Neuroinformatics Platform Documentation

Technical Manual

Last update: 2016-09-15

Table of Content

Technical Manual

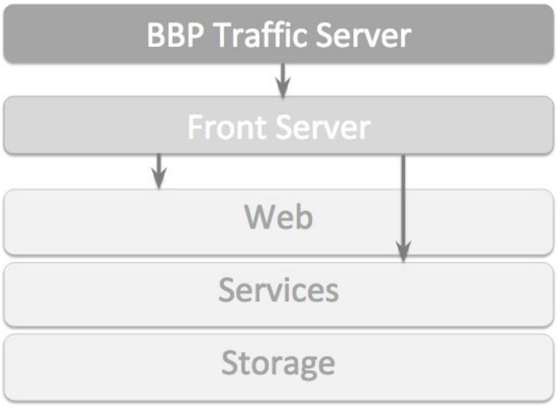
Intended Audience

This manual is intended for people who are interested to learn about the technical details of the NeuroInformatics Platform such as its architecture, the components we have developed and the data formats we are using.

Platform architecture

Overview

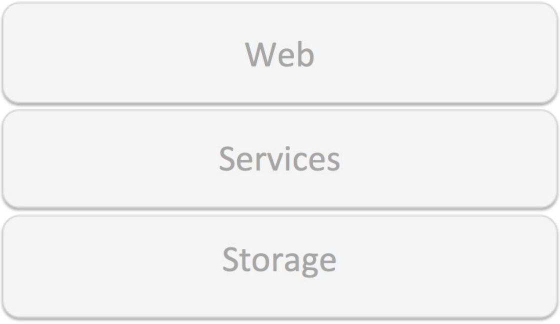
The Neuroinformatics Platform components are available under the nip.humanbrainproject.eu domain. It is hosted in the Blue Brain Project infrastructure as depicted below:



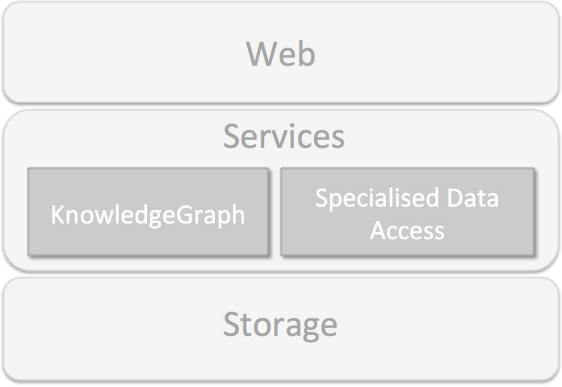
The Blue Brain Project manages the top-level humanbrainproject.eu domain. All traffic lands on the **Traffic Server**, which handles a first level of routing, caching and SSL handling across all the subdomains. This server runs the Apache HTTP server.

All traffic to the Neuroinformatics Platform components will be redirected to the nip.humanbrainproject.eu domain, which is managed by our **Front Server**. It acts as a single access point, handling routing to individual components (such as https://nip.humanbrainproject.eu/atlas), load balancing, and access control in some cases such as the Image Service. This server runs the NGINX HTTP server with Lua scripting extensions.

The Neuroinformatics Platform itself is built as a 3-tiers architecture as depicted below:



Before getting into the details of which components exist in each of these tiers, it is worth noting that there are two main responsibilities for the Services layer:



Let's detail a little more what these two sub classes of services are:

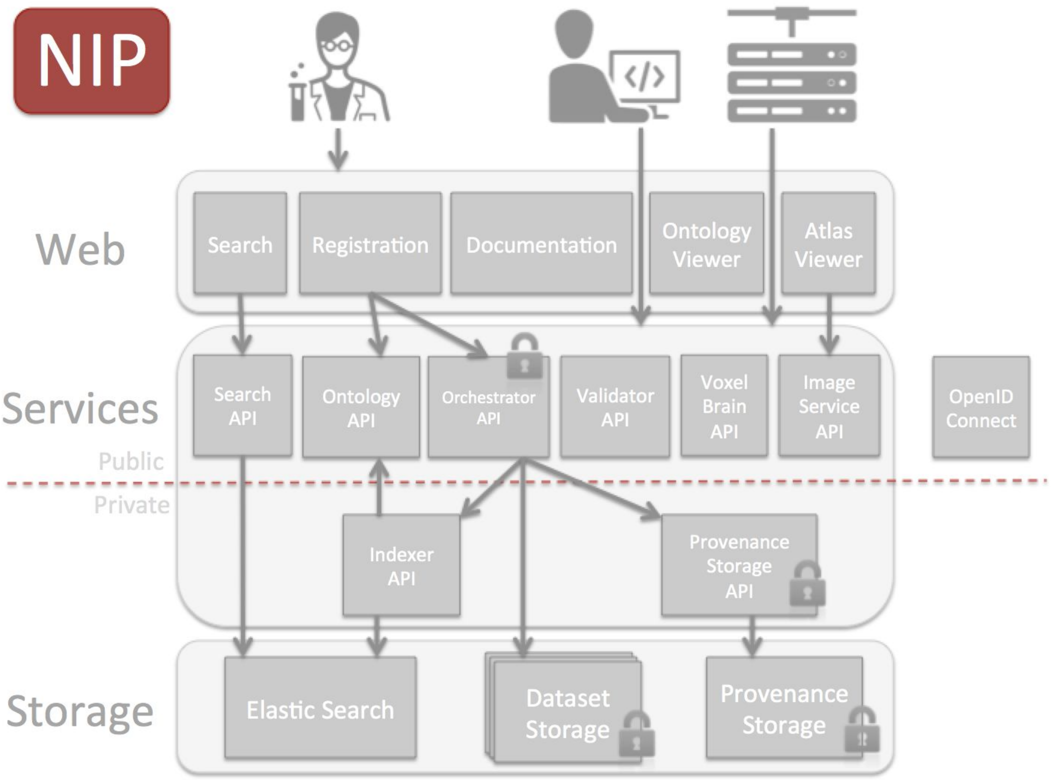
KnowledgeGraph

This component has for purpose to give access to the provenance and to all data integrated in the Neuroinformatics Platform, for the purpose of sharing it with the scientific community. It provides means for scientists to browse data using their favorite web browser while allowing developers and software components to access the same data programmatically. And for people working on the integration of data into the platform, it gives tools to describe the provenance (HBP-PROV) as well as workflows to integrate both provenance and raw data into the platform.

Specialised Data Access

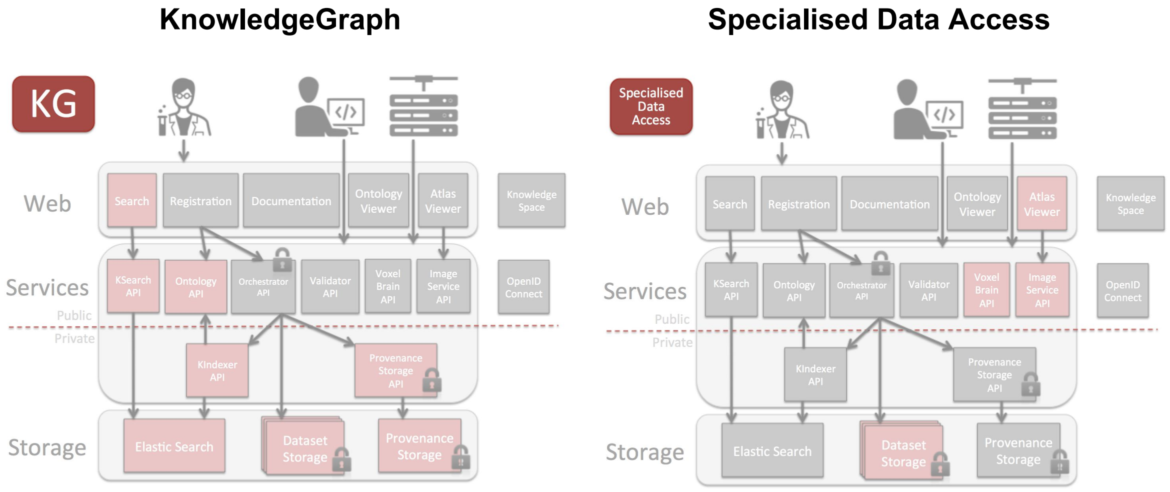
While the KnowledgeGraph is a general purpose engine to record the provenance of scientific processes and their outcome, the Neuroinformatics Platform hosts very specialised data types (e.g. large volumetric datasets) that deserve specialised data access to render them useful to the Neuroscience community. Thus, the Neuroinformatics Platform has developed several tailored services to address specific needs that were expressed by the platform users.

