

MSPViz is a web-based visualisation tool for structural plasticity models. It uses a novel visualisation technique based on the representation of neuronal information through the use of abstract levels and a set of representations in each level. This hierarchical representation lets the user interact and change the representation, modifying the degree of detail of the information to be analysed in a simple and intuitive way, through the navigation of different views at different levels of abstraction. The designed representations in each view only contain the necessary variables to achieve the desired tasks, thus avoiding overwhelming saturation of information. The multilevel structure and the design of the representations provide organised views, which facilitate visual analysis tasks. Moreover, each view has been enhanced adding line and bar charts to analyse trends in simulation data. Filtering and sorting capabilities can be applied on each view to ease the analysis. Additionally, some other views, such as connectivity matrices and force-directed layouts, have been incorporated, enriching the already existing views and improving the analysis process. Finally, this tool has been optimised to lower render and data loading times, even from remote sources such as WebDav servers.

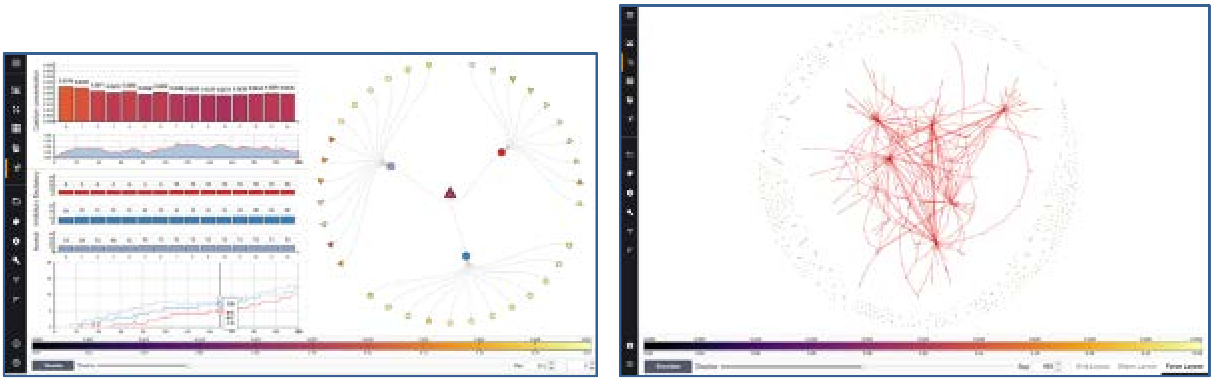

Figure 4: MSPViz

Two views of MSPViz to investigate structural plasticity models on different levels of abstraction. Left: connectivity of a single neuron; right: full network.

NeuroLOTs is a set of libraries and tools to generate 3D meshes that approximate the anatomy of neurons and brain vasculature, to visualise them at different detail levels using GPU-based tessellation. As a part of NeuroLOTs, **NeuroTessMesh** provides a visual environment for the generation of 3D polygonal meshes that approximate the membrane of neuronal cells, starting from the morphological tracings that describe neuronal morphologies. The 3D models can be tessellated at different levels of detail, providing either a homogeneous or an adaptive resolution of the model. The soma shape is recovered from the incomplete information of the tracings, applying a physical deformation model that can be interactively adjusted. The adaptive refinement process performed in the GPU generates meshes, that allow good visual quality geometries at an affordable computational cost, both in terms of memory and rendering time. NeuroTessMesh is the front-end GUI to the NeuroLOTs framework.

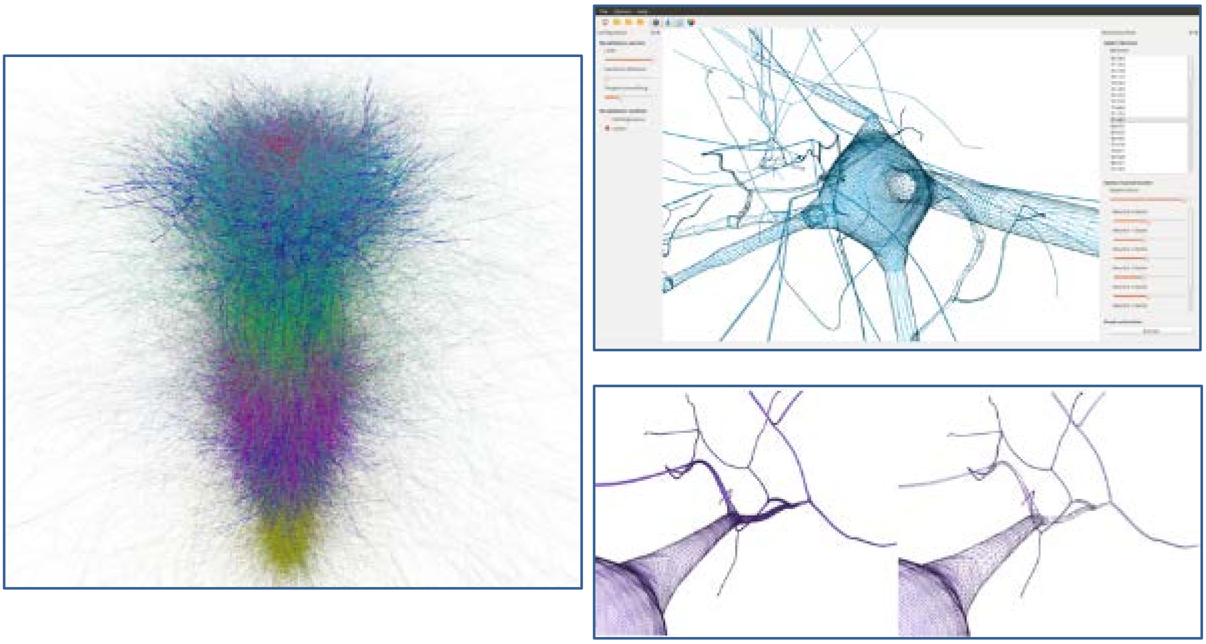

Figure 5: NeuroLOTs

NeuroLOTs visualisation methods for large realistic networks and tessellation of neurons.

NeuroScheme is a tool to navigate through circuit data at different levels of abstraction, using schematic representations for fast and precise data interpretation. It also allows filtering, sorting and making selections at these different levels of abstraction. Finally, it can be coupled to realistic visualisation or other applications using the ZeroEQ event library and it has been integrated into the multi-view framework, both are also developed in WP7.3.

This application allows analyses based on a side-by-side comparison using its multi-panel views, and it also provides focus-and-context. Additionally, its different layouts enable arranging data in different ways: grid, 3D, camera-based, scatterplot-based or circular. Besides, it provides editing capabilities, to create a scene from scratch or to modify an existing one. Another part of the NeuroScheme framework is **ViSimpl**, a prototype developed to analyse simulation data, using both abstract and schematic visualisations. This analysis can be done visually from temporal, spatial and structural perspectives, with the additional capability of exploring the correlations between input patterns and produced activity.

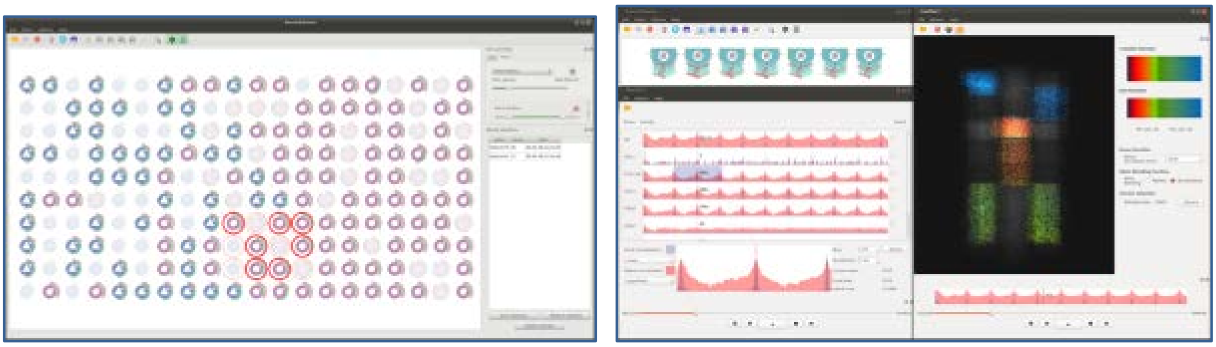

Figure 6: NeuroScheme and ViSimpl

Visualisations provided by NeuroScheme and ViSimpl. Overview of various neurons (left); user interface of ViSimpl visualising activity data emerging from a simulation of a neural network model (right).

NEST-simulated spatial-point-neuron data visualisation: Complementary to other viewer and visualisation implementations for NEST simulations, this component offers a rendering of activity and membrane potentials in a neural network simulated with NEST (left, Figure 7). A prototypical implementation exists that is based on vtk, the widely-used visualisation toolkit. This implementation can be generally run by computational neuroscientists on their workstations, imposing only moderate hardware requirements. Experiments using rendering on high-performance computing infrastructure were successful. The results indicate that this component is extensible towards large-scale simulations that require HPC resources and thus produce large output data. The hi-fidelity rendering used in this case provides very high quality images that may be suitable for publications (right, Figure 7).

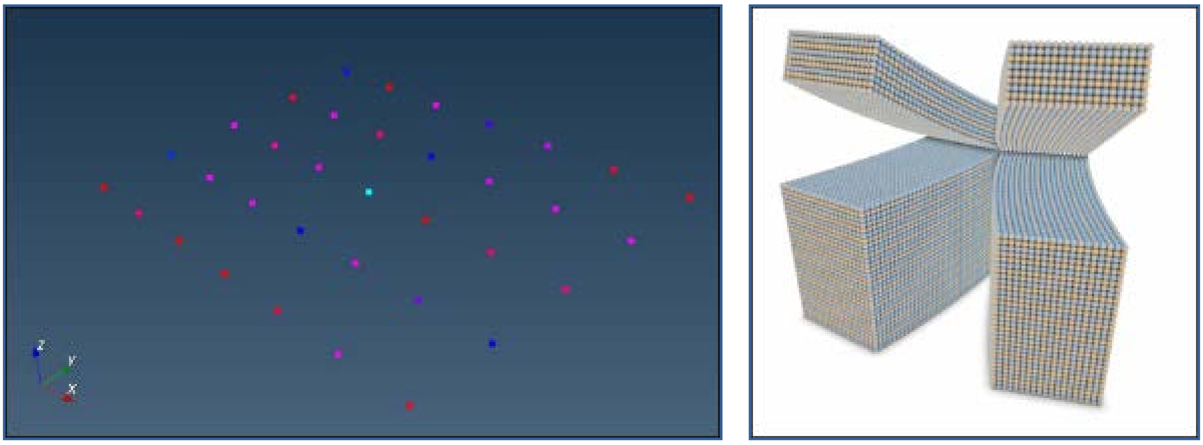

Figure 7: Visualisation of NEST simulation

Left: rendering of color-coded membrane potentials on spatial neurons from a running NEST simulation; right: proof of concept of a high-quality rendering of spatially organized point neurons.

PLIViewer: The study of the connectome investigates structural and functional connectivities in the brain. Structural connectivity refers to anatomical connections between brain areas, whereas functional connectivity describes the short-term, dynamic correlations between neural activities of distinct brain structures. 3D-Polarized Light Imaging (3D-PLI) is a recent neuroimaging technique, to study structural connectivity of the brain at unprecedented resolutions, within the micrometre range. The major outputs of 3D-PLI are four scalar fields (transmittance, retardation, inclination, direction maps) and a vector field (fibre orientation maps) which depict the 3D spatial orientation of myelinated nerve fibres. The PLIViewer is visualisation software for 3D-PLI, to interactively explore the scalar and vector datasets; it provides additional methods to transform data, thus revealing new insights that are not available in the raw representations. The high resolution provided by 3D-PLI produces massive, terabyte-scale datasets, which makes visualisation challenging. The **PLIViewer** tackles this problem by providing functionality to select areas of interests from the dataset, and options for downscaling. In addition, it makes it possible to interactively compute and visualise Orientation Distribution Functions (ODFs) and polar plots from the vector field, which reveal mesoscopic and macroscopic scale information from the microscopic dataset without significant loss of detail. Overall, the PLIViewer equips the neuroscientist with specialised visualisation tools he/she needs to explore 3D-PLI datasets through direct and interactive visualisation of the data.

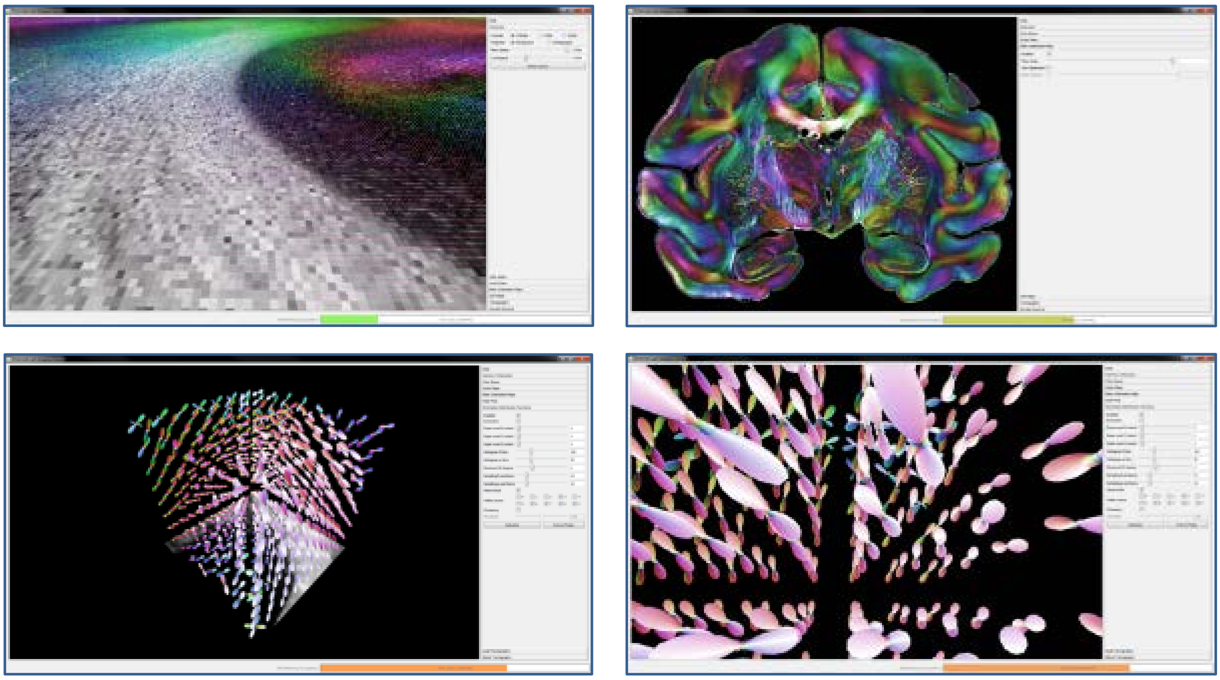

Figure 8: PLIViewer

Visualisations provided by the PLIViewer: the original dataset: Fibre Orientation Maps rendered on top of Retardation map (top left); a full slice from the Fibre Orientation Map of a Vervet Monkey (top right); Orientation Distribution Functions (ODFs) rendered with Streamline Tractography (bottom left); a close-up view of the ODFs (bottom right).

Software developed on Framework Layer:

[ZeroEQ](#) is a cross-platform C++ library for implementing event-driven architectures using modern messaging. It provides pub-sub and request-reply messaging using ZeroMQ and integrates REST APIs with JSON payload in C++ applications, using an optional http server. The main intention of ZeroEQ is to allow the linking of applications using automatic discovery. Linking can be used to link multiple visualisation applications, or to connect simulators with analysis and visualisation codes to implement streaming and steering. One example of the former is the interoperability of NeuroScheme with RTNeuron, and one for the latter is the streaming and steering between NEST and RTNeuron. Both were reported previously, whereas the current extensions focus on the implementation of the request-reply interface.