In the diagram below, we represented the components that belongs to each of these tiers:



Please note that the components above the red line are visible/accessible on the internet while those under are only accessible from inside the project internal network.

In the diagram below we will see the specific components involved in the two classes of services:

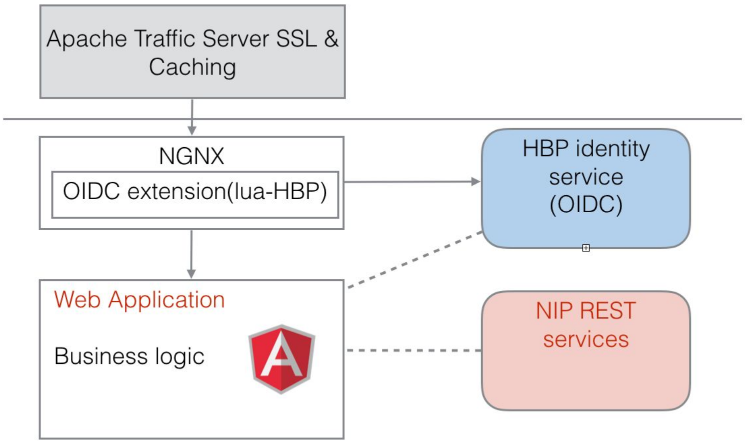


KnowledgeGraph **Specialised Data Access**

Architectural templates

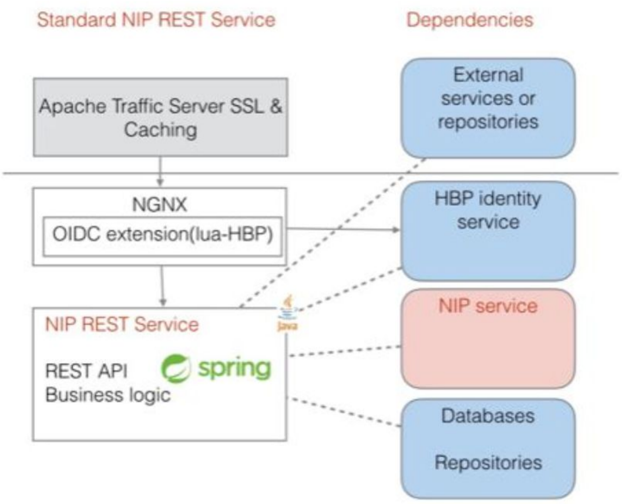
Web components

The diagram below depicts the standard architecture for all web applications of the Neuroinformatics platform:



Service components (REST APIs)

The diagram below depicts the standard architecture for the APIs of the Neuroinformatics platform:



Now we are going to describe all these components briefly, give an overview of their respective architecture and point to their respective documentation where it applies.

Web components

Search

This web application allows users to search and download datasets using a web browser. All data presented in this application is retrieved from the KSearch API.
Links: website | user manual

Registration

This web application allows users to import new datasets into the platform and share them with their peers. This application relies on the Orchestrator API to coordinate the data submission workflow.
Links: website | user manual

Documentation

This is where we document the Neuroinformatics Platform, the page you are currently reading is part of it.
Links: website

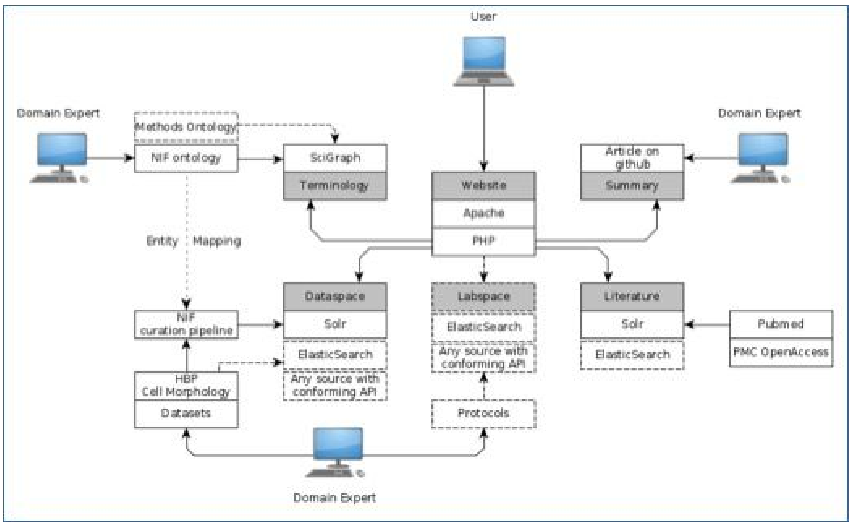
Ontology Viewer

This web application provides users with an integrated, user friendly view of the ontologies used in the Neuroinformatics Platform.
Links: website | user manual

Atlas Viewer
This web application is a specialised visualisation for brain volumetric datasets. All data presented in this application is retrieved from the Image Service API.
Links: website | user manual

KnowledgeSpace
KnowledgeSpace (KS) is a community encyclopaedia that links brain research concepts with data, models and literature from around the world. The aim of the KnowledgeSpace is to provide a unique interface between current scientific concepts and the evidence for their definition. It is an open project and welcomes participation and contributions from members of the global research community.

The KnowledgeSpace is developed in the context of an international collaboration to develop global standards for neuroscience vocabularies and federated search of diverse neuroscience repositories. This collaboration includes the International Neuroinformatics Coordinating Facility (INCF) and the Neuroscience Information Framework. The KnowledgeSpace development is coordinated by the INCF at the Karolinska Institute.



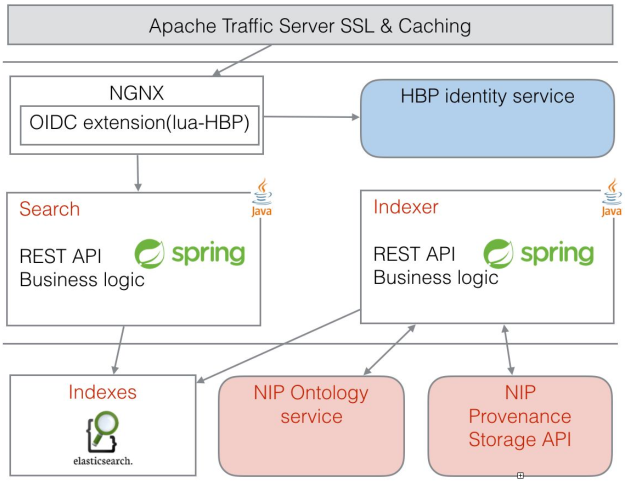
The website is developed in PHP. All ontologies are loaded in a Neo4j database using SciGraph. Domain experts contribute to the knowledge about concepts by contributing articles to a GitHub repository. Link to data tagged by NIF ontology concepts are curated and added to the Data space component for indexing. All indexes are built with the Apache Solr framework. The ontologies available in KnowledgeSpace are published in GitHub.

Links: website

Service components

KSearch API

This API allows users and computer programs to access the datasets integrated into the platform.



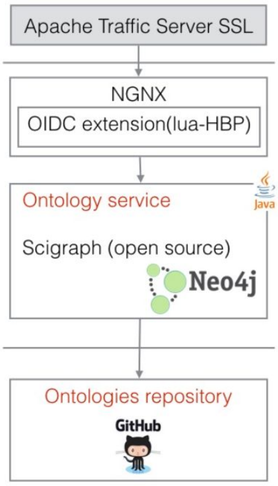
Links: user manual | endpoints documentation | demo

KIndexer API

This private API is intended at indexing HBP-PROV files into the KnowledgeGraph. Once indexed, the datasets become available in the Search API and thus, in the Search web application. The architecture representation is contained in the Search API section above.

Ontology API

This API gives access to the ontologies we use in the NeuroInformatics Platform.