[VTK-m](#) is a scientific visualisation and analysis framework that offers a wealth of building blocks to create visualisation and analysis applications. VTK-m facilitates scaling those applications to massively parallel shared memory systems, and it will – due to its architecture, most likely also run efficiently on future platforms. **HPX** is a task-based programming model. As such, it simplifies the formulation of well-scaling, highly-parallel algorithms. Integrating this programming model into VTK-m streamlines the formulation of its parallel building blocks and thus makes their deployment on present and emerging HPC platforms more efficient. Since neuroscientific applications require more and more compute power as well as memory, harnessing the available resources will become a challenge in itself. By combining VTK-m and HPX into task-based analysis and visualisation, we expect to provide suitable tools to effectively face this challenge and facilitate building sophisticated interactive visual analysis tools, tailored to the neuroscientists' needs.

For this purpose, parallel primitive algorithms required for VTK-m have been added to HPX along with API support to enable the full range of visualisation algorithms developed for VTK-m. Furthermore, a new scheduler has been developed that accepts core/numa placement hints from the programmer such that cache reuse can be maximised and traffic between sockets minimised. High performance tasks that access data shared by application and visualisation can use this capability to improve performance. Additionally, the thread pool management was improved to allow visualisation tasks, communication tasks, and application tasks to execute on different cores if necessary, which reduces latency between components and improves the overall throughput of the distributed application. Finally, RDMA primitives have been added to the HPX messaging layer. These improvements make it possible to scale HPX applications to very high node/core counts. Respective tests have been successful on 10k nodes using 650k cores.

The **NEST in situ framework** developed by RWTH facilitates visualisation and analysis of the output data of a NEST simulation while this is still running (left, Figure 9). For this purpose, membrane potentials, spikes and other data are streamed from the simulation. The framework builds on top of conduit, a well-established *in situ* library, for compatibility and extensibility reasons. The framework consists of a well-tested, compact C++ library that has to be linked into the NEST simulator in order to provide the streaming capabilities. Additionally, it can be linked into consumer applications for visualisation and analysis. Python bindings for consumer applications (visualisation and analysis) are also provided in order to make it more useful for computational neuroscientists who are familiar with NEST and Python. A – yet small – set of demos provides usage examples (right, Figure 9).

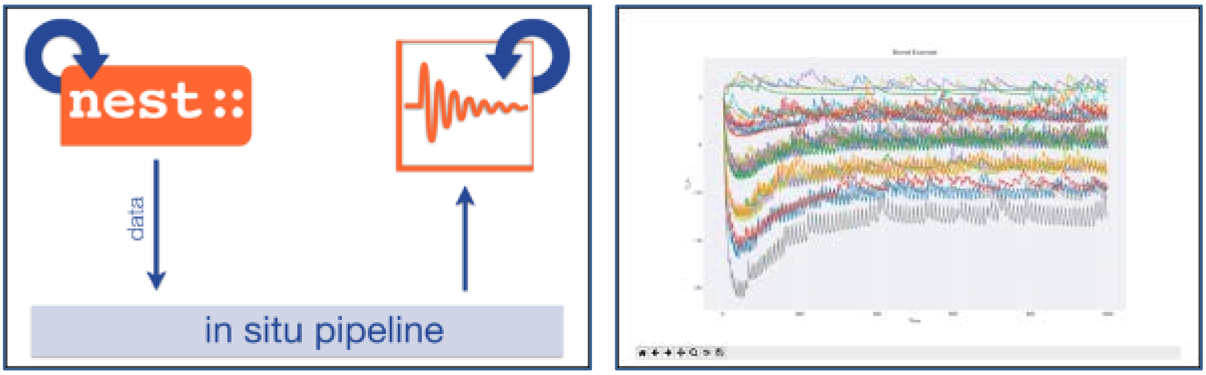

Figure 9: NEST *in situ* framework

Implemented data flow in the *in situ* pipeline (left); line plot of membrane potentials from a running NEST simulation (right).

The **Multi-View Framework** is a software component, which offers functionality to combine various visual representations (views) of one or more data sets in a coordinated fashion. Coordination of multiple views here refers to the communication of interaction events between such views. For instance, if the scientist selects a data item in one view, the framework communicates this selection to all connected views such that this item gets highlighted there accordingly. As the underlying software layer is generic, not only software components offering visualisation capabilities can be included in such a network, but also software components offering other functionality, such as statistical analysis (e.g.

Elephant). Furthermore, multi-display scenarios can be addressed by the framework as coordination information can be distributed over network between view instances running on distributed machines. The framework is composed of three libraries: *nett*, *nett-python* and *nett-connect*. nett implements a light-weight underlying messaging layer enabling the communication between views, whereas nett-python implements a python binding for nett, which enables the integration of python-based software components (such as Elephant) into a multi-view setup. nett-connect adds additional functionality to this basic communication layer, which enables non-experts to create multi-view setups according to their specific needs and workflows. Therefore, it offers a graphical user interface with which the scientist can select the view to be used, start it up and connect it with other already started views. Behind the scenes, this tool uses an ontology-based description of the various views/services provided, such that the system can validate the connections created by the user and suggest matching visual representations to be coordinated for the analysis process. Additionally, once created, setups can be stored, reused and adapted/extended later on. The framework is used in various use cases (as shown in Figure 10). For instance, we tested it for a scenario using Elephant as data analysis component, multiple views for steering a NEST simulation to investigate neural plasticity (publication submitted), for comparative analysis of NEST simulations and multi-device multi-user scenarios for collaborative work.

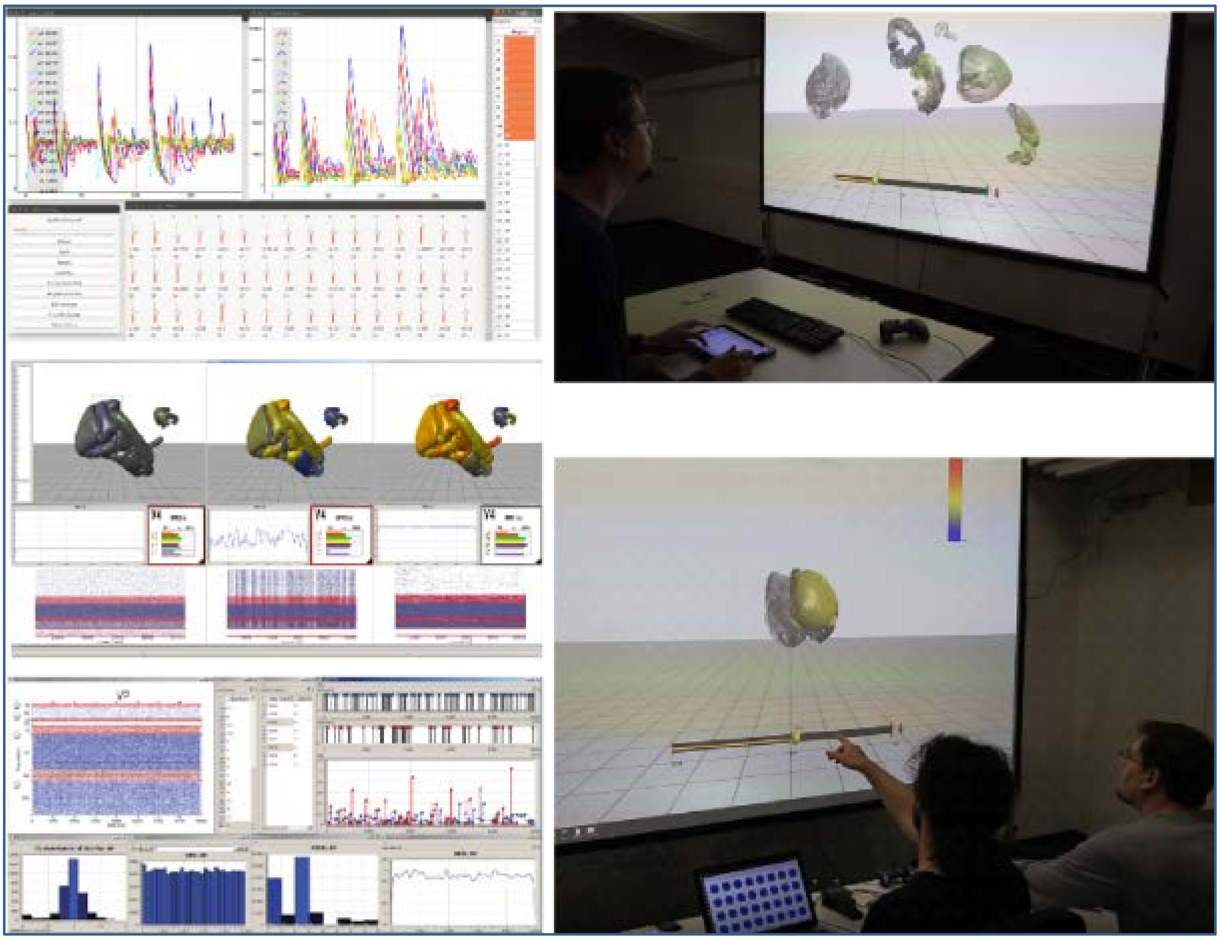

Figure 10: Use of the multi-view framework for various use cases.

Interactive optimization of parameters for structural plasticity in neural network models (top left); comparative analysis of NEST simulations (top right); statistical analysis of NEST simulations (bottom left); multi-device and multi-user scenarios (bottom right).

2.3.1 Achieved Impact

This work made two major contributions to the overall project. First, we implemented and provided several interactive visualisation applications (application layer) for a variety of use cases that emerged from close collaboration within HBP. We initiated new collaborations, intensified existing ones, and supported neuroscientists' use case dependent needs and requirements for visualisation, which also resulted in joint publications. Furthermore, most tools reached a high maturity level and they are ready for deployment on a larger scale in the project, which is planned for the next project phase.

Second, we continued the development of frameworks for various integration aspects (framework layer), such as *in situ* visualisation of large simulation and image data (continued for the next years), steering of neural simulation, and the integration and coordination of multiple views beyond individual analysis applications. We drove the framework development, guided by various neuroscientific use cases emerging from the project, which enabled co-design of analysis workflows and resulted in joint publications.

2.3.2 Component Dependencies

Component ID	Component Name	HBP Internal	Comment
2337	Large-scale PLI data explorer	No	
2336	NEST-simulated spatial-point-neuron data visualisation	No	
547	NEST in situ framework	No	
231	ZeroEQ	No	
543	RTNeuron	No	Currently closed source, but available on demand from the developers; used in Data store sources for interactive visualisation for selected data types
226	VTK-m	No	
548	Task based analysis on HPX, VTK and VTK-m	No	
234	MSPViz	No	
232	NeuroLOTs	No	
233	NeuroScheme	No	
549	Software library for multi-view analysis of neuroscientific data	No	Currently closed source, but available on demand from the developers

2.4 Dynamic resource management tools and techniques

Our main goal was to develop a novel approach for resource management in supercomputers, to enable changing resource assignment during the application run time. We call this approach *dynamic resource management*, also known as *malleability* (i.e. adaptable, able to change or adjust). Supercomputers use *job scheduler* software to match petitions of applications to be executed (i.e. jobs), to the available resources on the supercomputer (i.e. the nodes, with many resources to be managed, like CPUs, GPUs, main memory, secondary storage, …). To avoid interferences, job schedulers typically do not allow that different jobs share the same node on the supercomputer, so the maximum performance can be ensured for a single job, but at the price of achieving a lower overall utilisation of the resources, if the job is not able to fully exploit all resources reserved and assigned to it.

In addition, these days, supercomputer architectures tend to have more and more resources inside a single node. If we review current trends in computer architecture, we can see that nodes are commonly composed of chips with many computing cores (e.g. up to 72 cores in Intel's KNL), specific purpose computing units, such as GPUs or FPGAs, and a new layer of hierarchy in the memory of the system (i.e. Non-Volatile Memory). With this tendency of having more and more resources and complexity inside a single node, it gets unlikely that a single application is able to exploit all resources inside a node at the same time. This means that if we want to increase the overall utilisation of the supercomputer, which is very important due to the economic cost of the infrastructure, sharing resources between applications becomes a must, and that is where dynamic resource management makes a difference.

The dynamic resource management Key Result is composed of a set of results dealing with different topics but all working towards the same objective. In particular, the list of results we have achieved is:

- CoreNeuron with enabled malleability
- NEST with enabled malleability
- Definition of APIs for job schedulers
- Job scheduler with dynamic resource management capabilitiesJob scheduler with dynamic resource management capabilities
- Job scheduler with new scheduling policies

Our strategy has been to use two of the most important brain simulators in the HBP, i.e. CoreNeuron and NEST – The Neuronal Simulation Tool, as representative use cases to show what can be achieved with dynamic resource management, and as real applications to gather requirements from. Both codes have been adapted to allow them to change the number of computing cores they use at runtime. We have made them publicly available, with the intention of showing the types of adaptations that need be done to existing codes when malleability is targeted, therefore as a proof-of-concept, rather than as production-ready versions. These modified codes correspond to the components CoreNeuron with enabled malleability and NEST with enabled malleability; the code is publicly available.

Moreover, several tools have been used to study the performance of original, as compared to malleable, versions of the code such as **Paraver**[2] and **Extrae**[3], that enable the performance analysis, as well as **OmpSs**[4], a parallel programming model.

[2] https://hbp-hpc-platform.fz-juelich.de/?hbp_software=paraver
[3] https://hbp-hpc-platform.fz-juelich.de/?hbp_software=extrae
[4] https://hbp-hpc-platform.fz-juelich.de/?hbp_software=ompss

To enable malleability in current systems, information needs to flow between the different layers of the software stack so that scheduling can happen at a finer grained level, and interactions between the different schedulers of the system are supported. With that objective in mind, we defined a simple architecture of the different schedulers in a supercomputer, acting at different levels. We identified four levels of schedulers (job level, node level, application level and kernel level) as depicted in Figure 11, and we handled three of them with the efforts available in the project, i.e. all except the kernel level. Once identified, we defined different APIs to enable the communication between these scheduling levels, to achieve a holistic view of the scheduling and to make all layers work towards the same objective. The interaction of these layers is achieved with the Dynamic Load Balancing Library (see section 3.4.6), a software library that enables the resource sharing between applications. These results are summarised in the component "Definition of APIs for job schedulers". We produced an internal report describing these APIs (see section 3.4.3):

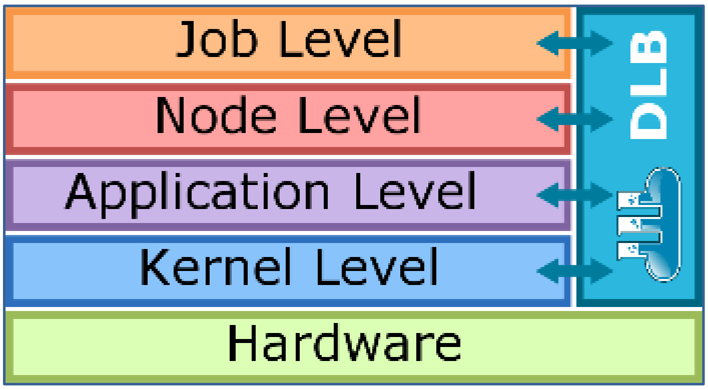

Figure 11: Scheduling levels in a supercomputing system

The central aspect of enabling malleability is to achieve that the main system in charge of scheduling of jobs to the supercomputer can support it. We have selected SLURM as a representative job scheduler to implement modifications to achieve the share of resources between jobs. SLURM is available at many supercomputing facilities around the world, and it is distributed as open source, so for us developing a prototype on top of it has been easier than it would have been for others. We have not distributed this modified version of SLURM yet, due to its early prototype status. However, in the future we consider to include this in the SLURM distribution, and we also keep a good record of the modifications needed to implement malleability, so any other job scheduling system could be adapted following similar steps. This is the component result named Job scheduler with dynamic resource management capabilities.

Another important aspect that dynamic resource management enables is the possibility of implementing new scheduling policies in the job manager (i.e. SLURM) that exploit malleability, which is the component referred as Job scheduler with new scheduling policies. Although it is listed in section 3.4 as a separate component, we have implemented our algorithms in our modified version of SLURM, that is described above. We have implemented two generic policies (that can be applied to any application) to enable malleability with the objective of reducing the response time of jobs and to increase the utilisation of the supercomputer. The first generic policy takes into account a maximum slowdown for a job when sharing its resources, to avoid excessive sharing of resources for a single application. The second generic policy uses the runtime of the application that is its sharing resources as a boundary, to ensure that the sharing does not make the application surpass a certain runtime. In addition, a specific policy has been tailored for CoreNeuron. This specific policy takes into account the phase in which the code is currently running (input reading or computation), and thus enables successful resource sharing with other processes with compatible requirements, considering the memory demands of each process, since CoreNeuron has a high memory utilisation. In this case we make sure that the new processes will not overload the node's memory. By parsing CoreNeuron input files we have been able to determine their size, and therefore predict memory utilisation and initialisation time without any user intervention.

2.4.1 Achieved Impact

Our work plan towards the achievement of this Key Result also includes investigating the sustainability of our solutions and a possible technology transfer to vendors. In particular, we worked with two of the vendors involved in the HBP Pre-Commercial Procurement (PCP), i.e. IBM-NVIDIA and Cray, which took place during the HBP Ramp-up Phase. In the first case, IBM has shown a clear acknowledgement of the future necessity of dynamic resource management, by arguing that malleability has been discussed with some of their customers. On the opposite, in the case of Cray, their experience with their clients does not show off that malleability may be needed by now. To gain insights about this necessity, Cray requested more real use cases from neuroscience, which we successfully delivered to them. For us, the contact with Cray has shown that the dynamic resource management technology is still in an early research phase, but that it has a lot of potential.

Apart from the PCP vendors, we have been in contact with Lenovo (Luigi Brochar) and Intel (Hans-Christian Hoppe), since BSC has specific, on-going research projects with these two companies, and we presented them the ideas and results achieved so far in dynamic resource management. Their feedback has been very valuable for us, and they have clearly shown interest in the future of the topic, as something that may be included in their products. This strategy was defined at the first year of the SGA1, and during the second year the contacts with the vendors have continued.

2.4.2 Component Dependencies

Component ID	Component Name	HBP Internal	Comment
537	CoreNeuron with enabled malleability	No	The CoreNeuron application has been made malleable
536	NEST with enabled malleability	No	The NEST simulator code has been made malleable
538	Definition of APIs for job schedulers	No	APIs that enable the holistic view of the scheduling, and provide a way of interaction between the four levels of scheduling in the system
539	Job scheduler with dynamic resource management capabilities	No	Malleability capacities implemented in SLURM
541	Job scheduler with new scheduling policies	No	New scheduling policies implemented in SLURM able to exploit malleability
223	Dynamic Load Balancing Library	No	A library able to change the amount of resources used by an application inside a node of the supercomputer

2.5 High Performance Analytics and Computing Platform v2

The High Performance Analytics and Computing (HPAC) Platform developed new services during the last twelve months, improved already existing ones and put all of them into operational state. At the end of SGA1, the HPAC Platform thus offers the following new capabilities to the users:

Enablement of PRACE network

HPAC sites provide appropriate services and endpoints (e.g. UFTPD server and client nodes) that can communicate over the PRACE network, in addition to services and endpoints on the public network.

Deployment of high-performance data transfer service

UFTP is a high-performance data transfer library based on FTP, fully integrated with the UNICORE authentication mechanisms and accessible via the UNICORE Rest API. To give users the ability to move datasets between the sites that are part of the data federation, a transfer service based on UNICORE File Transfer Protocol (UFTP) was deployed and configured on the four HPAC sites, i.e. at BSC, CINECA, ETHZ-CSCS and JUELICH-JSC.

Currently UFTP is available via two different interfaces:

- The UNICORE command line client (ucc), a full-featured client for the UNICORE middleware, able to start both direct data transfer (local machine to server) and third party transfer (server to server);
- Jupyter notebooks available in the Collaboratory web portal, which enable users to integrate data transfer steps inside more complex workflows.

Additional monitoring information

The HPAC monitoring service provides the following information:

- Status of UNICORE services
- Network reachability
- Scheduled maintenances

User creation workflow operational

Over 100 user accounts have been created for the HPAC Platform. *Ad hoc* APIs have been developed to provide uniform access to user management procedures to all HPAC sites.

OpenStack (IaaS) service operational

Users now have access to a production, state-of-the-art Infrastructure-as-a-Service (IaaS) provided by OpenStack. The system is integrated with the ETHZ-CSCS infrastructure and will be fully integrated with the HPAC Authentication and Authorisation Infrastructure. The OpenStack APIs offer a flexible tool for creating on-demand virtual computing infrastructures, which also enable the interaction with the traditional HPC systems, see “Lightweight virtualization service operational” below for further details.

At the end of SGA1, 58 virtual machines (VMs) are running for the Collaboratory team, and another 56 VMs for the Neurorobotics Platform (SP10).

Object Storage service operational

The HPAC Platform now offers an Object Storage service provided by OpenStack (SWIFT) and it is integrated with the ETHZ-CSCS GPFS storage infrastructure. Object Storage dramatically improves data accessibility and customisations from the Collaboratory, and serves as a key component for the HBP Archival Storage. The current usage of the service is about 8 TB.

Lightweight virtualisation service operational

Under the joint effort between the Neurorobotics Platform (NRP; SP10) and the HPAC Platform, all current components of the NRP have been containerised into functional Docker images: frontend server, back end server and Gazebo server. Moreover, two workflows have been enabled using Docker images:

1) Full container deployment over OpenStack, which is intended for allowing the NRP to run and manage workshops, this use case has workloads that require low performance but with higher user management overheads.
2) Front end and back end servers run on the OpenStack while the robotic simulation is executed by the Gazebo server, which is launched on the TDS of the supercomputer Piz Daint.

Jupyter notebook support by HPAC middleware operational

In collaboration with the Collaboratory team, the Python client library for the UNICORE middleware has been extended. Among other improvements, a new FUSE driver allows to mount HPC storage into Jupyter notebooks running in the Collaboratory.

User support ticket system operational

HPAC has implemented a dedicated ticket system for tracking HPAC-specific issues and user enquiries. BSC has been actively involved in the daily assessment of incoming tickets, including 1st level support and, when applicable, dispatching them to the appropriate HPC site for resolution. Another activity during the M13-M24 period is the improvement of the integration between the HPAC ticketing system and the local ticketing systems used by each HPC site to address 2nd level support tickets, in particular customising the integration with the OTRS system used at JSC, which required changes in the RT code to properly propagate messages.

The portfolio of services and user adoption of the HPAC infrastructure has significantly increased during the last twelve months, and that is reflected by a growing number of processed requests of more than 160, which is an increase by a factor six as compared to the previous twelve months. The reasons for the strong increase are on the one hand the significantly increased usage of the Platform as compared to previous years, and on the other hand the establishment of the HPAC-internal policy to handle all user requests via the ticketing system.

Achievement 10: Arbor library for performance-portable neural network simulators

Arbor is a performance-portable software library for simulators of networks of multi-compartment neurons. During the last year, performance portability was completed for the three main target HPC architectures available in the HBP: Intel x86 CPUs (AVX2 and AVX512), Intel KNL (AVX512) and NVIDIA GPUs (CUDA). The source code was released publicly on GitHub with an open source BSD license, along with documentation on *Read the Docs*.

2.5.1 Achieved Impact

The ultimate objective of the HPAC Platform is to provide an advanced data and computing infrastructure to enable scientific research, therefore considerable efforts have been put into engaging with other SPs. This work proved fruitful, as at least two other HBP Platforms and the Collaboratory framework are now served by the HPAC Platform, with others we have close collaborations. In particular, the consolidation of Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) has made it possible to integrate the following activities:

1) Neurorobotics Platform (NPR; SP10): Considerable effort has been put into the containerisation of the NRP application to make its deployment simpler and portable. The NRP architecture has been partly redefined in order to enable access to ETHZ-CSCS supercomputers through the HPAC IaaS.
2) Neuroinformatics Platform (SP5): Scientists are now heavily using the Object Storage service and taking advantage of its versatility. After the initial six months that this service is available, 13 TB of data are already stored.
3) Jupyter notebook support allowed workflows of the Brain Simulation Platform, for example CDP2 key result 1 “Single Cell Model Builder”, CDP2 key result 2 “Multi-scale

validation" or CDP2 key result 3 "In silico microcircuit experimentation", to rely on the HPAC middleware for accessing supercomputing resources.

4) The Collaboratory is now fully hosted on the HPAC IaaS. The current system is designed for production and provides advanced levels of reliability, better performance, greatly increased scalability and vicinity to data storage. The Collaboratory will also benefit from every upgrade and new services introduced in HPAC.

2.5.2 Component Dependencies

Component ID	Component Name	HBP Internal	Comment
329	SP7 Federated HPAC Data and Computing Services	No	Some resources and services are directly accessible, others require an application
566	HPAC Monitoring Service	No	The service can be used by any user of the HPAC Platform, HBP-internal or –external.
968	VM Services	No	Available to anyone with an HBP Identity account
562	Container-based software packaging and deployment technologies	No	The services hosted using these technologies, e.g. the Neurorobotics Platform, may be accessible for HBP-external users as well
338	PCP Pilot systems	Yes	HBP-external neuroscientists can get in touch with the HPAC Platform to get access, which is possible under certain conditions.
410	Active data repositories	No	These repositories will become part of the Fenix infrastructure targeting multiple communities.
409	Archive data repositories	No	See above
334	HPAC Network Service	No	
400	HPAC Reporting and Accounting Service	No	
416	Transfer services	No	Available to non-HBP users of the HPAC Platform with storage allocations
336	HPAC Authentication and Authorisation Infrastructure Services	No	
792	HPAC UNICORE services	No	
569	HPAC Security Services	No	
945	Arbor	No	
1071	Application Software Services	Yes	
1114	HPAC Ticket System	No	
577	HPAC User Support Services	No	
337	HPAC Platform Guidebook	No	

581	HPAC Platform validation	581	

3. Component Details

The following is a list of the newly released internal Components for this deliverable.

3.1 Simulation technology for Petascale systems: concepts, numerical algorithms and software technology

3.1.1 Continuous dynamics code in NEST

Field Name	Field Content	Additional Information
ID	510	
Component Type	Software	
Contact	FROMMER, Andreas	
Component Description	The continuous dynamics code in NEST enables simulations of rate-based model neurons in the event-based simulation scheme of the spiking simulator NEST. The technology was included and released with NEST 2.14.0. Furthermore, additional rate-based models for the Co-Design Project "Visuo-Motor Integration" (CDP4) have been implemented and scheduled for the next release NEST 2.16.0.	
Latest Release	NEST 2.14.0 20 Oct 2017	https://github.com/nest/nest-simulator/releases/tag/v2.14.0
TRL	TRL 7	
Location	Hosted by other non-HBP 3rd party	GitHub
Format	NEST, PYNN	
Curation Status	Not applicable	
Validation - QC	Pass	Formal NEST Initiative review
Validation – Users	Yes	CDP4, used in preprints
Validation – Publications	Yes	Hahne *et al.* (2017) Front. Neuroinform. 11,34. doi:10.3389/fninf.2017.00034 https://arxiv.org/pdf/1709.05650.pdf https://arxiv.org/pdf/1801.06046.pdf
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	

Licence	GPLv2/GPLv3	
Component Access URL	https://github.com/nest/nest-simulator/releases/tag/v2.14.0	
Technical documentation URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full	
Usage documentation URL	https://github.com/nest/nest-simulator/tree/master/pynest/examples	
Component dissemination material URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full https://github.com/nest/nest-simulator	

3.1.2 Report on numerical techniques for stochastic equations of population dynamics

Field Name	Field Content	Additional Information
ID	657	
Component Type	Report	
Contact	FROMMER, Andreas	
Component Description	Research article on the developed framework for the integration of continuous-time population models in a spiking neural network simulator. After a fruitful peer-review the article was further improved and finally published in Frontiers in Neuroinformatics.	
Latest Release	Doi:10.3389/fninf.2017.00034 24 May 2017	
TRL	Not applicable	
Location	Hosted by other non-HBP 3rd party	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full
Format	Report	
Curation Status	Not applicable	
Validation - QC	Pass	Formal peer-review process
Validation – Users	No	
Validation – Publications	Yes	Hahne *et al*. (2017) Front. Neuroinform. 11,34. doi:10.3389/fninf.2017.00034

Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	CC-BY 4.0	Attribution, https://creativecommons.org/licenses/by/4.0/
Component Access URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full	
Technical documentation URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full	
Usage documentation URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full	
Component dissemination material URL	https://www.frontiersin.org/articles/10.3389/fninf.2017.00034/full	

3.1.3 Simulator-simulator interfaces

Field Name	Field Content	Additional Information
ID	512	
Component Type	Software	
Contact	WITTUM, Gabriel	
Component Description	This component couples the simulation software NEST and UG4 by means of the MUSIC library. Since NEST can only send spike trains, i.e. the point in time, where spiking occurs, UG4 receives those in form of events arriving at synapses (timestamps). The time course of the extracellular potential in a cube (representing a piece of tissue) is simulated based on the arriving spike data. The evolution of the membrane potential in space and time is described by the Xylouris-Wittum model.	
Latest Release	Ug4.0.1 1 Dec 2017	
TRL	TRL 6-7	
Location	Hosted by other non-HBP 3rd party	GitHub
Format	• linenoise • Lua • Metis • ParMetis	http://ug4.github.io/docs/page_copyrights.html http://ug4.github.io/docs/page_external_libraries.html#secLicenses

	• lineintersect utils • RapidXML • Shiny Profiler • tribox • tritri • mpi • executable: ugshell http://ug4.github.io/docs/page_copyrights.html http://ug4.github.io/docs/page_external_libraries.html#secLicenses	
Curation Status	Not applicable	
Validation - QC	Pass	UG4 has been validated in numerous benchmarks and simulations, preferably in comparison with experiments. The interface has been validated by the developers using several tests.
Validation – Users	Yes	UG4 has been validated in numerous benchmarks and simulations, preferably in comparison with experiments.
Validation – Publications	Yes	Vogel, Andreas; Reiter, Sebastian; Rupp, Martin; Nägel, Arne; Wittum, Gabriel: UG 4 - A Novel Flexible Software System for Simulating PDE Based Models on High Performance Computers. Comput Vis Sci, 16 (4), pp 165-179, 2014. Xylouris, K., Wittum, G.: A three-dimensional mathematical model for the signal propagation on a neuron's membrane. Frontiers in Computational Neuroscience 2015.
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	LGPL/GPL v3	https://www.gnu.org/licenses/lgpl.html https://www.gnu.org/licenses/gpl.html
Component Access URL	https://github.com/UG4	
Technical documentation URL	http://ug4.github.io/docs/page_u_g4_usage.html	

Usage documentation URL	http://ug4.github.io/docs/page_u_g4_usage.html	
Component dissemination material URL	N/A	Plugins located on local server, aiming for HBP Model Catalogue App