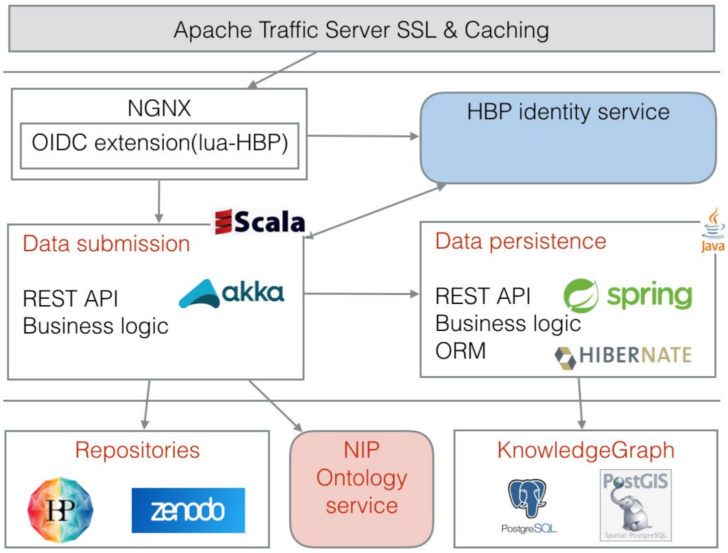


The ontologies available through the ontology service are dynamically loaded from a GitHub repository. When a new concept is needed for curating a dataset, the concept is added to the corresponding ontology in GitHub. Any update to the GitHub

repository or the configuration of which ontologies to load in SciGraph are automatically detected and triggers an update to the Neo4j database making the term available for data curation. Moreover additional sources of ontologies can be added if needed.
Links: user manual | endpoints documentation | demo [todo: create demo section]

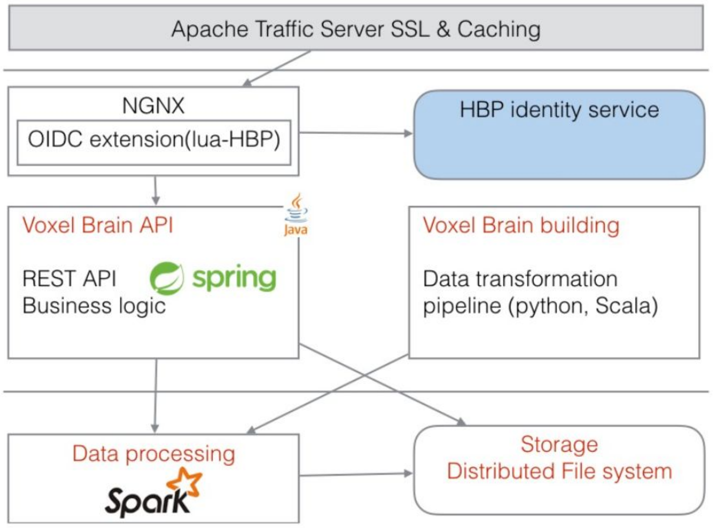
Orchestrator API

This API enables running of the Registration web application, it manages the submission workflow and coordinate a variety of NIP services and handle the upload of dataset to NIP storages. This API is not intended for public use, a dedicated data submission API will be made available toward late 2016.



Voxel Brain API

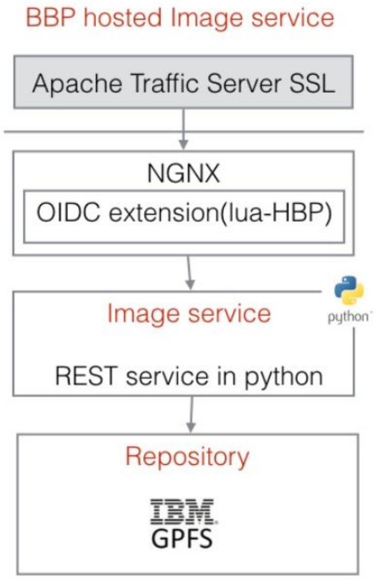
This specialised data access API serves voxel based volumetric datasets, and allows searching and filtering them.



Links: user manual | endpoints documentation | demo

Image Service API

This specialised data access API serves the purpose to access and serve very large volumetric datasets (BBIC format) in the form of image tiles.

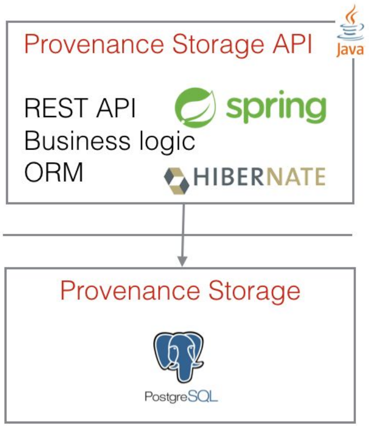


Links: user manual

Provenance Storage API

This private API has the responsibility to manage the long term storage of provenance data files. This include storage, retrieval and management of a registration lifecycle. Furthermore, this API exposes some reusable provenance activity’s attributes and their assigned unique identifiers:

- Agents (contributors, institutions, software)
- Protocols



HBP’s OpenID Connect (OIDC)

This service was not created by the Neuroinformatics Platform but is heavily relied upon to manage the identity to the user of the platform. Here are several use cases where the platform relies on OIDC:

a. When a web application requires to verify if a user is HBP accredited, to do so we verify via a token if the user is correctly logged in and if not, the web application redirects to the user to a login page (hosted by OIDC) which upon successful completion returns to the original application so the user can proceed with its original workflow.

b. When a REST API needs to verify the identity of the caller, a token is passed with the API call request and the service can contact OIDC to check if the token is valid, furthermore the service can request extra information about that token and get details about the identity as well as the groups that user belongs to.

Links: documentation

Storage components

ElasticSearch

This component is used to index verticals (e.g. Datasets) and provide a high performance means to query, and aggregate results.

The ElasticSearch component is scalable thanks to the Blue Brain Project Core Services team, which is able to launch new instances of the component at will on BBP private cloud.

Links: technology

Dataset Storages

This component is composed of various storage types:

a. Web Storage: storage used to download dataset via the Search web application or programmatically. This type of storage is used when users do a data submission through the Registration application, uploaded data gets uploaded in such storage automatically (e.g. HBP Document Service, Zenodo).
b. Long term storage: storage used for the purpose of archiving dataset (e.g. tape backup).
c. Analytics storage: high performance storage used for the purpose of processing dataset efficiently (eg. GPFS).

Zenodo

This is the primary *archival data repository* used by the Neuroinformatics Platform. It has been developed as part of the OpenAirePlus project and is operated by the data services of the CERN using the extensive data storage infrastructure of the CERN to provide long term storage, with replication and tape backup of large quantities of data. The storage in Zenodo is provided to HBP for free and the organisation has agreed that up to 1PB of data can be stored by the HBP with no cost. This is sufficient for depositing virtually all deliverables (including documents, presentations, etc.) in the HBP and accepting community contributions. In addition, Zenodo is integrated with the

European Commission reporting system and has a dedicated Human Brain Project community area.
Links: website | API documentation

Document Service
This storage area is developed and maintained by HBP for the purpose of providing storage to HBP Collabs. The Neuroinformatics Platform is using this storage when privacy requirement require us to keep integrated data within our institution or finer grained control than Zenodo can provide.
Links: API documentation

GPFS
The General Parallel File System (GPFS) is a high-performance clustered file system developed by IBM. It can be deployed in shared-disk or shared-nothing distributed parallel modes. It is used by many of the world's largest commercial companies, as well as some of the supercomputers on the Top 500 List. BBP maintains several GPFS clusters and the Neuroinformatics platform uses it to store and process large volumetric brain datasets.
Links: documentation

Provenance Storage
This component is the long term storage for the registrations made into the KnowledgeGraph. It is implemented using a relational database (PostgreSQL) and not only handles the raw provenance description (HBP-PROV file) but also the lifecycle of these registration. Furthermore, this database keeps track of some reusable provenance activity’s attributes and their assigned unique identifiers:

- Agents (contributors, institutions, software)
- Protocols

Links: technology

Cloud usage

The Neuroinformatics Platform was built to leverage modern cloud technologies. The platform comprises of many web and service components that can be deployed on arbitrary virtual machines and the front server component tie them together under a unified URL schema as described below:

- Service components: https://nip.humanbrainproject.eu/api
- Web components: https://nip.humanbrainproject.eu/

Federation

The Neuroinformatics Platform allows external contributors to register data that they host themselves (as opposed to be copied to and hosted in the NIP), as long as the data is accessible through a standardized interface. This is what we call **Active Data Repositories**.

For instance, a BBIC Image Stack dataset could be served from a third-party Image Service instance, as long as its REST API is identical to the NIP Image Service.

The KnowledgeGraph enables such federation scenarios by not being tied to the sole NIP storage components. It can refer to external services (Active Data Repositories) as ways to access a dataset. A given dataset may also be accessible through multiple services hosted in multiple places.

For instance, we collaborated with EGI on deploying instances of the Image Service on multiple datacenters across Europe. All public Atlas datasets were therefore accessible through any of the EGI datacenters, in addition to being accessible through the NIP Image Service instance.

Thanks to the KnowledgeGraph, we can discover which Active Data Repositories are available to access registered datasets. We are working on a new service component to make it easier to access data from multiple sources.

Software development and service deployment

The functionalities of the Neuroinformatics Platform are exposed via REST services. In addition, web client applications provide interactions with the content and functionalities of the Platform. The NIP REST services, web applications and databases are hosted on BBP infrastructure. All Platform modules are developed and deployed following BBP development best practices.

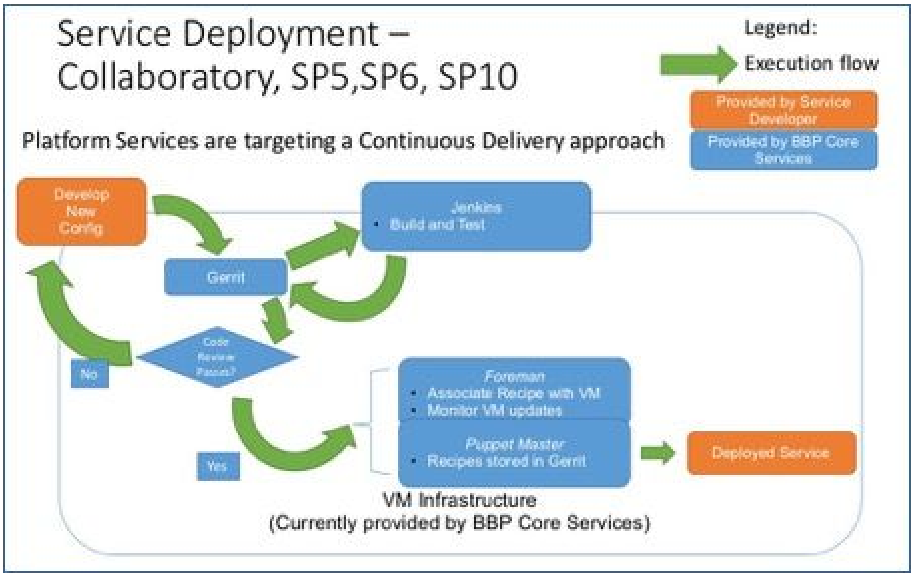
All services and UI components are developed using the following best practices:

- Unit testing and assessment of test coverages
- Utilisation of code review
- Automated Integration testing suites (automated post-deployment system testing)
- Continuous integration
- Ticket management in Jira for project feature planning
- Coding standards

The Platform is developed following Agile methodology (SCRUM), with new features and functionalities developed iteratively and released continuously. Thus repeatable deployment software is a crucial component to deliver continuous improved software to users.
Best practices from Agile software development have been adopted and implemented with a pragmatic DevOps model.

- Deployment lifecycle in development, staging and production environments for all services and web application,
- Continuous integration, i.e. unit testing, integration testing, automated software builds and package releases,
- VM configuration managed in source control and associated with a specific VM,
- Continuous automated deployment of the latest version of the software in dev,
- Automated release of the Platform in staging environment. The Platform is released in production after running integration and manual tests,
- Internet Gateway with caching proxy server.

The Neuroinformatics Platform follows the BBP Standard development and deployment model, as described in the HBP System Engineering documentation and represented below:



Service Deployment – Collaboratory, SP5,SP6, SP10

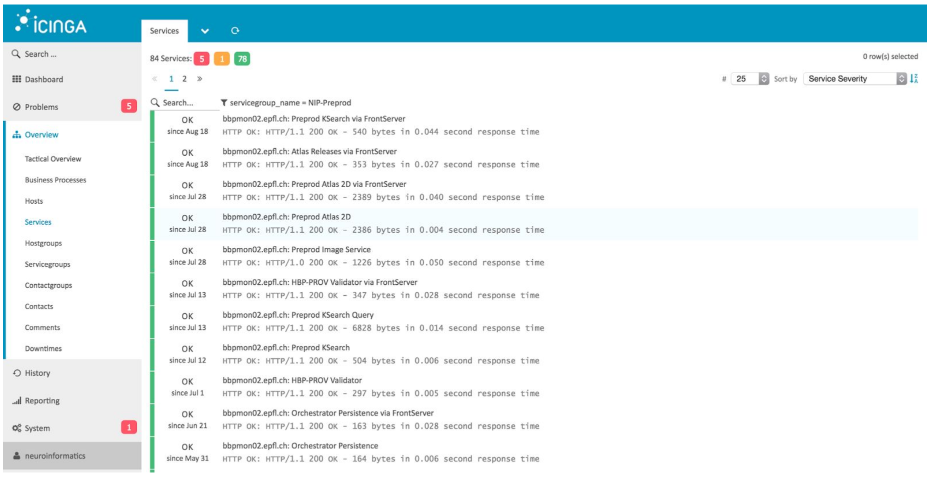
Platform Services are targeting a Continuous Delivery approach

This process makes extensive use of the following open source tools:

- **Git** – a leading distributed Version Control system [link]
- **Jenkins** – a leading Java-based Continuous Integration system [link]
- **Gerrit** – a leading Git and Java-based source code change review system [link]
- **Puppet** – a leading *devops* system configuration system [link]
- **Foreman** – associates Puppet recipes with particular hosts [link]

Platform monitoring and maintenance

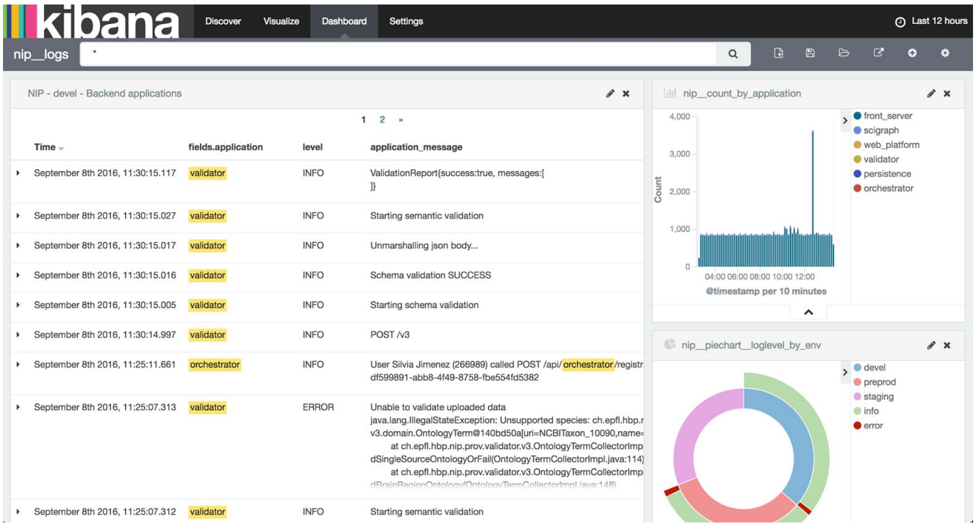
All NIP web applications and service are continuously monitored and alerts are raised should one of these health checks fail. All developers of the Neuroinformatics Platform team are notified via email of such failure and they take responsibility to correct applicative faults or escalate the issue to the BBP core services if the need arise. A dashboard is visible where the developers work as an extra mean of raising awareness of failing service and diminish the time it takes to restore a failed service. The screen capture below shows this dashboard:



servicegroup_name = NIP-Preprod

```
OK             bbpmon02.epfl.ch: Preprod KSearch via FrontServer
since Aug 18   HTTP OK: HTTP/1.1 200 OK - 540 bytes in 0.044 second response time
OK             bbpmon02.epfl.ch: Atlas Releases via FrontServer
since Aug 18   HTTP OK: HTTP/1.1 200 OK - 353 bytes in 0.027 second response time
OK             bbpmon02.epfl.ch: Preprod Atlas 2D via FrontServer
since Jul 28   HTTP OK: HTTP/1.1 200 OK - 2389 bytes in 0.040 second response time
OK             bbpmon02.epfl.ch: Preprod Atlas 2D
since Jul 28   HTTP OK: HTTP/1.1 200 OK - 2386 bytes in 0.004 second response time
OK             bbpmon02.epfl.ch: Preprod Image Service
since Jul 28   HTTP OK: HTTP/1.0 200 OK - 1226 bytes in 0.050 second response time
OK             bbpmon02.epfl.ch: HBP-PROV Validator via FrontServer
since Jul 13   HTTP OK: HTTP/1.1 200 OK - 347 bytes in 0.028 second response time
OK             bbpmon02.epfl.ch: Preprod KSearch Query
since Jul 13   HTTP OK: HTTP/1.1 200 OK - 6828 bytes in 0.014 second response time
OK             bbpmon02.epfl.ch: Preprod KSearch
since Jul 12   HTTP OK: HTTP/1.1 200 OK - 504 bytes in 0.006 second response time
OK             bbpmon02.epfl.ch: HBP-PROV Validator
since Jul 1    HTTP OK: HTTP/1.1 200 OK - 297 bytes in 0.005 second response time
OK             bbpmon02.epfl.ch: Orchestrator Persistence via FrontServer
since Jun 21   HTTP OK: HTTP/1.1 200 OK - 163 bytes in 0.028 second response time
OK             bbpmon02.epfl.ch: Orchestrator Persistence
since May 31   HTTP OK: HTTP/1.1 200 OK - 164 bytes in 0.006 second response time
```