3.1.4 Report on multi-scale challenges in the HBP simulator hierarchy

Field Name	Field Content	Additional Information
ID	520	
Component Type	Report	
Contact	WITTUM, Gabriel	
Component Description	In this report, challenges are described concerning the coupling of simulators working at different scales (only time, 1D and time, 3D and time). An analysis of the HBP engines (NEST, NEURON, UG4, STEPS, MIIND) and potential simulation coordinators (MUSIC, PyCOMPSs) was performed. Approaches to couple the different scales were described.	
Latest Release	13 Nov 2017	https://collab.humanbrainproject.eu/#/collab/264/nav/45044 https://emdesk.humanbrainproject.eu/shared/5a05ac544cef7-d9b6700bce0ddc2eddb6e6950a7e955f
TRL	Not applicable	
Location	Hosted by Subproject	
Format	PDF	https://www.adobe.com/devnet/pdf/pdf_reference.html
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	
Licence	Not applicable	
Component Access URL	https://collab.humanbrainproject.eu/#/collab/264/nav/45044 https://emdesk.humanbrainproject.eu/shared/5a05ac544cef7-	

	d9b6700bce0ddc2eddb6e6950a7e955f	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.1.5 NEST code with abstracted neuron model representations

Field Name	Field Content	Additional Information
ID	514	
Component Type	Software	
Contact	DIESMANN, Markus	
Component Description	The major result of this task is the development of a novel modelling approach for spiking neurons called NESTML. NESTML is a domain-specific language that supports the specification of neuron models in a precise and concise syntax, based on the syntax of Python. Model equations can either be given as a simple string of mathematical notation or as an algorithm written in the built-in procedural language. The equations are analysed by NESTML to compute an exact solution if possible, or use an appropriate numeric solver otherwise.	
Latest Release	NESTML 2.1.1 05 Jan 2018	https://github.com/nest/nestml/releases/tag/2.1.1
TRL	TRL 7	
Location	Hosted by other non-HBP 3rd party	GitHub
Format	Command-line executable	
Curation Status	Not applicable	
Validation - QC	Pass	Continuous Integration and Code Review Process
Validation – Users	Yes	The developed framework has been validated by NEST users who developed new neuron models with NESTML instead of implementing them as C++ code. NESTML was introduced in numerous tutorials at neuroscience events, including NEST User Workshop 2016 and NEST Conference 2017, LASCON 2016, LASCON 2017. NESTML was additionally validated by using

		questionnaires filled by NESTML applicants.
Validation – Publications	Yes	D. Plotnikov, I. Blundell, T. Ippen, J. M. Eppler, A. Morrison, B. Rumpe: NESTML: a modeling language for spiking neurons. (2016) (http://www.se-rwth.de/publications/NESTML-a-modeling-language-for-spiking-neurons.pdf?PHPSESSID=692226fb540b39a64ee0dca7e68074dc) Inga Blundell, Dimitri Plotnik, Jochen Martin Eppler and Abigail Morrison: Automatically selecting a suitable integration scheme for systems of differential equations in neuron models (to appear 2018). Poster: NESTML: a modeling language for spiking neurons, Blundell, Inga and Plotnikov, Dimitri and Eppler, Jochen Martin and Morrison, Abigail and Rumpe, Bernhard, NEST, IAS Symposium 2016 Poster: NESTML: a modeling language for spiking neurons, Blundell, Inga and Plotnikov, Dimitri and Eppler, Jochen Martin and Morrison, Abigail and Rumpe, Bernhard, NEST, NEST user workshop 2016
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	GPL v2 or later	https://www.gnu.org/licenses/gpl.html
Component Access URL	https://github.com/nest/nestml/	
Technical documentation URL		
Usage documentation URL	https://github.com/nest/nestml/tree/master/doc	
Component dissemination material URL	Not applicable	

3.1.6 Report on neuronal modelling languages and corresponding code generators

Field Name	Field Content	Additional Information
ID	519	

Component Type	Report	
Contact	DIESMANN, Markus	
Component Description	Research article on a number of code generation pipelines that have been developed in the computational neuroscience community, and which differ considerably in aim, scope and functionality. This article provides an overview of existing pipelines currently used within the community and contrasts their capabilities and the technologies and concepts behind them.	
Latest Release	Not yet applicable	
TRL	Not applicable	
Location	Hosted by other non-HBP 3rd party	https://www.frontiersin.org/articles
Format	Report	Scientific publication
Curation Status	Not applicable	
Validation - QC	Pass	Formal peer-review process
Validation – Users	No	
Validation – Publications	Yes	Publication under review
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	CC-BY 4.0	Attribution, https://creativecommons.org/licenses/by/4.0/
Component Access URL	Not applicable	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.1.7 NEST - The Neuronal Simulation Tool

Field Name	Field Content	Additional Information
ID	209	
Component Type	Software	
Contact	DIESMANN, Markus	

Component Description	NEST - The Neural Simulation Tool - is a highly scalable simulator for networks of point or few-compartment spiking neuron models. It includes multiple synaptic plasticity models, gap junctions, and rate-based models. NEST also provides techniques to define complex network structure.	
Latest Release	NEST 2.14.0 20 Oct 2017	DOI 10.5281/zenodo.882971
TRL	TRL 7-8	
Location	Hosted by other non-HBP 3rd party	GitHub
Format	Command-line executable	
Curation Status	Not applicable	
Validation - QC	Pass	Continuous Integration and Code Review Process QC owner: NEST Initiative (Hans E. Plesser, President)
Validation – Users	Yes	See D11.3.3, Sec 6.3
Validation – Publications	Yes	Over 40 publications based on NEST simulations in 2016 and 2017, see http://www.nest-simulator.org/publications
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	GPL v2 or later	https://www.gnu.org/licenses/gpl.html
Component Access URL	http://github.com/nest/nest-simulator	
Technical documentation URL	http://nest.github.io/nest-simulator/	
Usage documentation URL	http://www.nest-simulator.org/documentation/	
Component dissemination material URL	http://www.nest-simulator.org	

3.1.8 Community contacts established related to NEST

Field Name	Field Content	Additional Information
ID	518	

Component Type	Report	
Contact	DIESMANN, Markus	
Component Description	Close contact with the user community to ensure that the NEST software development and evolution serves the needs of NEST users throughout the scientific community. This component fosters the contact between NEST developers and users within and outside the HBP.	
Latest Release	Not applicable	
TRL	Not applicable	
Location	Hosted by other non-HBP 3rd party	Websites, mailing lists, user workshops and conferences
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Yes	Over 100 postings on NEST mailing list with short reply times; NEST User Workshop 2016 and NEST Conference 2017 with 50-70 participants
Validation – Publications	No	
Privacy Constraints	No privacy constraints	
Sharing	Not applicable	
Licence	Not applicable	
Component Access URL	http://www.nest-simulator.org/community/	Past events: https://indico-jsc.fz-juelich.de/event/52/ https://indico-jsc.fz-juelich.de/e/nest2016
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.1.9 NEST Requirements Management

Field Name	Field Content	Additional Information
ID	622	

Component Type	Service	
Contact	PLESSER, Hans Ekkehard	
Component Description	This component provides a central contact point for scientists requiring extensions and adaptations of the NEST simulator for their work. Requirements are then systematised and forwarded to the NEST developers. During SGA1, we have guided the implementation of several features requested by partners in the neuroscience Subprojects of the HBP.	
Latest Release	Not applicable	
TRL	Not applicable	
Location	Not applicable	
Format	Personal contact between requesters/users and the NEST team (T7.5.5)	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Yes	Positive feedback from users on implemented requirements
Validation – Publications	Not yet	Informal agreement with requesters/users to credit the developers in publications based on initial use of newly implemented features
Privacy Constraints	No privacy constraint	
Sharing	Anonymous / partner	Depending on the policy of the requesting partner, requests and follow-up actions may be kept confidential until requested components are integrated into public NEST release.
Licence	Not applicable	No license for requests per se, developed source code will follow NEST GPL v2 or later
Component Access URL	http://www.nest-simulator.org/community/	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	

Component dissemination material URL	Not applicable	

3.1.10 NEST Support for Providers

Field Name	Field Content	Additional Information
ID	661	
Component Type	Service	
Contact	PLESSER, Hans Ekkehard	
Component Description	This component provides support to computing centres and other service providers on issues related to installing and providing NEST as a simulation tool. In SGA1, we have in particular supported colleagues in the High-Performance Analytics and Computing Platform of the HBP in implementing *in situ* data analysis and visualisation protocols for cutting-edge hardware, and guided improvements to the NEST installation and help generation mechanism.	
Latest Release	Not applicable	
TRL	Not applicable	
Location	Not applicable	
Format	Personal contact between providers and the NEST team (T7.5.5)	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Yes	Regular meetings with pertaining developer groups; confirmation from users that installation problems are solved
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraints	
Sharing	Anonymous / partner	Depending on the policy of the collaborating parties, requests and follow-up actions may be kept confidential during development; all changes will be integrated into public NEST releases
Licence	Not applicable	No license for collaboration efforts *per se*, developed source code integrated into NEST will follow NEST GPL v2 or later

Component Access URL	http://www.nest-simulator.org/community/	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.1.11 NEST Support for Modellers

Field Name	Field Content	Additional Information
ID	660	
Component Type	Service	
Contact	PLESSER, Hans Ekkehard	
Component Description	This component provides support to groups using NEST as simulation tool, including advice on porting models to and implementing models in NEST. In SGA1, we successfully supported several groups in the neuroscience Subprojects of the HBP in porting models from the literature or other simulators to NEST.	
Latest Release	Not applicable	
TRL	Not applicable	
Location	Not applicable	
Format	Personal contact between modellers and the NEST team (T7.5.5)	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Yes	Users were able to perform the simulations they desired using NEST.
Validation – Publications	Not yet	Informal agreement with users to credit developers in publications based on initial use of newly implemented features
Privacy Constraints	No privacy constraints	
Sharing	Anonymous / partner	Depending on the policy of the collaborating parties, requests and follow-up actions may be kept confidential during development; all

		changes will be integrated into public NEST releases
Licence	Not applicable	No license for collaboration efforts per se, developed source code integrated into NEST will follow NEST GPL v2 or later
Component Access URL	http://www.nest-simulator.org/community/	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.2 Data-intensive supercomputing technology

All tests and performance explorations related to components listed in this section have been performed on the pilot systems JULIA and JURON installed at Jülich Supercomputing Centre (JSC). JULIA is a Cray CS-400 system with four DataWarp nodes integrated, each equipped with two Intel P3600 NVMe drives. These and the compute nodes were integrated in an Omnipath network. Ceph was deployed on the DataWarp nodes. The other system, JURON, is based on IBM Power S822LC HPC ("Minsky") servers. Each server comprises a HGST Ultrastar SN100 card. On this system BeeGFS, DSS and different key value stores (Cassandra and Scylla) were deployed.

More information about the pilot systems is available in the HPAC Platform Guidebook (https://hbp-hpc-platform.fz-juelich.de/).

3.2.1 Hierarchical data store software components (prototype)

Field Name	Field Content	Additional Information
ID	523	
Component Type	Software	Software library/API
Contact	ODEN, Lena (Library and tools for handling data in object storage)	CUGNASCO, Cesare (Software facilitating usage of key-value data stores)
Component Description	**[1] Library and tools for handling data in object storage:** A library (Python/C++) to handle scientific data in an object storage like CEPH, using the RADOS interface. Tools to translate HDF5 files into RADOS objects **[2] Software facilitating usage of key-value data stores:** This component contains the software necessary to facilitate the usage of key-value data stores on HPC infrastructures. It is composed of three main modules:	

	• a library implementing an interface with key-value databases (named Hecuba) • a set of scripts to automatically deploy and configure databases on allocations of HPC infrastructures (BigData4HPC) • a system providing support to multi-dimensional queries and to approximated queries (Qbeast)	
Latest Release	[1] Not applicable [2] 0.1.1	
TRL	[1] TLR 3-4 [2] TRL 4 (for Hecuba and BigData4HPC) TRL 6 (for Qbeast)	
Location	Hosted by other non-HBP 3rd party	Github repository
Format	[1] Specific library, command-line tool [2] Library, scripts and system	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	[1] Consortium [2] Anonymous	
Licence	[1] GPLv2/GPLv3 [2] Apache v2 license (Hecuba and BigData4HPC) Closed source, contact software owner (for Qbeast)	
Component Access URL	[1] https://github.com/LenaO/librados_science [2] https://github.com/bsc-dd/	
Technical documentation URL	[1] https://github.com/LenaO/librados_science [2] https://github.com/bsc-dd/hecuba/wiki	

Usage documentation URL	[1] https://github.com/LenaO/librados_science [2] https://github.com/bsc-dd/hecuba and https://github.com/bsc-dd/BigData4HPC	
Component dissemination material URL	Not available	

3.2.2 Evaluation of the hierarchical data store prototypes

Field Name	Field Content	Additional Information
ID	524	
Component Type	Report	
Contact	PLEITER, Dirk	
Component Description	Traditional storage technologies continue to feature impressive growth in terms of capacity. Compared to the increase of compute performance, speed of these storage devices is improving at a much slower pace. This problem can be mitigated by integrating new types of storage devices based on non-volatile memory. The report summarises the results from evaluating different technologies exploiting these new types of storage devices.	
Latest Release	1.0, 31 March 2018	
TRL	Not applicable	
Location	Collab	
Format	Technical report	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	
Licence	All Rights Reserved, Copyright	
Component Access URL	https://emdesk.humanbrainproject.eu/shared/5a83f3dc331eb-	

	a2a91d037d5852bec3bec4cba39b69bc (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66954 (Collaboratory)	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not available	

3.2.3 Report on hierarchical data store software components design specification

Field Name	Field Content	Additional Information
ID	522	
Component Type	Report	
Contact	PLEITER, Dirk	
Component Description	Traditional storage technologies continue to feature impressive growth in terms of capacity. Compared to the increase of compute performance, speed of these storage devices is improving at a much slower pace. This problem can be mitigated by integrating new types of storage devices based on non-volatile memory. To meet both, performance as well as capacity requirements, it is thus unavoidable to move towards hierarchical storage architectures. The report provides guiding design principles for deploying hierarchical storage within HPAC.	
Latest Release	1.0 (31 March 2018)	
TRL	Not applicable	
Location	collab	
Format	Technical report	
Curation Status	Unchecked	
Validation - QC	Not applicable	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	
Licence	All Rights Reserved, Copyright	

Component Access URL	https://emdesk.humanbrainproject.eu/shared/5a83f3dc331eb-a2a91d037d5852bec3bec4cba39b69bc (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66954 (Collaboratory)	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not Available	

3.2.4 Identification of basic data types for coupling data analytics and visualisation to simulation

Field Name	Field Content	Additional Information
ID	525	
Component Type	Software	
Contact	FAVREAU, Cyrille	
Component Description	To enable interactive steering of future brain simulators, it is crucial to couple these to data analytics applications and visualisation pipelines such that results can be presented to the user within a sufficiently short amount of time to allow for interactive feedback loops. The basic data types that have been identified for coupling data analytics and visualisation to simulation are described in the Brion I/O library documentation	
Latest Release	2.0.0, 6 June 2017	
TRL	TRL 5	
Location	Non-HBP third party	
Format	Specific library	
Curation Status	Unchecked	
Validation - QC	No	
Validation – Users	Not applicable	
Validation – Publications	No privacy constraint	
Privacy Constraints	Anonymous	

Sharing	Not applicable	
Licence	Not applicable	
Component Access URL	http://bluebrain.github.io/Brion-1.10/data.html	
Technical documentation URL	Not applicable	
Usage documentation URL	Not Available	
Component dissemination material URL	None	

3.2.5 Data store sources for interactive visualisation for selected data types

Field Name	Field Content	Additional Information
ID	526	
Component Type	Software	
Contact	FAVREAU, Cyrille	
Component Description	Brion provides two libraries Brion and Brain. The former is a collection of file readers and writers intended for low level access to the data model. The latter is a set of higher level classes that wrap low level data objects with a use-case oriented API. Keyv is a Key-Value based storage and caching using a variety of back ends. Keyv provides a unified C++ keyv::Map front end to store data in ceph, memcached and leveldb.	
Latest Release	2.0.0, 6 June 2017	
TRL	TRL 5	
Location	Non-HBP third party	
Format	Library	
Curation Status	Unchecked	
Validation - QC	No	
Validation – Users	No	
Validation – Publications	No privacy constraint	
Privacy Constraints	Anonymous	
Sharing	LGPLv2-LGPLv3	
Licence	https://github.com/BlueBrain/Brion	

	https://github.com/BlueBrain/Keyv	
Component Access URL	http://bluebrain.github.io	
Technical documentation URL	http://bluebrain.github.io	
Usage documentation URL	Not Available	
Component dissemination material URL	None	

3.2.6 Compute and data resource co-allocation

Field Name	Field Content	Additional Information
ID	528	
Component Type	Software	
Contact	WOLF, Felix	
Component Description	Based on the results of the report on data resource allocation schemes the strategies were implemented in simulators to facilitate simulations on the proposed co-allocation strategies. The foundations were laid to adapt the simulator also to real resource managers like SLURM.	
Latest Release	1.0.0, 31 March 2018	
TRL	TLR 3	Proof of Concept implementation
Location	Collab	
Format	Python libraries	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	
Licence	LGPLv3 and BSD license	
Component Access URL	https://emdesk.humanbrainproject.eu/shared/59b10b906f7f7-	

	2ecbbfa21b220f33b442154c4cc9054c (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66955 (Collaboratory)	
Technical documentation URL	https://emdesk.humanbrainproject.eu/shared/59b10b906f7f7-2ecbbfa21b220f33b442154c4cc9054c (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66955 (Collaboratory)	
Usage documentation URL	Not Available	
Component dissemination material URL	Not Available	

3.2.7 Evaluation of co-allocation prototype

Field Name	Field Content	Additional Information
ID	529	
Component Type	Report	
Contact	WOLF, Felix	
Component Description	The prototypes that facilitate the co-allocation strategies will be evaluated and compared in simulations, which will be used to model the real conditions of HPC systems and tested under large workloads inside the simulations. Depending on the outcome of this evaluation, next steps towards integration into relevant resource managers like SLURM were analysed.	
Latest Release	1.0.1, 9 June 2017	
TRL	Not Applicable	
Location	Not Applicable	
Format	Technical Report	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	

Sharing	Consortium	
Licence	All Rights Reserved, Copyright	
Component Access URL	https://emdesk.humanbrainproject.eu/shared/59b10b906f7f7-2ecbbfa21b220f33b442154c4cc9054c (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66955 (Collaboratory)	
Technical documentation URL	Not Available	
Usage documentation URL	Not Available	
Component dissemination material URL	Not Available	

3.2.8 Report on data resource allocation schemes and resource co-allocation strategies

Field Name	Field Content	Additional Information
ID	527	
Component Type	Report	
Contact	WOLF, Felix	
Component Description	The HPAC Platform needs to support both scalable computing, as well as to process extreme volumes of data. This will require the integration of precious compute and storage resources and the ability to schedule both types of resources in a way that efficient utilisation is maximised. This requires a significant change to the current paradigm of scheduling resources in supercomputers, which almost exclusively focuses on compute resources. As only limited work has been performed in the past to address this challenge, this activity will initially focus on laying the foundations by defining suitable data resource allocation schemes and identify resource co-allocation strategies.	
Latest Release	1.0.0, 31 March 2018	
TRL	Not applicable	
Location	Not applicable	
Format	Technical Report	
Curation Status	Not applicable	

Validation - QC	Unchecked	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No Privacy Constraint	
Sharing	Consortium	
Licence	All Rights Reserved, Copyright	
Component Access URL	https://emdesk.humanbrainproject.eu/shared/59b10b906f7f7-2ecbbfa21b220f33b442154c4cc9054c (public) https://collab.humanbrainproject.eu/#/collab/264/nav/66955 (Collaboratory)	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not Available	

3.2.9 Data analytics workflows for the human brain atlas

Field Name	Field Content	Additional Information
ID	530	
Component Type	Software	
Contact	DICKSCHEID, Timo	
Component Description	To establish an early end-to-end solution using technologies developed as part of this key result (see above), a full analysis workflow will be implemented that will exploit both, the HPC resources of the HPAC Platform and its federated data stores. This effort will also push cross-platform efforts as it requires interaction with metadata stored in neuro-informatics platform and contributes to the co-design project "Multi-level Human Brain Atlas". This activity will initially focus on workflows for well-established processes for image registration, segmentation, and low-level analysis in the context of the Big Brain and high-resolution PLI data. As part of this effort, specific requirements to the platform will be formulated, provided solutions be tested and gaps be identified.	
Latest Release	0.8	

TRL	TRL 3	
Location	Workflow installed in user environment on HPC systems JURECA and JURON. Code will be made available for the review on https://fz-juelich.sciebo.de/s/4cj7PUo9S7jXLLn	
Format	Highly optimized C/C++ and CUDA code, using the following libraries: • Gco MRF solver • Petsc • MPI • ITK • Boost • CUDA	
Curation Status	Not applicable	
Validation - QC	Manual software testing passed	
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Collab	
Licence	Closed source, contact software owner	
Component Access URL	Not applicable	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not Available	

3.3 Interactive visualisation and visual data analysis

3.3.1 Large-scale PLI data explorer

Field Name	Field Content	Additional Information

ID	2337	
Component Type	Software	
Contact	VIERJAHN, Thomas	
Component Description	3D-Polarized Light Imaging Viewer for interactive exploration of PLI data.	
Latest Release	1.1.0, 07 Feb 2018	
TRL	TRL 4-5	
Location	Hosted by Task	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	
Validation – Publications	Yes	
Privacy Constraints	No privacy constraints	
Sharing	Consortium	
Licence	MIT license	
Component Access URL	https://devhub.vr.rwth-aachen.de/VR-Group/pli_vis	
Technical documentation URL	https://devhub.vr.rwth-aachen.de/VR-Group/pli_vis	
Usage documentation URL	https://devhub.vr.rwth-aachen.de/VR-Group/pli_vis	
Component dissemination material URL	https://devhub.vr.rwth-aachen.de/VR-Group/pli_vis	

3.3.2 NEST-simulated spatial-point-neuron data visualisation

Field Name	Field Content	Additional Information
ID	2336	
Component Type	Software	
Contact	VIERJAHN, Thomas	
Component Description	Complementary to other viewers and visualisation implementations for NEST simulations, this component offers a rendering of activity and membrane potentials in a neural network simulated with NEST.	

Latest Release	Available on demand	
TRL	TRL 2	
Location	Hosted by task	
Format	Source code (C++)	
Curation Status	Not applicable	
Validation - QC	Pass	Unit test
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No Privacy Constraint	
Sharing	Collab	
Licence	Apache v2	
Component Access URL	Available on demand	
Technical documentation URL	None yet	
Usage documentation URL	None yet	
Component dissemination material URL	None yet	

3.3.3 NEST in situ framework

Field Name	Field Content	Additional Information
ID	547	
Component Type	Software	
Contact	VIERJAHN, Thomas	
Component Description	The NEST *in situ* framework implements *in situ* capabilities for the NEST simulator.	
Latest Release	Version 18.02.0, 07 Feb 2018	
TRL	TRL 3	
Location	Hosted by task	
Format	Source code (C++)	
Curation Status	Not applicable	

Validation - QC	Pass	Unit test
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No Privacy Constraint	
Sharing	Anonymous	
Licence	Apache v2	
Component Access URL	https://devhub.vr.rwth-aachen.de/VR-Group/nest-in-situ-vis	
Technical documentation URL	None yet	
Usage documentation URL	None yet	
Component dissemination material URL	None yet	

3.3.4 ZeroEQ

Field Name	Field Content	Additional Information
ID	231	
Component Type	Software	
Contact	LAPERE, Samuel	
Component Description	ZeroEQ is a cross-platform C++ library for modern messaging. It provides pub-sub and request-replay messaging using ZeroMQ. It also integrates REST APIs with JSON payload in C++ applications using an optional http::Server.	
Latest Release	0.9.0, 06 Feb 2018	
TRL	TRL 5	
Location	Non-HBP 3rd party	
Format	Not applicable	
Curation Status	Pass	Agile QA
Validation - QC	No	
Validation – Users	No	
Validation – Publications	No constraints	

Privacy Constraints	Anonymous	
Sharing	LGPLv2 / LGPLv3	
Licence	https://github.com/HBPVis/ZeroEQurl	
Component Access URL	https://hbpvis.github.io/ZeroEQ-0.9/index.html	
Technical documentation URL	https://hbpvis.github.io/ZeroEQ-0.9/index.html	
Usage documentation URL		
Component dissemination material URL		

3.3.5 RTNeuron

Field Name	Field Content	Additional Information
ID	543	
Component Type	Software	
Contact	LAPERE, Samuel	
Component Description	RTNeuron is a tool for interactive visualisation and media production for detailed neuronal circuit models and their simulation results.	
Latest Release	2.13.0, 8 Feb 2018	
TRL	TRL 7	
Location	Non-HBP 3rd party	
Format	Library, application	
Curation Status	Not applicable	
Validation - QC	Pass	Agile QA
Validation – Users	No	
Validation – Publications	No	
Privacy Constraints	No privacy constraints	
Sharing	Consortium	
Licence	Closed source, contact software owner	Open sourcing scheduled for June 2018

Component Access URL	https://bbpteam.epfl.ch/documentation/RTNeuron-2.13/index.html	
Technical documentation URL	https://bbpteam.epfl.ch/documentation/RTNeuron-2.13/index.html	
Usage documentation URL	https://bbpteam.epfl.ch/documentation/RTNeuron-2.13/index.html	
Component dissemination material URL	https://www.youtube.com/watch?v=wATHwvRFGz0	

3.3.6 VTK-m

Field Name	Field Content	Additional Information
ID	226	
Component Type	Software	
Contact	BIDDISCOMBE, John	
Component Description	C++ multithreaded visualisation library	
Latest Release	1.1, 21 Nov 2017	
TRL	TRL 6	
Location	Non-HBP 3rd party	
Format	Library	
Curation Status	Not applicable	
Validation - QC	No	
Validation – Users	No	
Validation – Publications	No privacy constraints	
Privacy Constraints	Anonymous	
Sharing	VTK-m Open Source License	
Licence	https://gitlab.kitware.com/vtk/vtk-m/blob/master/LICENSE.txt	
Component Access URL	https://gitlab.kitware.com/vtk/vtk-m	
Technical documentation URL	http://m.vtk.org/images/c/c8/VTKmUsersGuide.pdf	

Usage documentation URL	http://m.vtk.org/images/c/c8/VTKmUsersGuide.pdf	
Component dissemination material URL		

3.3.7 Task based analysis on HPX, VTK and VTK-m

Field Name	Field Content	Additional Information
ID	548	
Component Type	Software	
Contact	BIDDISCOMBE, John	
Component Description	C++ Parallelism and Concurrency Framework	
Latest Release	1.0	
TRL	TRL 9	
Location	Non-HBP 3rd party	
Format	Library	
Curation Status	Not applicable	
Validation - QC	No	
Validation – Users	No	
Validation – Publications	No privacy constraints	
Privacy Constraints	Anonymous	
Sharing	Boost Software License 1.0	
Licence	https://github.com/STEllAR-GROUP/hpx/blob/master/LICENSE_1_0.txt	
Component Access URL	https://github.com/STEllAR-GROUP/hpx	
Technical documentation URL	https://stellar-group.github.io/hpx/docs/html/	
Usage documentation URL	https://stellar-group.github.io/hpx/docs/html/	
Component dissemination material URL		

3.3.8 MSPViz

Field Name	Field Content	Additional Information
ID	234	
Component Type	Software	
Contact	BRITO, Juan-Pedro	
Component Description	MSPViz is a visualisation tool for structural plasticity models. It uses a novel visualisation technique based on the representation of the neuronal information through the use of abstract levels and a set of representations into each level.	
Latest Release	0.2.6, March 2018	
TRL	TRL 5	
Location	Hosted by non-HBP party	GitHub
Format	Software	Web-based application
Curation Status	Not applicable	
Validation - QC	Unchecked	Tests with a statistically significant number of users
Validation – Users	Yes	1 internal user
Validation – Publications	Yes	DOI: 10.3389/fnana.2016.00057
Privacy Constraints	No privacy constraints	
Sharing	Anonymus	
Licence	GPLv-3.0	
Component Access URL	https://github.com/gmrvvis/MSPViz	
Technical documentation URL	https://github.com/gmrvvis/MSPViz	
Usage documentation URL	https://github.com/gmrvvis/MSPViz	
Component dissemination material URL	http://gmrv.es/gmrvvis	

3.3.9 NeuroLOTs

Field Name	Field Content	Additional Information
ID	232	
Component Type	Software	

Contact	TOHARIA, Pablo	
Component Description	NeuroLOTs is a set of libraries and tools for generating 3D meshes that approximate the anatomy of neurons and brain vasculature, allowing to visualise them at different levels of detail using GPU-based tessellation.	
Latest Release	NeuroLots 0.2.0, March 2018 NeuroTessMesh 0.0.1, March 2018	
TRL	TRL 5	
Location	Not applicable	
Format	Software	
Curation Status	Not applicable	
Validation - QC	Unchecked	Tests with a statistically significant number of users
Validation – Users	Yes	External
Validation – Publications	Yes	DOI: https://dx.doi.org/10.3389/fninf.2017.00038
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	GPLv-3.0	
Component Access URL	https://github.com/gmrvvis/neurolots https://github.com/gmrvvis/NeuroTessMesh	
Technical documentation URL	https://gmrvvis.github.io/doc/neurolots/	
Usage documentation URL	https://github.com/gmrvvis/neurolots/blob/master/README.md http://gmrv.es/neurotessmesh/NeuroTessMeshUserManual.pdf	
Component dissemination material URL	http://gmrv.es/gmrvvis/neurolots/	

3.3.10 NeuroScheme

Field Name	Field Content	Additional Information

ID	233	
Component Type	Software	
Contact	TOHARIA, Pablo	
Component Description	NeuroScheme is a tool that allows users to navigate through circuit data at different levels of abstraction using schematic representations for a fast and precise interpretation of data. It also allows filtering, sorting and selections at the different levels of abstraction.	
Latest Release	NeuroScheme 0.2, March 2018	
TRL	TRL 6	
Location	Not applicable	
Format	Software	
Curation Status	Not Applicable	
Validation - QC	Unchecked	Tests with a statistically significant number of users
Validation – Users	Yes	4 Internal users / direct relation with users.
Validation – Publications	Yes	DOI: 10.2312/ceig.20151208 DOI: 10.3389/finf.2016.00044
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	GPLv-3.0	
Component Access URL	https://github.com/gmrvvis/NeuroScheme	
Technical documentation URL	https://github.com/gmrvvis/NeuroScheme	
Usage documentation URL	https://github.com/gmrvvis/NeuroScheme	
Component dissemination material URL	http://gmrv.es/gmrvvis	

3.3.11 Software library for multi-view analysis of neuroscientific data

Field Name	Field Content	Additional Information
ID	549	
Component Type	Software	

Contact	WEYERS, Benjamin	
Component Description	This component is composed of the three libraries nett, nett-python, and nett-connect, which offer dynamic creation of coordinated multiple-view applications. It uses a semantic (ontology-based) description of services to be connected and offers a GUI for interactive creation of coordinated multiple-view systems.	
Latest Release	November 2017	
TRL	TRL 4	
Location	Not applicable	
Format	Software	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	
Validation – Publications	(Yes)	Publication is under review
Privacy Constraints	No privacy constraints	
Sharing	Public authenticated	
Licence	Closed source, contact software owner	Planned to be changed at the end of SGA1 to BSD or LGPL
Component Access URL	https://devhub.vr.rwth-aachen.de/cnowke/nett-connect	
Technical documentation URL	https://devhub.vr.rwth-aachen.de/cnowke/nett-connect	
Usage documentation URL	https://devhub.vr.rwth-aachen.de/cnowke/nett-connect	
Component dissemination material URL		

3.4 Dynamic resource management tools and techniques

3.4.1 CoreNeuron with enabled malleability

Field Name	Field Content	Additional Information
ID	537	
Component Type	Software	

Contact	SIRVENT, Raül	
Component Description	This software is a modification of the CoreNeuron simulator to enable changing the resources it uses at runtime (malleability). It is published as a branch of the GitHub CoreNeuron repository.	
Latest Release	19 December 2017	
TRL	TRL 5	Tested on MareNostrum IV supercomputer, with realistic inputs, but not used in production
Location	Hosted by other non-HBP 3rd party	Public on GitHub
Format	Service	
Curation Status	Not applicable	
Validation - QC	Pass	LOPEZ, Víctor. Checkpoint QA
Validation – Users	No	
Validation – Publications	Yes	Paper submitted to ISC 2018
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	BSD 3-clause	
Component Access URL	https://github.com/BlueBrain/CoreNeuron/tree/hbp_dlb	
Technical documentation URL	Not available	
Usage documentation URL	Not available	
Component dissemination material URL	https://pm.bsc.es/dlb	

3.4.2 NEST with enabled malleability

Field Name	Field Content	Additional Information
ID	536	
Component Type	Software	
Contact	SIRVENT, Raül	
Component Description	This software is a modification of the NEST simulator to enable changing the resources it uses at runtime (malleability). It is published as a branch of the GitHub NEST repository.	