Furthermore, all API services have their application logs collected and centralised using FileBeat (log shipping), LogStash (log processing) and Kibana (log inspection and dashboard facilities). The figure below shows a basic kibana dashboard that highlights that some application have reported ERROR logs in both dev and staging environment, a developer is then able by clicking on that red section of the pie chart (bottom right panel) to see the specific logs (left side panel), even if they potentially came from different virtual machines.



Time	fields.application	level	application_message
September 8th 2016, 11:30:15.117	validator	INFO	ValidationReport{success:true, messages:[ ]}
September 8th 2016, 11:30:15.027	validator	INFO	Starting semantic validation
September 8th 2016, 11:30:15.017	validator	INFO	Unmarshalling json body...
September 8th 2016, 11:30:15.016	validator	INFO	Schema validation SUCCESS
September 8th 2016, 11:30:15.005	validator	INFO	Starting schema validation
September 8th 2016, 11:30:14.997	validator	INFO	POST /v3
September 8th 2016, 11:25:11.661	orchestrator	INFO	User Silvia Jimenez (266989) called POST /api/orchestrator/registr df599891-abb8-4f49-8758-fbe554fd5382
September 8th 2016, 11:25:07.313	validator	ERROR	Unable to validate uploaded data java.lang.IllegalStateException: Unsupported species: ch.epfl.hbp.r v3.domain.OntologyTerm@140bd50a[uri=NCBITaxon_10090,name= at ch.epfl.hbp.nip.prov.validator.v3.OntologyTermCollectorImp dSingleSourceOntologyOrFail(OntologyTermCollectorImpl.java:114) at ch.epfl.hbp.nip.prov.validator.v3.OntologyTermCollectorImp dBrainRegionOntology(OntologyTermCollectorImpl.java:148)
September 8th 2016, 11:25:07.312	validator	INFO	Starting semantic validation

Data Formats

HBP-PROV Format

Purpose

HBP-PROV is the recommended format for data submission in the Neuroinformatics Platform. It is based on the W3C standard PROV-O and allows to capture the information about dataset provenance . The following information are described within a HBP-PROV instance:

- when was the data created,
- which samples have been used,
- what are the characteristics of the animal these samples were derived from,
- how was the sampling organized, what protocols have been used,
- what brain regions the data is about,
- which coordinates on an atlas does it have,
- which datatypes the files in the dataset have,
- which files have the same content and differ only in format,
- what analysis and transformations were applied to the data? What software has been used for that,
- what are the information about agents (persons, organizations, software) involved in the generation of the dataset,

Design

HBP-PROV uses the main concepts of W3C PROV: Agent, Activity and Entity. The entities are generated by activities and can be used as input by other activities. The activities are attributed to the agents and the when the new Entity is generated it's associated with the corresponding agents. PROV-PRIMER has detailed yet informal introduction to PROV standard.

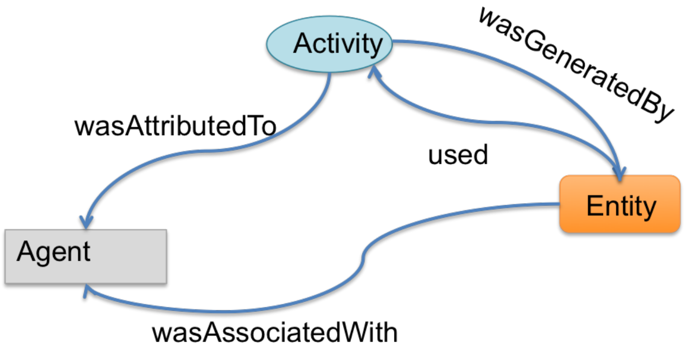


Fig. Core concepts of W3C PROV

HBP-PROV introduces the following entities (informal definitions):

- Resource: a file available by its URI
- Dataset: a set of files annotated with neuroscientific metadata
- Registration: registration activity, regrouping all activities described in
- the file.
- Specimen - an individual animal, plant, or single-celled life form.
- Sample - usually a piece of tissue taken from the specimen
- Model - a special kind of dataset that represents the computer model of some biological entity of phenomenon.

HBP-PROV introduces the following Agents:

- Contributor
- Organisation
- Software

HBP-PROV introduces many types of activities, as defined by Activity Ontology.

HBP-PROV elements identification

Entities, activities and agents are identified by their ids in the form of UUID.

Entities and their metadata

Registration

registration activity, regrouping all activities described in the file

Attribute name	***Type***	***Example***
id*	*UUID*	
name	*Free text*	
comment	*Free text*	
submission_date*	*date*	
curation_date	*date*	
release_date*	*date*	
agents*	*Array(Agent)*	
release*	*enum*	*public, hbp_only, private*

Specimen

An individual animal, plant, or single-celled life form.

Attribute name	***Type***	***Example***
id*	*UUID*	
name	*Free text*	
comment	*Free text*	
age	*Structured (period + unit + age_range*	*Post-natal, 14*
developmental stage	*ontology*	
sex	*ontology*	
species*	*ontology (taxonomy)*	*rat*
strain	*ontology (taxonomy)*	*Han Wistar*
transgenic	*ontology*	
disease	*ontology*	*alzheimer disease*

Activity

An activity represents the process of one or several agents implementing a protocol, consuming entities and generating other entities.

Attribute name	***Type***	***Example***
id*	*UUID*	
name	*Free text*	
comment	*Free text*	
type*	*Ontology*	*Post-natal, 14*
start_date	*date*	
end_date	*date*	
agents*	*Agent + Ontology (role)*	*contributor with role "researcher", principal investigator"*
protocols	*Protocol*	

methods	[Ontology](#) (Methods)	*Biocytin staining, multi electrode recording*
sources entities*	*UUID reference (specimen, sample, dataset, resource, model)*	

Protocol

A protocol captures a precise description of the way an activity was performed.

Attribute name	***Type***	***Example***
id*	*UUID*	
title*	*Free text*	
description*	*Free text*	
designer	*Free text*	*BBP*
Publication	*Structured (type, id)*	*Pubmed, DOI*

Agents

Three types of agents are supported:

Contributor

People who contributed to an activity.

Attribute name	***Type***	***Example***
id*	*UUID*	
family_name*	*Free text*	
given-name	*Free text*	
affiliation	*Structured (organisation ref, lab)*	
email	*date*	

Organisation

Organisations that contributed to an activity.

Attribute name	***Type***	***Example***
id*	*UUID*	
name	*Free text*	

Software
Software used in an activity to generate a dataset.

Attribute name	***Type***	***Example***
id*	*UUID*	
name	*Free text*	
version	*Free text*	

Sample

Representation of a biological material used in the process of generating a dataset.

Attribute name	***Type***	***Example***
id*	*UUID*	
name*	*Free text*	
comment	*Free text*	
activity_ref*	*UUID reference (Activity)*	
brain_region	*Ontologies: rat, mouse, marmoset, opossum, human, macaca mulata*	

Dataset

List of generated datasets to be registered.

Attribute name	***Type***	***Example***
id*	*UUID*	
name*	*Free text*	
description	*Free text*	
categories*	*Array (Ontology (Category))*	*Electrophysiology*
activity_ref*	*UUID reference (Activity)*	
brain_region	*Ontologies: : rat, mouse, marmoset, opossum, human, macaca mulata*	
atlas_location	*Atlas Location*	*bounding box in Waxholm space atlas*
licence	*Ontology*	
representations*	*Array (Resource)*	
documentation_file	*Structured*	
publication	*Structured*	*PMID or DOI*
attributes	*Key (ontology) + Value (free text)*	*(stimuli, IDrest) (receptor, "GABA receptor" from Cell receptor ontology*

Resource

List of files used in the registration.