Latest Release	26 June 2017	
TRL	TRL 5	Tested on MareNostrum IV supercomputer, with realistic inputs, but not used in production
Location	Hosted by other non-HBP 3rd party	Public in GitHub
Format	Service	
Curation Status	N/A	
Validation - QC	Pass	GARCIA, Marta. Checkpoint QA
Validation – Users	No	
Validation – Publications	Yes	Paper submitted to ISC 2018
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	GPLv2	
Component Access URL	https://github.com/mggasulla/nest-simulator/tree/malleability	
Technical documentation URL	Not available	
Usage documentation URL	Not available	
Component dissemination material URL	https://pm.bsc.es/dlb	

3.4.3 Definition of APIs for job schedulers

Field Name	Field Content	Additional Information
ID	538	
Component Type	Report	
Contact	CORBALÁN, Julita	
Component Description	Internal Report describing the APIs defined in the four level architecture designed to enable dynamic resource management	
Latest Release	2 June 2017	
TRL	Not applicable	

Location	Hosted on Collaboratory storage	SP7 Collaboratory: Internal Reports – SGA1
Format	Report	Internal Report
Curation Status	Not applicable	
Validation - QC	Pass	Editorial review of internal report
Validation – Users	Not applicable	
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	CC0 Public Domain	
Component Access URL	https://collab.humanbrainproject.eu/#/collab/264/nav/28730 (Collaboratory) https://emdesk.humanbrainproject.eu/shared/59b10b86b2e6c-d329ab08281a67adadb4a165057c7bb9 (public)	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	https://pm.bsc.es/dlb	

3.4.4 Job scheduler with dynamic resource management capabilities

Field Name	Field Content	Additional Information
ID	539	
Component Type	Software	
Contact	CORBALÁN, Julita	
Component Description	Modified version of SLURM with dynamic resource management capabilities implemented	
Latest Release	February 2018	
TRL	TRL 5	Tested on MareNostrum IV with both CoreNeuron and NEST malleable versions

Location	Hosted by task providing dataset	BSC-internal GitLab
Format	Source Code	
Curation Status	Not applicable	
Validation - QC	Pass	D'AMICO, Marco. Checkpoint QA
Validation – Users	Yes	2 BSC internal users
Validation – Publications	Yes	Paper in JSSPP'17 (IPDPS'17). Submitted paper to ISC 2018.
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	But not publicly published yet
Licence	GPLv2	SLURM's license
Component Access URL	https://pm.bsc.es/gitlab/mdamico/slurm/tree/slurm-dlb	BSC-internal access only
Technical documentation URL	Not available	
Usage documentation URL	https://pm.bsc.es/dlb-docs/doxygen/dlb__drom_8h.html	Job Scheduler level API
Component dissemination material URL	https://pm.bsc.es/dlb	

3.4.5 Job scheduler with new scheduling policies

Field Name	Field Content	Additional Information
ID	541	
Component Type	Software	
Contact	CORBALÁN, Julita	
Component Description	Modified version of SLURM with dynamic resource management capabilities implemented adding two generic scheduling policies and one specific policy for CoreNeuron	
Latest Release	March 2018	
TRL	TRL 5	Tested on MareNostrum IV with both CoreNeuron and NEST malleable versions
Location	Hosted by task providing dataset	BSC internal GitLab

Format	Source Code	
Curation Status	Not applicable	
Validation - QC	Pass	D'AMICO, Marco. Checkpoint QA
Validation – Users	Yes	BSC-internal users
Validation – Publications	No	
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	But not publicly published yet
Licence	GPLv2	SLURM's license
Component Access URL	https://pm.bsc.es/gitlab/mdamico/slurm/tree/slurm-dlb	BSC Internal access only
Technical documentation URL	Not available	
Usage documentation URL	Not available	
Component dissemination material URL	https://pm.bsc.es/dlb	

3.4.6 Dynamic Load Balancing Library

Field Name	Field Content	Additional Information
ID	223	
Component Type	Software	
Contact	SIRVENT, Raül	
Component Description	DLB is a library devoted to speedup hybrid parallel applications. At the same time, DLB improves the efficient use of the computational resources inside a computing node. The DLB library will improve the load balance of the outer level of parallelism by redistributing the computational resources at the inner level of parallelism. This readjustment of resources will be done at dynamically at runtime. This dynamism allows DLB to react to different sources of imbalance: Algorithm, data, hardware architecture and resource availability among others.	
Latest Release	21 December 2017	https://collab.humanbrainproject.eu/#/collab/19/nav/2108?state=software,Dynamic-Load-Balancing
TRL	TRL 6	Used on MareNostrum IV supercomputer for some applications
Location	Hosted by other HBP party	Public on GitHub

Format	Library	
Curation Status	Not applicable	
<Validation - QC	Pass	GARCIA, Marta. Checkpoint QA
Validation – Users	Yes	9 users from different domains
Validation – Publications	Yes	13 citations: https://scholar.google.es/scholar?oi=bibs&hl=ca&cites=11364401518038771379&as_sdt=5
Privacy Constraints	No privacy constraints	
Sharing	Anonymous	
Licence	LGPLv3	
Component Access URL	https://pm.bsc.es/dlb https://github.com/bsc-pm/dlb	
Technical documentation URL	Not available	
Usage documentation URL	https://pm.bsc.es/dlb-docs/user-guide/	
Component dissemination material URL	https://hbp-hpc-platform.fz-juelich.de/?hbp_software=dynamic-load-balancing https://pm.bsc.es/sites/default/files/ftp/dlb/doc/Tutorial_DLB.pdf	

3.5 High Performance Analytics and Computing Platform v2

3.5.1 SP7 Federated HPAC Data and Computing Services

Field Name	Field Content	Additional Information
ID	329	
Component Type	Service	
Contact	LIPPERT, Thomas	
Component Description	This component summarises all resources and services of the High Performance Analytics and Computing Platform. The high performance computing resources, WAN VPN network, large-scale data storage systems, and monitoring resources including security components, the authentication and authorisation infrastructure, security devices like intrusion detection/prevention systems, firewalls as well as the defined security policies and processes and CSIRT teams all together are part of the HPAC infrastructure. This infrastructure has	

	been provided to the HBP community in a production ready manner and as one globally managed research infrastructure.	
Latest Release	Not applicable	Elements of the HPAC Platform are released once ready
TRL	Not applicable	The components of the HPAC Platform all have different maturity levels, see the tables below for details.
Location	Hosted by HPAC Platform	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Yes	See sections below for the quality control of the different HPAC Platform elements
Validation – Users	Yes	The HPAC Platform has users from all other HBP Subprojects, who directly or indirectly (by using other Platforms running on top of it, e.g. the Neurorobotics Platform) use it for their R&D.
Validation – Publications	Yes	See below for publications related to the elements of the HPAC Platform
Privacy Constraints	No privacy constraints	
Sharing	The services of the HPAC Platform have different sharing classes; see sections below for details.	
Licence	See below for details on licenses of the HPAC Platform elements	
Component Access URL	https://hbp-hpc-platform.fz-juelich.de/ https://collab.humanbrainproject.eu/#/collab/264	
Technical documentation URL	https://hbp-hpc-platform.fz-juelich.de/	
Usage documentation URL	https://hbp-hpc-platform.fz-juelich.de/	
Component dissemination material URL	https://hbp-hpc-platform.fz-juelich.de/ https://collab.humanbrainproject.eu/#/collab/264	

	Twitter: https://twitter.com/HBPHighPerfComp	

3.5.2 HPAC Monitoring Service

Field Name	Field Content	Additional Information
ID	566	
Component Type	Service	
Contact	NIEDERBERGER, Ralph	
Component Description	The HPAC Monitoring Service will include the underlying network infrastructure, the servers providing the HPAC service, including gateway systems and HPC systems themselves, as well as the services running on these systems, like Unicore, gridftp service, ssh service, and relevant storage systems and services like backup and archive service. Furthermore, a maintenance information board is included where users can check service availability, i.e. in maintenance, production, in error state, or down. A monitoring service for the UNICORE component as well as the network infrastructure has been setup already, as well as a maintenance information board (web page). Monitoring of the other services is done at all local partner sites, but has not (yet) been integrated into HBP monitoring of services.	
Latest Release	0.9.1, January 2018	
TRL	TRL 7	
Location	Collaboratory and web server at JUELICH-JSC	
Format	HTTPS	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	In particular developers of the HBP Platforms
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Public authenticated	Available in Collaboratory after authentication
Licence	No license chosen yet	
Component Access URL	https://collab.humanbrainproject.eu	"HBP maintenance" page: https://collab.humanbrainproject.eu/#/collab/7071/nav/54070 UNICORE monitoring: https://collab.humanbrainproject.eu/#/collab/264/nav/4307

		HBP network status: https://collab.humanbrainproject.eu/#/collab/7071/nav/54069
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	https access via Collaboratory
Component dissemination material URL	Not applicable	

3.5.3 VM Services

Field Name	Field Content	Additional Information
ID	968	
Component Type	Service	Includes underlying hardware infrastructure
Contact	HARDT, Marcus	
Component Description	The service provides OpenStack-based Infrastructure-as-a-Service, which enables users to configure and launch virtual machines and Docker containers. The service is integrated with the HBP Authorization and Authentication Infrastructure, allowing users to easily connect using their HBP Identity accounts, if they were granted access to the service. This component is integrated with component 1190 (KIT Cloud Storage Service) to the extent that VM images are stored in a highly available object storage.	
Latest Release	OpenStack Mitaka release, 07 April 2016	https://releases.openstack.org/mitaka/index.html
TRL	TRL 6-7	
Location	Hosted by other HBP party	KIT
Format	service	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	Eight users (granted access through HBP Identity Management Interface)
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Public authenticated	Available to anyone with an HBP Identity account who requested and was granted access to the service

Licence	Apache v2 license	(license of OpenStack software)
Component Access URL	https://oscloud-1.scc.kit.edu	Requesting access to the service is documented at: https://collab.humanbrainproject.eu/#/collab/264/nav/16576
Technical documentation URL	http://wiki.scc.kit.edu/lsdf/index.php/OpenStack_in_HBP	(for administrators)
Usage documentation URL	https://collab.humanbrainproject.eu/#/collab/264/nav/16576 https://docs.openstack.org/horizon/latest/user/index.html	Login procedure for the VM Services and official OpenStack user documentation of the dashboard
Component dissemination material URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=1326	

3.5.4 Container-based software packaging and deployment technologies

Field Name	Field Content	Additional Information
ID	562	
Component Type	Service	
Contact	MCMURTRIE, Colin	
Component Description	Providing support for container-based software packaging and deployment technologies (e.g. Docker, which can be considered as light-weight VMs) on production HPC systems	
Latest Release	Not applicable	
TRL	TRL 5	
Location	Service hosted by the HPAC Platform	
Format	Neurorobotics Platform has been containerized using Docker image format. Services are run through containers using: (1) Docker engine for components deployed under OpenStack; and (2) Shifter runtime for simulation components deployed on HPC systems	
Curation Status	Not applicable	
Validation - QC	Pass	

Validation – Users	Yes	External users of Neurorobotics Platform
Validation – Publications	Yes	
Privacy Constraints	Not applicable	
Sharing	Not applicable	
Licence	Not applicable	
Component Access URL	https://pollux.cscs.ch	
Technical documentation URL	https://user.cscs.ch/scientific_computing/supported_applications/shifter/advanced_shifter/	
Usage documentation URL	https://user.cscs.ch/scientific_computing/supported_applications/shifter/	
Component dissemination material URL	Not applicable	

3.5.5 PCP Pilot systems

Field Name	Field Content	Additional Information
ID	338	
Component Type	Hardware	
Contact	PLEITER, Dirk	
Component Description	The pilot systems JULIA and JURON are the result of the HBP Pre-Commercial Procurement (PCP) that took place during the Ramp-up Phase of HBP. They are meanwhile integrated into the HBP infrastructure. HBP scientists can get access to both HPC systems without having to go through a peer-review process as it is the case for getting HPC grants on the other, larger supercomputers. JULIA has been developed by Cray, JURON by a consortium of IBM and NVIDIA.	
Latest Release	In place since August (JULIA) and October (JURON) 2016	The software stack on both pilot systems is continuously updated and extended.
TRL	TRL 7	
Location	Hosted by HPAC Platform	
Format	Not applicable	
Curation Status	Not applicable	

Validation - QC	Pass	
Validation – Users	Yes	Several users from multiple HBP Subprojects as well as HPC experts from the HPAC Platform
Validation – Publications	No	
Privacy Constraints	No privacy constraints	
Sharing	Consortium	
Licence	Not applicable	All software and libraries installed come with their own licenses.
Component Access URL	JULIA: https://hbp-hpc-platform.fz-juelich.de/?page_id=1063 JURON: https://hbp-hpc-platform.fz-juelich.de/?page_id=1073	
Technical documentation URL	https://trac.version.fz-juelich.de/hbp-pcp/wiki/Public	Further technical documentation is available to the users
Usage documentation URL	https://trac.version.fz-juelich.de/hbp-pcp/wiki/Public	Further usage documentation is available to the users
Component dissemination material URL	http://www.fz-juelich.de/SharedDocs/Pressemitteilungen/UK/EN/2016/16-09-27hbp_pilotsysteme.html	

3.5.6 Active data repositories

Field Name	Field Content	Additional Information
ID	410	
Component Type	Service	
Contact	SCHULTHESS, Thomas	
Component Description	Active data repositories are defined as follows: • Close to computational or visualisation resources: high performance access to the data from these systems is possible • Not holding the main/master copy of the data (which is stored in an archive data repository), but holding temporary copies of it for improving the access performance • Available similarly to the attached computational or visualisation resources	