Attribute name	***Type***	***Example***
id*	*UUID*	
addresses*	*Array(Address)*	
mime_type	*Ontology*	
checksum	*Free text*	*MD5*
size	*Integer (in Bytes)*	
original_filename	*Free text*	
retrieval_date	*Date*	
comment	*Free text*	
activity_ref	*UUID reference (activity)*	
attributes	*Key (ontology) + Value (Free text)*	

Address

The location of a given resource in the form of a storage type and a URI.

Attribute name	***Type***	***Example***
id	*UUID*	
storage	*Ontology*	*GPFS*
uri	*Free text*	*/a/b/cell,nwb*

Atlas Location

Details for registering a dataset in a reference atlas.
Note that you can only put one of either atlas_template or parent_space.

Attribute name	***Type***	***Example***
Atlas_template	Ontology (HBP atlas template ontology)	*Waxholm Space rat brain atlas v.2.0*
parent_space	*UUID reference (dataset)*	*Allen CCFv3 Atlas*
bounding_box*	*2x (x,y,z) coordinate*	
transformation	*Structured(Type, Matrix)* *Type = Enum(linear, polynomial)*	

Classification

A list of classification terms derived from a manual or automated analysis.

Attribute name	***Type***	***Example***
classified_entity_ref*	*UUID reference (sample)*	*Sample (neuron)*
assigned_class*	*Ontology (cell type)*	*- Pyramidal cell - cSTUT (classical stuttering cell)*
activity_ref	*UUID reference (activity)*	
evidences	*UUID reference (dataset)*	*- Dataset (morphology) - (Dataset (ephys)*

Model

In silico models, used to perform simulations and in silico experimentation.

Attribute name	***Type***	***Example***
id*	*UUID*	
name*	*Free text*	
comment	*Free text*	
categories*	*Array(Ontology)*	
activity_ref*	*UUID reference (Activity)*	
brain_region	*Ontologies: : rat, mouse, marmoset, opossum, human, macaca mulata*	
representations*	*Array(Resource)*	

JSON Schema

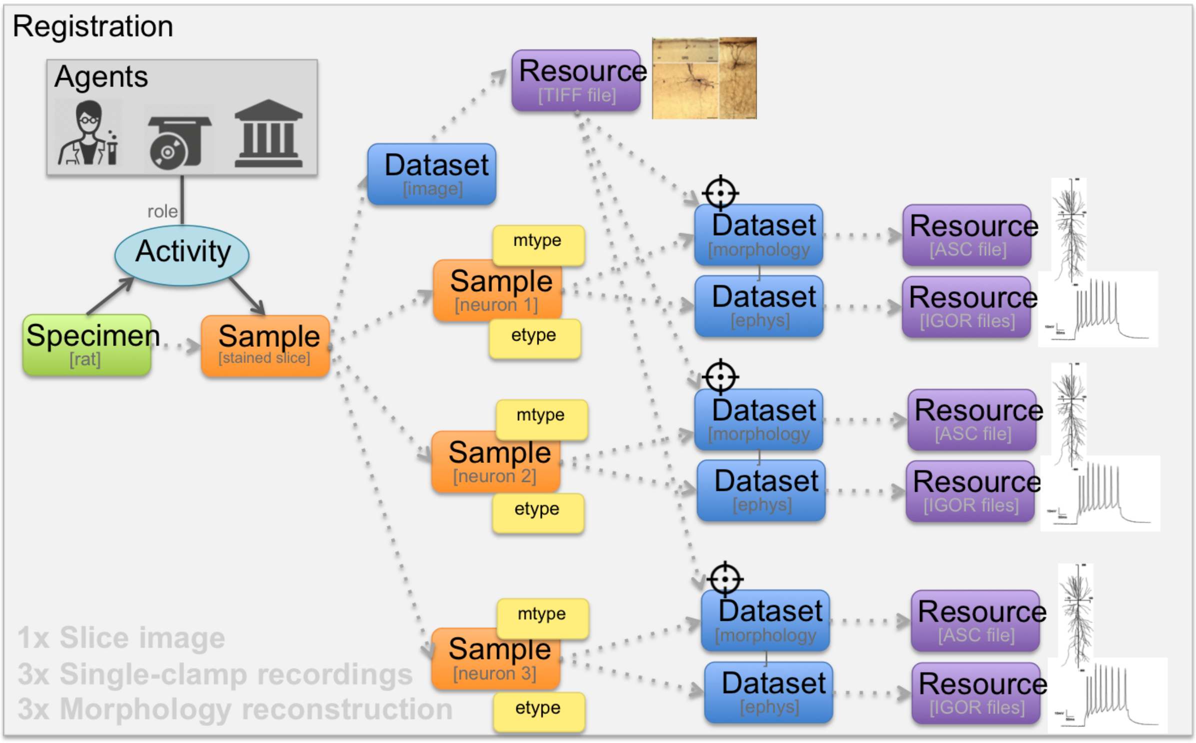
A json schema defines the correct syntax of an HBP-PROV instance https://schema.humanbrainproject.eu/neuroinformatics/hbp_prov/hbp-prov-schema-v3.0.json For each metadata the schema holds a description section which defines it. The schema is be the main source of reference, it’s used in the validators of the Knowledge Graph.

Validator

A validator service allows users to verify if an HBP-PROV instance is correct, more information can be found in the corresponding user manual section.

Usage example

Here's is an example of how the provenance for the results of work of LNMC laboratory can be represented using HBP-PROV. Note that not all the Activities are shown - the dotted lines represent them for simplicity. For example the Sample neuron1 isDerivedFrom the stained slice, using the Derivation Activity.



Known shortcomings

The list of known issues are planned to be addressed in the future and released to the community:

- **An HBP-PROV instance doesn't store a link to its corresponding schema.** Just like a json schema point to a schema (e.g. $schema = ''), the json instances do not have this facility. We need to roll up our own. Other project are known to have done this (http://snowplowanalytics.com/blog/2014/05/15/introducing-self-describing-jsons/).
- **Atlas location not available on Sample**. In the case of a Neuron for which we know it's spatial coordinate and given 2 datasets (electrophysiology, morphology), it would be desirable to store the atlas location of the Sample representing the Neuron rather than repeating twice on the datasets.
- **Relationship between Dataset and Resource**. all relationships in the PROV model are directed toward the object used in the corresponding activity. However, a Dataset holds a list of representations
- **An HBP-PROV file represent a single registration**. when we start tacking derivation of datasets across multiple registrations, it will become necessary to have

the ability to merge several HBP-PROV files into one, and thus store multiple registrations. That way a tools tasked to present the exhaustive provenance could gather all data in a single payload. A dedicated NIP service could be responsible to merge the transitive provenance of a given dataset into a single file.

- **Attributes are needed on Specimen, Sample, Model, Activity**. Currently they are only available on Dataset and Resource.
- **Concept of Dataset Revision needs to be more deeply integrated**.
- In the case of an **Activity that consumes multiple entities and generate multiple entities** (eg. multi-clamp recording to generate a connectivity dataset), it is not possible to infer which ephys trace correspond to which neuron. So far we have worked around this problem by having a single activity per neuron recording and having a merging activity that takes all traces to infer a connectivity dataset.

XLS format

Purpose

Data can be manually curated prior to its registration in NIP platform. To support such activity, an excel format have been created and is seen as a table and less verbose version of HBP-PROV format which remains the reference.
An excel file for manual curation is made of blocks (set of excel rows) corresponding to the different HBP-PROV elements.

Disclaimer

The excel format described below is not yet finalized and can be subject of non backward compatible changes.

General considerations about blocks

An excel registration file is supposed to be self-contained to be correctly converted in the HBP-PROV format. This means:

- All blocks (activities, entities, agents,cross_reference,...) referenced in an other block (through its identifier) should be defined in the current excel file.

Activity blocks

Activity agents are indicated by the mean of two properties: agent_id and agent_role. Those two properties can be repeated as much as necessary but with the following constraints:

- The number of agent_id should be equal to the number of agent_role
- The role of an agent is the value of the agent_role following it's agent_id

Registration block:
See registration object in the HBP-PROV format.