	• Able to replicate data from the master copy in case active repository is lost or corrupt. In this case there may be a performance impact, but no data loss.	
Latest Release	Not applicable	
TRL	TRL 9	
Location	Data hosted by HPAC Platform	
Format	The service relies on parallel file systems available on supercomputers at all HPAC sites	
Curation Status	Not applicable	
Validation - QC	Pass	
Validation – Users	Yes	
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraints	
Sharing	Not applicable	
Licence	Not applicable	
Component Access URL		Active data repositories can be accessed in different ways: e.g. from command-line interfaces or Jupyter notebooks in the Collaboratory; see documentation for more details
Technical documentation URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=676	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.5.7 Archive data repositories

Field Name	Field Content	Additional Information
ID	409	
Component Type	Service	
Contact	SCHULTHESS, Thomas	
Component Description	Archive data repositories are defined as follows:	

	• They are optimized for capacity, reliability and (tier-1) availability • They are used for storing large data objects permanently • They hold the main/master copy of the data • Data is not replicated to other sites and thus will be temporarily unavailable when the repository is unavailable	
Latest Release	Not applicable	
TRL	TRL 9	
Location	Data hosted by HPAC Platform	
Format	The service relies on Object Storage and archiving systems available at all HPAC sites	
Curation Status	Not applicable	
Validation - QC	Pass	
Validation – Users	Yes	
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraint	
Sharing	Not applicable	
Licence	Not applicable	
Component Access URL		Archive data repositories can be accessed in different ways, depending on how they are implemented by the different sites; see documentation for more details
Technical documentation URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=676	https://pollux.cscs.ch (for archive at ETHZ-CSCS)
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.5.8 HPAC Network Service

Field Name	Field Content	Additional Information
ID	334	
Component Type	Service	
Contact	NIEDERBERGER, Ralph	

Component Description	The HBP HPAC network infrastructure is based on the PRACE MDVPN connecting all the HBP HPAC sites via a virtual private network operated by GÉANT and the relevant national research and education networks (NRENs) of the HBP partner sites.	
Latest Release	Not applicable	
TRL	TRL 9	
Location	VPN defined on local HPAC networks at HBP partners involved in HPAC, corresponding NRENs and GÉANT infrastructure	
Format	IPv4 network	
Curation Status	Not applicable	
Validation - QC	Pass	
Validation – Users	Yes	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	See guidance in the Annex
Sharing	Consortium	Only HPC systems of the HPAC infrastructure can use this network (Virtual Private Network)
Licence	Not applicable	
Component Access URL	Not applicable	
Technical documentation URL	Not applicable	
Usage documentation URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=58	
Component dissemination material URL	Not applicable	

3.5.9 HPAC Reporting and Accounting Service

Field Name	Field Content	Additional Information
ID	400	
Component Type	Service	
Contact	NIEDERBERGER, Ralph	
Component Description	An LDAP server with a REST API on top is used as database for the administration of HBP users and groups of the HPAC Platform. These are created in a central LDAP and distributed to the account infrastructures	

	at each HPC centre. Ultimately, each centre conducts their local procedures for creating accounts on their HPC resources. Site local synchronization libraries import user and group data into the local administration systems. With that, each HBP user is assigned to a single HPC account name, which can be used on all participating HPC centres. Based on these accounts a storage and quota browser can be used to display used resources per user and project.	
Latest Release	2.0.0, 15 July 2016	
TRL	TRL 9	
Location	HPAC infrastructure at JSC	
Format	Software: OpenLDAP, JSC.hbpaccounting, REST-API for central LDAP server	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	JSC HPC users and CSCS users of HPAC
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	BSD	
Component Access URL	https://hbpacc.zam.kfa-juelich.de/	
Technical documentation URL	https://trac.version.fz-juelich.de/hbpaccounting/wiki	
Usage documentation URL	https://trac.version.fz-juelich.de/hbpaccounting/wiki	
Component dissemination material URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=66	

3.5.10 Transfer services

Field Name	Field Content	Additional Information
ID	416	
Component Type	Service	

Contact	MUCCI, Roberto	
Component Description	Services required for transferring datasets between the sites that are part of the data federation.	
Latest Release	UFTP 2.5.2 8 Jan 2018	
TRL	TRL 8	
Location	Hosted by HPAC platform (BSC, CINECA, ETHZ-CSCS, JUELICH-JSC)	
Format	Software: UNICORE File Transfer Daemon, UNICORE Commandline Client	
Curation Status	Not applicable	
Validation - QC	Pass	Bernd Schuller Agile QA
Validation – Users	Yes	Around 400 downloads from Sourceforge for the UFTPD server
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	Also available to non-HBP users of the HPAC Platform with required allocations
Licence	BSD license	
Component Access URL	BSC: https://unicore-hbp.bsc.es:8080/BSC-MareNostrum/services/StorageManagement CINECA: https://grid.hpc.cineca.it:9111/CINECA-MARCONI/services/StorageManagement JSC: https://hbp-unic.fz-juelich.de:7112/HBP_JURECA/services/StorageManagement	
Technical documentation URL	https://www.unicore.eu/documentation/	

Usage documentation URL	https://www.unicore.eu/docstore/ucc-7.9.0/ucc-manual.html https://collab.humanbrainproject.eu/#/collab/3656/nav/60667 https://collab.humanbrainproject.eu/#/collab/3656/nav/29210	
Component dissemination material URL		

3.5.11 HPAC Authentication and Authorisation Infrastructure Services

Field Name	Field Content	Additional Information
ID	336	
Component Type	Service	See guidance in the Annex
Contact	NIEDERBERGER, Ralph	See guidance in the Annex
Component Description	HPAC AAI is a set of services and processes that work together for allowing HBP end-users to authenticate using the HBP central OIDC server and to provide authorisation to access high-performance computing systems as well as file systems and data stores. HPAC AAI relies on external off-the-shelf components such as Unity and LDAP. Each HPAC site stays in full control of their resources, since authorisation is done locally (e.g. by using UNICORE's user mapping components)	
Latest Release	1.0 (Match 30, 2017)	
TRL	TRL 5	
Location	Hosted by HPAC Platform	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Pass	NIEDERBERGER, Ralph; Checkpoint quality assurance
Validation – Users	Yes	
Validation – Publications	Not applicable	
Privacy Constraints		
Sharing		
Licence	Not applicable	
Component Access URL	Not applicable	See documentation for details

Technical documentation URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=66	
Usage documentation URL	https://hbp-hpc-platform.fz-juelich.de/?page_id=66	
Component dissemination material URL	Not applicable	

3.5.12 HPAC UNICORE services

Field Name	Field Content	Additional Information
ID	792	
Component Type	Service	
Contact	SCHULLER, Bernd	
Component Description	UNICORE is a set of middleware services that work together to provide access to high-performance computing systems as well as file systems and data stores. UNICORE integrates with the existing facilities at a HPC centre as well has external authentication systems (such as the HBP OIDC service), and maps external users to their correct internal UNIX accounts and groups. UNICORE has functions for (batch) job submission and management, file and data access, file upload/download, third party transfer and more. UNICORE provides both SOAP/XML and REST APIs that can be used by a variety of clients including end-user Python code in the HBP Collaboratory.	
Latest Release	7.9.0 14 Nov 2017	
TRL	TRL 8	
Location	Not applicable	Part of HPAC Platform, hosted by HPC sites in HPAC
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Pass	SCHULLER, Bernd; Agile quality assurance
Validation – Users	Yes	
Validation – Publications	Not applicable	
Privacy Constraints		
Sharing		
Licence	BSD license	
Component Access URL	Not applicable	

Technical documentation URL	https://www.unicore.eu/documentation/	
Usage documentation URL	https://sourceforge.net/p/unicore/wiki/REST_API/	
Component dissemination material URL	Not applicable	

3.5.13 HPAC Security Services

Field Name	Field Content	Additional Information
ID	569	
Component Type	Service	
Contact	NIEDERBERGER, Ralph	
Component Description	The HPAC Security Service is composed of “Definition of security policies”, “Definition of security processes handling incidents” and “Security operations”, as there is, e.g. setting up of an HBP CSIRT and doing day to day operational security services, e.g. handling of any security incidences, secure integration of new services and components, security awareness, etc. The security policies exchanged between partners and agreed on via a self-assessment procedure as well as the interaction of security teams of the HPAC infrastructure partner sites, guarantee a “Net of Trust”, which allows to work securely in a collaborating environment.	
Latest Release	Not applicable	
TRL	TRL 9	
Location	At every HPAC infrastructure system and network component including firewalls, IDS systems etc.	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Partners of the HPAC infrastructure	
Licence	Not applicable	
Component Access URL	Not applicable	

Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.5.14 Arbor

Field Name	Field Content	Additional Information
ID	945	
Component Type	Software	
Contact	CUMMING, Benjamin	
Component Description	Arbor is a software library designed from the ground up for simulators of large networks of multi-compartment neurons on hybrid/accelerated/many core computer architectures. In the last year, performance portability was completed for the three main target HPC architectures available through the HBP: Intel x86 CPUs (AVX2 and AVX512), Intel KNL (AVX512) and NVIDIA GPUs (CUDA). Optimised kernels are automatically generated to target each architecture, and the system used in Arbor can be extended to new architectures in the future. The source code was released publicly on GitHub with an open source BSD license, along with documentation on *Read the Docs*, and automatic testing was set up on Travis CI. The other enhancements and features implemented in Arbor over this period are: • Fully parallelised event generation and queueing from spikes. • Efficient sampling of model state on CPU and GPU implementations, e.g. voltage and current. • Significant refactoring to prepare the code for general release. • A Python interface for users.	
Latest Release	0.1.0	
TRL	TRL 4	
Location	Hosted by other non-HBP party	Source hosted on GitHub and documentation on Read the Docs.
Format	Software in C++ with some Python	
Curation Status	Not applicable	

Validation - QC	Pass	Owner: Ben Cumming. Unit testing and validation performed using continuous integration.
Validation – Users	No	Some early testing and modelling performed by PhD students.
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Anonymous	
Licence	BSD license	
Component Access URL	https://github.com/eth-cscs/arbor	
Technical documentation URL	http://arbor.readthedocs.io	
Usage documentation URL	http://arbor.readthedocs.io	
Component dissemination material URL	Not applicable	

3.5.15 Application Software Services

Field Name	Field Content	Additional Information
ID	1071	
Component Type	Service	
Contact	CUMMING, Benjamin	
Component Description	This activity focuses on providing support for the migration of simulation codes to hybrid and/or accelerator-enabled architectures. During the last year, all work has been focussed on developing the Arbor library, in deliverable Arbor (see above).	
Latest Release	Not applicable	
TRL	Not applicable	
Location	Not applicable	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Not applicable	

Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraint	
Sharing	Not applicable	
Licence	Not applicable	
Component Access URL	Not applicable	hpac-support@humanbrainproject.eu
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.5.16 HPAC Ticket System

Field Name	Field Content	Additional Information
ID	1114	
Component Type	Software	
Contact	VICENTE, David	
Component Description	The HPAC Ticketing system is the tool used to manage all the requests and petitions generated for the users of the different HPAC services. The HPAC Ticket System provides the first and second level support for the HPAC infrastructure services.	
Latest Release	30 Sept 2016	
TRL	TRL 9	
Location	Hosted by HPAC platform	
Format	Software - service	
Curation Status	Not applicable	
Validation - QC	Unchecked	
Validation – Users	Yes	
Validation – Publications	No	
Privacy Constraints	No privacy constraint	
Sharing	Ticket system open only to the helpdesk teams of the HPC and data centres	

	Tickets can be opened by anyone inside and outside HBP sending an email to hpac-support@humanbrainproject.eu	
Licence	GPLv2	
Component Access URL	https://hpac-hbp.bsc.es/	
Technical documentation URL	https://collab.humanbrainproject.eu/#/collab/264/nav/12826	
Usage documentation URL	https://collab.humanbrainproject.eu/#/collab/264/nav/12826	
Component dissemination material URL	None	

3.5.17 HPAC User Support Services

Field Name	Field Content	Additional Information
ID	577	
Component Type	Service	
Contact	VICENTE, David	
Component Description	The user support service is the service providing the first and second level support to the HPAC infrastructure. This service uses HPAC Ticket System to manage the requests. The service is in charge of the 1st level support for all the HPAC services and to check track of the tickets when it requires the participation of the 2nd level support teams.	
Latest Release	30 Sept 2016	
TRL	TRL 9	
Location	Hosted by HPAC Platform	
Format	Not applicable	
Curation Status	Not applicable	
Validation - QC	Pass	Regular internal review of response times etc.
Validation – Users	Yes	Already multiple support request from inside and outside the HPAC Platform
Validation – Publications	No	
Privacy Constraints	No privacy constraints	

Sharing	The service is available to current and future users of the HPAC Platform	
Licence	Not applicable	
Component Access URL	hpac-support@humanbrainproject.eu	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	Not applicable	

3.5.18 HPAC Platform Guidebook

Field Name	Field Content	Additional Information
ID	337	
Component Type	Report	
Contact	LÜHRS, Anna	
Component Description	The HPAC Platform Guidebook is the collection of all technical and user documentation of the HPAC Platform. This website also contains a calendar of all relevant training and other events organised by the HPAC partners.	
Latest Release	Continuously updated	
TRL	Not applicable	
Location	Hosted by HPAC Platform	Web server at JUELICH-JSC
Format	Website, based on WordPress	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Yes	
Validation – Publications	No	
Privacy Constraints	No privacy constraints	
Sharing	Public	
Licence	Not applicable	

Component Access URL	https://hbp-hpc-platform.fz-juelich.de/	
Technical documentation URL	Not applicable	
Usage documentation URL	Not applicable	
Component dissemination material URL	https://hbp-hpc-platform.fz-juelich.de/	

3.5.19 *HPAC Platform validation*

Field Name	Field Content	Additional Information
ID	581	
Component Type	Service	
Contact	PEYSER, Alexander	
Component Description	The HPAC Platform Validation takes care of collecting and analysing relevant, representative use cases for the HPAC Platform from other HBP Subprojects. The template used for this analysis is regularly revised based on the feedback received during the process of describing new use cases together with the users.	
Latest Release	Continuous process	
TRL	Not applicable	
Location	Hosted by HPAC Platform	
Format	Reports	
Curation Status	Not applicable	
Validation - QC	Not applicable	
Validation – Users	Not applicable	
Validation – Publications	Not applicable	
Privacy Constraints	No privacy constraint	
Sharing	Consortium	Details of the use case analysis might be kept confidential if requested by the use case owner
Licence	Not applicable	
Component Access URL	Process description: https://collab.humanbrainproject.eu/#/collab/264/nav/38113	

	Use cases related to Fenix: https://emdesk.humanbrainproject.eu/shared/5a536d07b8b97-fe5aec20ddb941c283d042af128ef220 SGA1 use cases: https://emdesk.humanbrainproject.eu/shared/5a536cef0f608-f9693f5ac4f159b47b3b4511a4481a8e	
Technical documentation URL		
Usage documentation URL		
Component dissemination material URL		

4. Conclusion and Outlook

The role of the HPAC Platform in the Human Brain Project is to develop, integrate and operate the base infrastructure for the project, i.e. it provides the hardware and software components required to run large-scale, data-intensive, interactive brain simulations, to manage large amounts of data and to implement and to manage complex workflows comprising concurrent simulation, data analysis and visualisation workloads.