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
activity	*registration*	***agent_id****	*string*		
		agent_role*	*string*	*Ontology term's curie*	*Property value example: principal investigator* *ontology: HBP_ROLE:0000040*
		release*	*public,hbp*		*Property value example: public*
		submission_date*	*date*		*Property value example: 01/08/2015*
		curation_date	*date*		*Property value example:*

					16/03/2016

Experiment block:
See Activity object in the HBP-PROV format.

Node	Type	Property_name	Property value	Ontology	Comments
activity	experiment	agent_id*	string		
		agent_role*	string	Ontology term’s curie	Property value example: principal investigator ontology: HBP_ROLE:0000040
		source_entity_id*	string		The id of an entity (specimen, sample, datasets,...)used as an input of the experiment.
		output_entity_id*	string		The id of an entity (sample, datasets,...) generated by the experiment.
		protocol_title	string		Property value example: Construction of version 3 (v3) of the Allen Mouse Common Coordinate Framework (CCF)
		protocol	string		Protocol text.
		date	date		Property value example: 16/03/2016
		end_date	date		Property value example: 16/04/2016

Computation block:
Same as experiment block description.

Entity blocks

Specimen block:

See Specimen object in the HBP-PROV format.

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
entity	*specimen*	***id****	*string*		
		species*	*string*	*Ontology term's curie*	*Example:* *Property value: Mus musculus* *ontology: obo:NCBITaxon_10090*
		strain	*string*	*Ontology term's curie*	*Example:* *Property value:C57BL/6J* *ontology: efo:EFO_0000606*
		sex	*string*	*Ontology term's curie*	*Example:* *Property value: male* *ontology: HBP_SEX:0000001*
		age_value_start	*number*		*Example: 56*
		age_value_end	*number*		
		age_unit	*days, weeks, months, years*		*Example:* *days*
		brain_region	*string*	*Ontology term's curie*	*Example:* *Property value: brain* *ontology: ABA:997*

Dataset block:
See Dataset object in the HBP-PROV format.

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
entity	*dataset*	***id****	*string*		
		name*	*string*		*Example:* *Property value: P40 mouse hippocampus endothelial cell single-cell transcriptomics*
		description	*string*		
		data_modality*	*string*	*Ontology term's curie*	*Example:* *Property value: Single cell transcriptomics* *ontology: HBP_DAMO:0000023*
		publication_id	*string*		*The identifier of a publication describing the dataset*
		attributes_id	*string*		*The identifier of an attribute block. This property can be repeated if multiple attributes are needed.*
		atlas_template	*string*	*Ontology term's curie*	*This property (and the following ones) is filled if the dataset is an atlas.* *Example:* *Property value: Allen reference atlas v3* *ontology: HBP_BATT:0000002*
		available_directions	*coronal, axial, sagittal*		*If the three directions are needed, the property will be repeated.*

		anterior_posterior_resolution	number		Example: Property value: 10
		superior_inferior_resolution	number		Example: Property value: 10
		left_right_resolution	number		Example: Property value: 10
		resolution_units	string		Example: Property value: microns per pixel

Resource block:
See Resource object in the HBP-PROV format.

Node	Type	Property_name	Property value	Ontology	Comments
entity	dataset	**id***	string		
		dataset_id	string		The identifier of a dataset block
		checksum	string		
		original_filename	string		
		size	string		
		retrieval_date	date		Correspond to the date when the curator get the files Example: Property value: 15/10/2015
		data_type	string	Ontology term’s curie	Example: Property value: image/vnd.nrrd;version=0 ontology: HBP_DATT:0000071
		url*	string		Example: Property value: http://download.alleninstit

					ute.org/informatics-archive/current-release/mouse_ccf/average_template/average_template_10.nrrd Where the files are retrieved from. The url property can be repeated if many files are involved. In such case, each url should have a storage_type companion
		storage_type	string		Example: Property value: Source ontology: HBP_STO:0000005

Agent blocks
See Agent object in the HBP-PROV format.

Contributor block:

Node	Type	Property_name	Property value	Ontology	Comments
agent	contributor	**id***	string		
		family_name*	string		
		given_name	string		
		email	string		
		lab_name	string		
		organization	string		

Organization block:

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
agent	*organization*	***id****	*string*		
		organization	*string*		*The name of the organization* *Example:* *Property value: Allen Institute for Brain Science, Washington, USA*

Software block:

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
agent	*software*	***id****	*string*		
		name	*string*		*The software name.*
		version	*string*		*The software version*
		url	*string*		*The software download page*

Cross_reference blocks

Publication block:

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
cross_reference	*publication*	***id****	*string*		
		One of : [doi, pubmed_id, zenodo, web]*	*string*		*Example:* *Property value: 10.5281/zenodo.48065*

		storage_type*	*string*	*Ontology term's curie*	*Currently the supported storage types are: doi (HBP_STO:0000012), pubmed_id (HBP_STO:0000011), zenodo (HBP_STO:0000007) and web (HBP_STO:0000002)* *Example:* *Property value: HBP_STO:0000012*

Annotation blocks
Attributes block:

Node	***Type***	***Property_name***	***Property value***	***Ontology***	***Comments***
annotation	*attributes*	***id****	*string*		
		Ontology term's curie *	*string*	*Ontology term's curie*	*Example:* Property_name: HBP_DTAT:0000021 (Receptor type) *Property value: AMPA receptor* *ontology: MESH:D018091*

Atlasing - BBIC

Image sections

BBIC stands for Blue Brain Imaging Container. The BBIC itself is HDF5 format. The motivation to create this in-house standard format, to contain image tiles in each orientation at different zoom level, was to allow Multiresolution Atlas Viewer (client) uniformly access image tiles. Because visualizing and querying different brain atlases with various granularities from different species in one browser application through integrating them is a challenging task.

BBIC structures are described below.
Dataset: bbic/stacks/n/levels/k/x/y/slice
Stacks by orientation:

0 refers to sagittal
1 refers to coronal
2 refers to axial

Levels (zoom levels or level of detail):
Depending on original resolution or dimension of images, zoom levels vary.

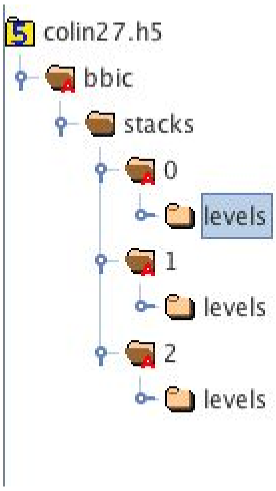


Figure 1. General structure of BBIC

X (row number of where the tile is located)
Y (column number of where the tile is located)

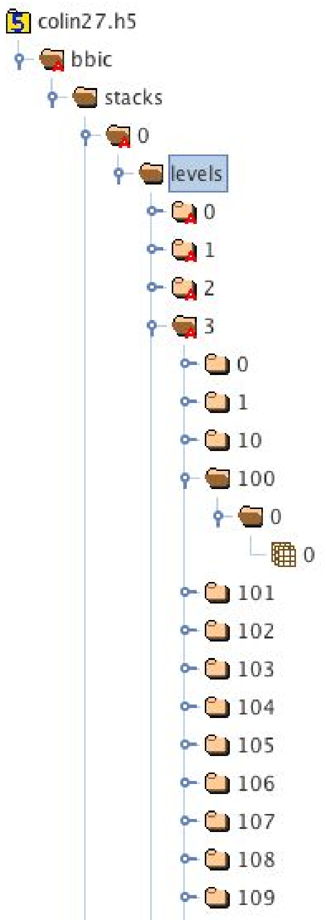


Figure 2. Slice, X, Y structure of BBIC

In the above example, we see a tile dataset with zoom level at 3, slice number 100, x 0, y 0. Logical dimensions of this dataset is 256 by 256.

Annotations

Currently, we have brain regions as annotations of the atlas images. These brains regions were originally created either from expert drawing or segmentation result of images. Either way, we store brain regions in SVG format. Each orientational section (slice) has brain region drawings in SVG. All the SVG files are written into HDF5 container in BBIC format. Structure of the BBIC annotations is following:

bbic/stacks/n/slice

Ongoing International data standardisation efforts

A number of key standardisation efforts are underway. These fall into several distinct categories with neuroscience data repositories, metadata standards and ontologies, volumetric data standards and APIs, vocabulary services.
In addition, the specific experiences curating and integrating data for HBP and community use cases will result in additional standards and recommendations.