During the last year, important steps in this direction have been taken. The collaboration with the other HBP Platforms has been significantly intensified, so that now the Neuroinformatics Platform, the Brain Simulation Platform, the Neurorobotics Platform and the Collaboratory run (part of) their services on HPAC infrastructure. This required a very close collaboration and was a true co-designing effort. These collaborations will be continued in the next two years, and further collaborations will start soon. Also, due to the closer integration with the other Platforms, the number of HPAC Platform users has significantly grown in the last two years. A user support team and ticketing system have therefore been put in place to ensure that the users' requests and questions are solved fast and professionally.

Apart from the close integration with the other Platforms towards the HBP Joint Platform, the unification of the six Platforms, the focus in the next two years will be to migrate the HPAC Platform to run on top of the federated Fenix infrastructure so that the HPAC Platform takes care of the community-specific, platform-level interfaces to the generic Fenix services and resources. Fenix receives funding through the ICEI (Interactive Computing e-Infrastructure) project, which is about to start. It is built by the same four supercomputing and data centres that are part of the HPAC Platform, together with the TGCC of CEA that will hence also join the HPAC Platform in April 2018. The HPAC Platform will in particular develop software components in close relation to Fenix, for instance data location and transfer services.

The HPAC Platform builds on pre-existing technology wherever possible. In some areas the requirements from the neuroscience community are different as compared to those of other communities and are hence pushing the limits. One of these areas is data-intensive supercomputing, where therefore a lot of efforts have been spent on the development of new solutions, for example with respect to hierarchical data stores, coupling of data analytics and visualisation to simulation or the co-allocation of compute and data resources. While focussing initially on low-level architectural aspects related to the realisation of data-intensive applications on the HPAC Platform, the team started to deploy software components and provided proofs-of-concept for real-life workflows. This work will be continued in the next two years, where in particular the results for the co-allocation of compute and dense memory resources are foreseen as basis for the implementation of scheduler plug-ins. Additionally, development work related to accessing data within the federated Fenix infrastructure is planned.

For enabling interactive workflows, where, for instance, a visualisation or analysis are coupled to an already running simulation, the management of resources needs to be more dynamic than it is typically handled. Important steps have been made towards the enabling of dynamic resource management, where the resources assigned to an application can be adjusted at runtime as needed. The simulator NEST and CoreNeuron have been modified so that they support the change of their computing resources at runtime. An architecture has been defined covering all layers in the software stack, from the kernel to the job scheduler level, and APIs have been defined to enable the communication regarding malleability between the layers. The SLURM job scheduler has been used to implement the needed changes to enable the dynamicity aimed to achieve with malleability, both as a proof of concept and to use it for testing purposes to demonstrate our hypotheses. Changes have been well tracked so they can be applied to other job schedulers in the future if they want to support dynamic resource management. Two generic scheduling policies that use malleability have been defined and implemented. Also, a specific policy for CoreNeuron has been developed to enable the sharing of resources during its initialisation phase. Contacts with several vendors (IBM-NVIDIA, Cray, Lenovo and Intel) have been established to investigate the sustainability of our proposal, and overall the feedback has been very positive. Work on dynamic resource management will be continued in the next years.

Also at the application level, significant improvements could be achieved. The NEST simulator received major updates, for example to support rate models of neuronal networks, to eliminate critical bottlenecks in the construction of large-scale networks, the development of NESTML as domain-specific language and the integration with the simulator UG4 using MUSIC. The interactions with the community have been intensified, for example by means of the first NEST Conference. The focus in the next years will be on developments to make the simulator better suitable for Exascale architectures, for example solvers for phenomenological models or code generation for network components. The efforts with regards to multi-scale multi-simulator interaction will also be continued.

Also the new library Arbor (formerly called NestMC) for high-performance simulators has been advanced, for example the kernels have been optimised for different target architectures such as Intel KNL and NVIDIA GPUs. The source code and documentation are publicly available on GitHub. The development will be continued in the next years, where efforts will, for example, be spent on solvers for many-compartment models and optimisations for Exascale architectures.

The development of visualisation tools and methods has been continued during the last years, both at the framework and application layer. Several tools are already in use by the neuroscience community. A close collaboration with the HPC and data centre experts ensures that the tools can make use of the federated infrastructure. The integration of standard libraries for massively parallel visualisation has been started to address upcoming challenges for the visualisation of neuroscientific data. This aspect will be extended in the next years through community building efforts on visualisation support in HBP and to offer wider and more convenient access to visualisation solutions developed in the HBP, as well as open source tools and solutions existing outside.

Overall, we made very good progress during the last year, and processed mostly according to plan, so that we could achieve all our goals. The work will be continued for the next years, focusing on the development of components needed for the Fenix infrastructure, the migration of the HPAC Platform towards a unified platform running on top of Fenix, the development of Exascale-capable technology for brain simulations and of interactive data analytics and *in situ* visualisation technology integrated with the Fenix infrastructure, and the establishment of (even more) advanced user support services, training and community building. All these efforts will be driven by requirements from the HBP and neuroscience community, as well as by the needs of the other HBP Platforms. The creation of the HBP Joint Platform will be the overall theme for the next two years, and the HPAC Platform will hence also contribute to and closely collaborate with the HBP High-Level Support Team that will be established soon.

Annex A: HPAC Platform usage

Here, we present the share of "HPAC resources" used by HBP at the end of SGA1, i.e. of HPC systems, Cloud resources and storage. The usage of network bandwidth is not tracked by user or project, therefore, no usage numbers are available.

As can be seen, the usage of HPC systems by the HBP is moderate (up to 1.6% of the system capacity is used by HBP), whereas HBP is one of the main users of storage (up to 17% of the storage capacity at JUELICH-JSC is used by HBP). Overall, these numbers reemphasize our strategy to focus more strongly on data analytics, and not only on large-scale simulations.

Table 1: Usage of HPC systems by HBP during SGA1

For the pilot systems JULIA and JURON no accounting information is available. They were almost exclusively used by HBP.

System	Site	Usage by HBP [million core-h]	Usage by HBP [% of capacity]
JUQUEEN	JUELICH-JSC	91.08	1.32%
JURECA CPU		7.25	1.08%
JURECA Booster		0.25	0.05%
JURECA GPUs		0.2212	1.62%
JURECA Vix		0.0025	0.16%
JULIA		N/A	N/A
JURON		N/A	N/A
HPC systems	CINECA	0.17	~2.0%
MareNostrum 4	BSC	0.27	0.03%

Table 2: Usage of the HPC storage by HBP at the end of SGA1

System	Site	Usage by HBP [TB]	Usage by HBP [% of capacity]
JUST (Minimum)	JUELICH-JSC	470.25	4%
JUST (Average)		941.39	9%
JUST (Maximum)		1768.43	17%
HPC storage	CINECA	100.00	~1%
MareNostrum 4 storage	BSC	65.68	0.85%

ETHZ-CSCS started to provide OpenStack-based Infrastructure-as-a-Service to the HBP for hosting web services for other HBP Platforms, in particular the Collaboratory, the Brain Simulation Platform and the Neurorobotics Platform, see above sections for more details. The OpenStack *KeyStone* is integrated with *KeyCloak* to allow for the integration of the federated Authentication and Authorisation Infrastructure (bp0 accounts).

Project Name	Domain	VCPUs #				RAM (MB)			
		In-Use	%	Quota	%	In-Use	%	Quota	%
bp0	cscs	13	11	120	100	51200	20	256000	100
bp00cold	default	93	78	120	100	301056	75	400000	100
bp00colh	cscs	91	76	120	100	337920	66	512000	100
bp00sp01	cscs	0	-	0	100	0	-	0	100
bp00sp02	cscs	0	-	0	100	0	-	0	100
bp00sp03	cscs	0	-	0	100	0	-	0	100
bp00sp04	cscs	0	-	0	100	0	-	0	100
bp00sp05	cscs	0	-	0	100	0	-	0	100
bp00sp06	cscs	16	100	16	100	131072	94	140000	100
bp00sp07	cscs	0	0	20	100	0	0	51200	100
bp00sp10	cscs	420	64	652	100	1720320	60	2864291	100
TOTAL Aggregate		633		1048		2541568		4223491	

Project Name	Domain	Block Storage (GB)				Object Storage (GB)				Floating IP @ #			
		In-Use	%	Quota	%	In-Use	%	Total	%	In-Use	%	Quota	%
bp0	cscs	177	4	4096	100	0	0	0	100	3	30	10	100
bp00cold	default	4964	83	6000	100	0	0	0	100	19	95	20	100
bp00colh	cscs	5524	67	8192	100	0	0	0	100	11	55	20	100
bp00sp01	cscs	0	-	0	100	699	0.70	100000	100	0	-	0	100
bp00sp02	cscs	0	-	0	100	5855	5.86	100000	100	0	-	0	100
bp00sp03	cscs	0	-	0	100	10	0.01	100000	100	0	-	0	100
bp00sp04	cscs	0	-	0	100	0	0	100000	100	0	-	0	100
bp00sp05	cscs	0	-	0	100	2657	2.66	100000	100	0	-	0	100
bp00sp06	cscs	1000	91	1100	100	3689	3.69	100000	100	1	50	2	100
bp00sp07	cscs	9000	900	1000	100	0	0	0	100	0	0	50	100
bp00sp10	cscs	0	0	10168	100	0	0	0	100	81	81	100	100
TOTAL Aggregate		20665		30556		12910				115		202	

Figure 12: Usage of the OpenStack IaaS Cloud Infrastructure at ETHZ-CSCS by HBP at the end of SGA1

The project names indicate the user groups: The leading "bp00" indicates that these are HBP projects, according to the naming scheme agreed upon in the HPAC Platform. The remaining part specifies the actual user group. "cold" and "colh" are the Collaboratory development and production projects. "sp[xx]" denotes HBP Subprojects. The "bp0" project is the default project for HBP scientists.

ETHZ-CSCS made 100 TB of Swift Object Storage available to HBP, which is located closely to HPC resources. This Object Storage is used for hosting the data curated by the Neuroinformatics Platform. First data arrived in M22 of the SGA1, with a strong increase in the amount of data until the end of SGA1, when already about 13% of the capacity was used by HBP.

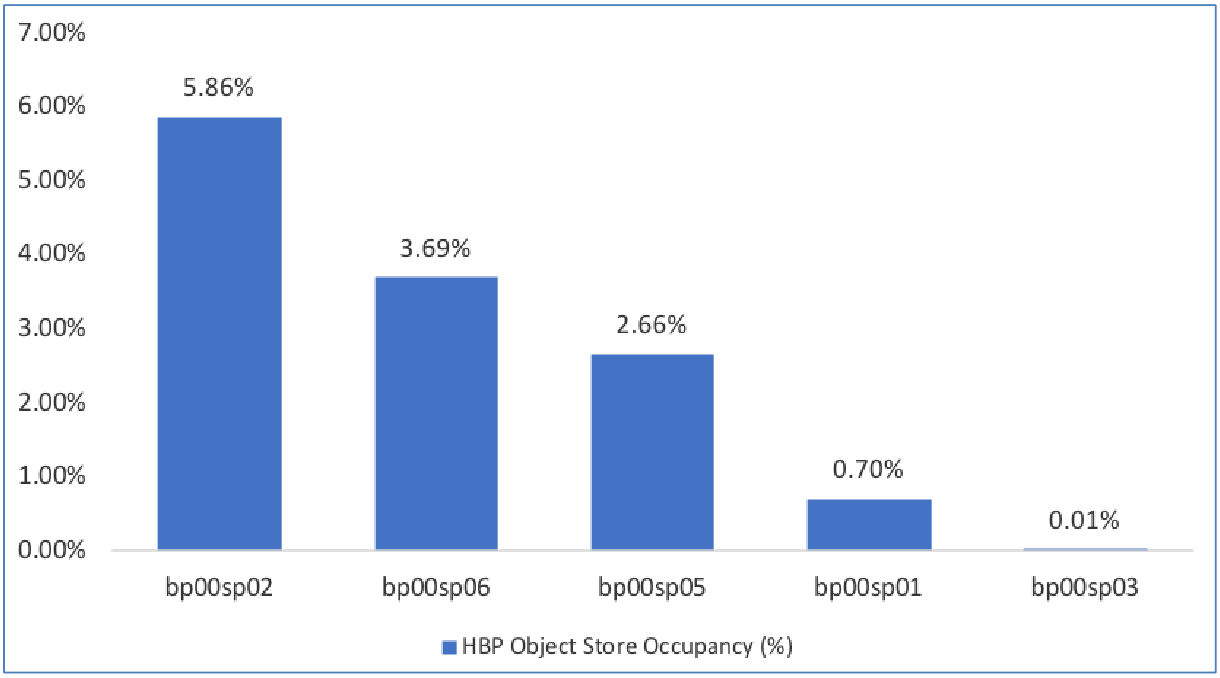

Figure 13: Usage of the Object Store at ETHZ-CSCS by HBP SPs at the end of SGA1

The Object Store at ETHZ-CSCS is used for storing HBP curated data. The project IDs are the same as in Figure 12.

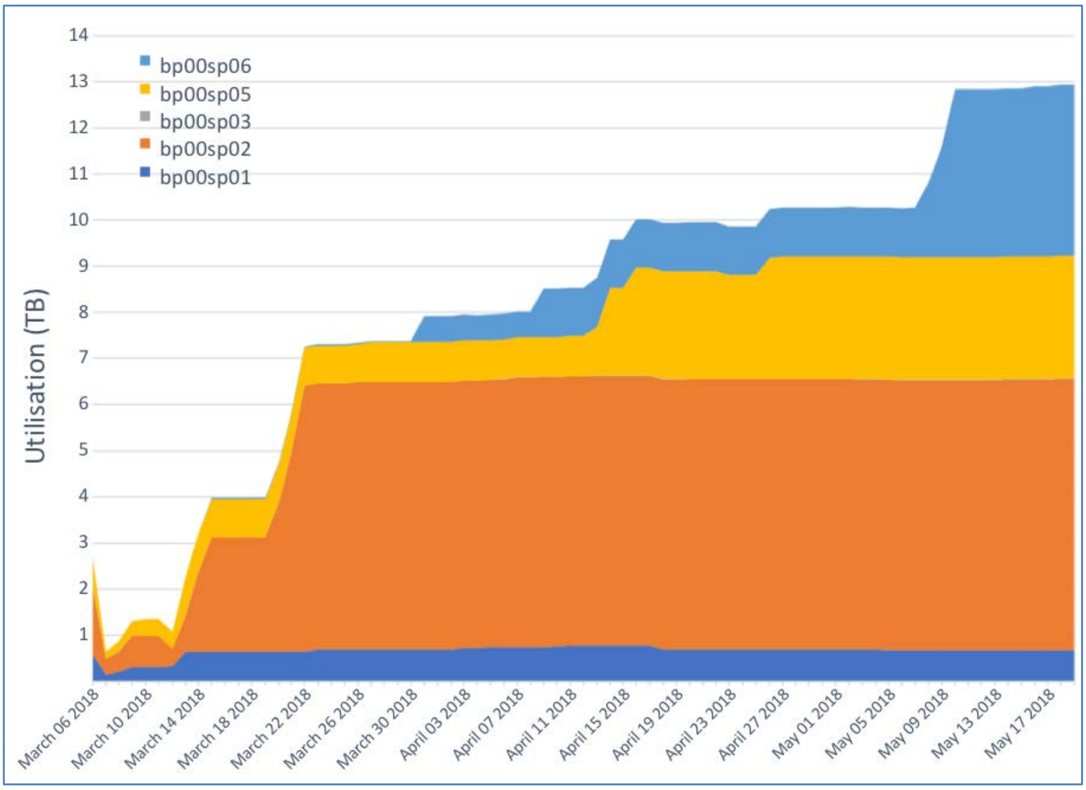

Figure 14: Evolution of Object Storage at ETHZ-CSCS usage by HBP (March - May 2018)