KnowledgeSpace for Standardised Neuroscience Vocabularies

The KnowledgeSpace provides an open community space to standardise on neuroscience vocabularies and ontologies. By providing a standard vocabulary service, Github repository and public forum with hypothes.is integration for community comment and annotation, the KnowledgeSpace provides a community forum for standardising on neuroscience vocabulary terms. This work builds off of years of work curating and integrating neuroscience ontologies from NIF, EBI, GO, INCF and others. A series of Knowledge Hackathons are being planned in collaboration with INCF to bring data scientists, neuroscientists and ontologists together to curate key branches of the neuroscience ontologies used by the KnowledgeSpace.

MINDS – Minimal Information for Neuroscience DataSets

A major challenge in neuroscience is that of discovering diverse neuroscience datasets and identifying those that are similar or can be combined. We are working to establish a "Minimal Information for Neuroscience Data Sets" (MINDS). MINDS should contain metadata describing the specimen or subject details (including species, age, strain etc); contributor (principal investigator, contributing scientist, technicians, etc); brain location (using a standard atlas or ontology); methods and/or protocols (what method types, analysis methods, equipment, parameters, and specific protocols were used to produce the data); data category (such as EEG, intracellular recordings, MRI, fMRI, etc); data format (the binary file format used, such as WaveCloud, Igor, EDF, NIFTI, DICOM, etc); and a persistent identifier(s) for where the data can be accessed. MINDS metadata will use INCF and NIF standardised neuroscience vocabularies populating the KnowledgeSpace. With respect to data repositories, virtually all data repositories support exposure of metadata for archived data objects through the Open Archive Initiative - Protocol for Metadata Harvesting (OAI-PMH). Thus, the objective is to deliver an open specification (based on HBP-PROV) for exposing metadata according to OAI-PMH standards. This specification will be employed in international collaborations with existing European and Canadian infrastructures and it will be proposed to US, Chinese and Japanese initiatives looking to accomplish similar objectives. The proposed standard will be widely disseminated and promoted as the way to ensure data is discovered from any OAI-PMH compatible data repositories (virtually all are including CKAN, Invenio (Zenodo), etc). Adhering to this standard will also ensure automatic discoverability of data in the Dataspace and KnowledgeSpace.

Volumetric Imaging Data Formats and Services

Building off of the outcome of an RDA Workshop on Infrastructure for Understanding the Human Brain (organized by SP5), activities are underway to establish common standards for large-scale volumetric image data and APIs for accessing and analysing these data from distributed data repositories. Current technologies under review include DVID (from Janelia

Research, neurodata.io, and array databases. These are all being evaluated for neuroscience atlas use case usability and performance requirements. The work reviewing array databases for neuroscience use cases is being performed within the context of RDA activities led by Peter Wittenberg. Further evaluations are taking place of volumetric services and APIs used by such annotation software as CATMAID ,Vaa3D, ESPINA, etc. The SP5 BBIC, DVID and neurodata.io (formerly OpenConnectome) services are being evaluated. A community-based workshop is being planned to follow-up and discuss additional community use cases and requirements.

Neuron morphology data formats

Currently, a few formats currently dominate the description of neuron reconstructions. A data format from the authors of Neurolucida (a commercial reconstruction package) is widely used, as is another format SWC used by competing software. In conjunction with the INCF Neuroinformatics meeting in September, 2016 - a workshop is planned to evaluate the requirements for a common standard neuron reconstruction data format. This includes looking at standardising metadata and keeping track of different versions and the provenance of neuron morphologies. The workshop will include members of the BigNeuron initiative, neuromorpho.org and other community members organised by INCF and SP5.

Neuron Electrophysiology data formats

The Neurodata without Borders initiative was initiated by the Allen Institute and Kavli Foundation to establish a standard data format for neurophysiological data including optical, electrical, behavioural and their associated events. This format has been deployed with the release of the Allen Institute cell types data base (celltypes.brain-map.org). In addition, several leading labs have converted their datasets into NWB for public release (including Marcus Meister, Gyorgy Buzsaki, Ken Harris, Karel Svoboda, etc).

Glossary

Voxel Brain annotation layer

In the context of the Voxel Brain, an annotation layer is the dataset we attach to the voxels of a data volume.

Here are a few examples:

a. A brain region annotation layer would store an ontology term describing a given brain region or brain parcel per voxel in the atlas space.

b. a cell density Annotation layer would store a float value to every voxel of a given atlas space.

c. A gene expression annotation layer would store in each voxel of the atlas space a list of gene names with associated numerical expression level.

Atlas space

Atlas space refers to three-dimensional coordinate system of a given brain atlas.

Brain atlas

A brain atlas is a spatial representation of the brain that comprises ontology, parcellation, coordinate system, and multiple features/layers of the cerebral characteristics.

Brain parcel

A brain parcel is a specific brain region that is defined deterministically or probabilistically in the context of a given brain atlas.

Contributor

Individual or institution that produced the Data Set. The data will be accessible to the whole community. Therefore, experimental data are not permitted. Moreover, no personal information related to patients or de-anonymized data are accepted.

Data Registration

A unique set of information related to one specific dataset. It includes all piece of information required by the API or service registration pages as well as all piece of information about the Contributor registration, the contributors of the dataset and data user conditions.

Dataset

Digital data, either raw or derived, and its metadata provided through the data access portal interface, or through other media, including data and metadata derived from HBP monitoring protocols, field observations, collections, laboratory analysis, camera trap images, all written, recorded, graphic, audio, visual, and other materials in any media, whether or not subject to copyright protection, or the post-processing of existing data and identified by a unique identifier issued by the HBP.

All datasets provided by contributors should have been produced following EC ethical regulation.

Dataset Contact

Party designated in the accompanying metadata of the dataset as the primary contact for the dataset.

Data User

Individual to whom access to this dataset may be granted, subject to acceptance of these Terms and Conditions by such individual and his or her immediate collaboration sphere, defined here as the institutions, partners, students and staff with whom such individual collaborates and to whom access must be granted in order to fulfill the such individual’s intended use of the dataset.

Human Brain Project

Human Brain Project (HBP) is a European Commission Future and Emerging Technologies Flagship that aims to achieve a multi-level, integrated understanding of brain structure and function through the development and use of information and communication technologies (ICT).

HBP-PROV

HBP-PROV is an exchange data format intended to represent how data was created, which organism was used, how it was processed - including which contributor participated, which software was used - and what dataset/files resulted from it.

KnowledgeGraph

KnowledgeGraph is a metadatabase built on a provenance data model HBP-PROV. In other words, it is a provenance graph to which data are registered for discovery, search, access and tracking. Currently, the KnowledgeGraph consists of Search API which is public and Indexer API which is private.

KnowledgeSpace

KnowledgeSpace (KS) is a community encyclopaedia that links brain research concepts with data, models and literature from around the world. It is an open project and welcomes participation and contributions from members of the global research community.

Ksearch

Ksearch is the search component of the Neuroinformatics Platform. It is a REST API allowing searching curated datasets using different filters that are mainly taken from MINDS.

MeSH

MeSH is the National Library of Medicine's controlled vocabulary thesaurus used for indexing articles for PubMed. It consists of sets of terms naming descriptors in a hierarchical structure that permits searching at various levels of specificity.

MINDS

MINDS stands for Minimum Information for Neuroscience DataSets. Data shared on the Neuroinformatic Platform are enriched with minimal metadata to provide essential information in order to ensure traceability of any data artefact.

Neuroinformatics Platform

Neuroinformatics Platform (NIP) is Sub-Project 5 of the Human Brain Project. The Neuroinformatics Platform and the Brain Atlases allow neuroscientists to collaboratively curate, analyse, share, and publish heterogeneous neuroscience data.

Parcellation

Brain parcellations define spatial or volumetric boundaries of brain region/structure. They could be represented in 2D and 3D depending on the use case.

Registration

For 2D or 3D volumetric datasets, we use different registration methods to convert one dataset from its own space to another space. Following registration methods are used:
Linear registration registration is used to capture the global transformation between the subject image and the atlas image.

Landmark registration module is used in the 3D Slicer to pick corresponding landmark in both subject image and atlas image.
Deformable registration is a process consisting of establishing functional or spatial correspondences between two images.

NOTE: There are “Data registration” and “Registration App”, which are separate terms.

Voxel

The etymology of the work comes from a *blend* of ‘*volumetric’* and ‘*pixel’.* A voxel is the three-dimensional analogue of a pixel in a two-dimensional space.

Voxel Brain

Voxel brain is a REST service provides access to volumetric brain atlas and their annotation layers in the form of voxels.