

Web-based visualization and analysis of neurophysiology data

Jeremy Magland, Center for Computational Mathematics, Flatiron Institute

Open Data in Neurophysiology (ODIN) Symposium 2023,
McGovern Institute for Brain Research, MIT



Why web-based software?

- **Easy to use:** no installation
- **Easy to share:** copy-paste the link
- **Cross-platform:** all desktop options and mobile
- **Development cycle advantages:** simplifies distribution, etc.
- **Integrates naturally with cloud resources:** e.g., DANDI
- **Limitations:** no native access to local files/software, requires internet connection, limited access to previous versions, requires coding in JavaScript



I will introduce three web-based tools for neurophysiology visualization and analysis

- [Figurl](#): framework for creating shareable interactive visualizations
- [Neurosift](#): static web app for visualizing NWB files hosted in the cloud (DANDI)
- [Protocaa](#)s: **prototype** web app for analyzing neurophysiology data in the cloud (or with cluster/local compute)

Figurl: overview

- Simplifies sharing of interactive figures
 - Run a Python script to generate a shareable URL
- Create custom visualization plugins
 - Static HTML bundles in the cloud
 - React/typescript
- Promotes scientific collaboration, communication, reproducibility
- Focus on neurophysiology, but can be used for other domains



Figurl: Plotly example

```

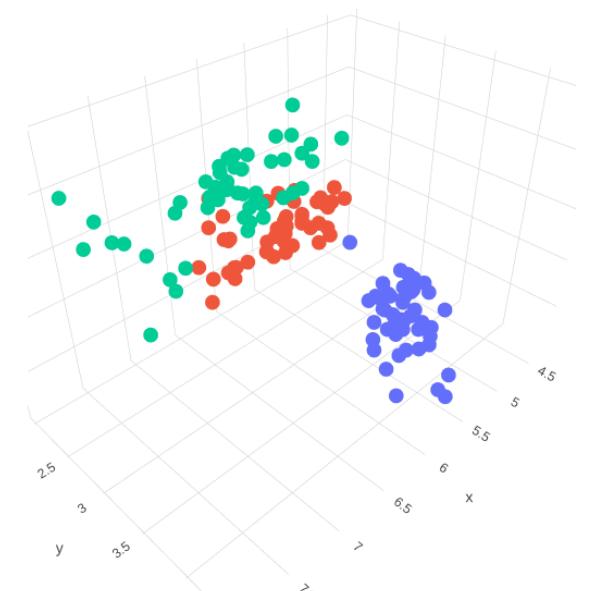
import plotly.express as px
import figurl as fig

# Load the iris dataset and create a Plotly figure
iris = px.data.iris()
ff = px.scatter_3d(iris, x='sepal_length',
                   y='sepal_width', z='petal_width',
                   color='species')

# Create and print the figURL
url = fig.Plotly(ff).url(label='plotly example - iris 3d')
print(url)

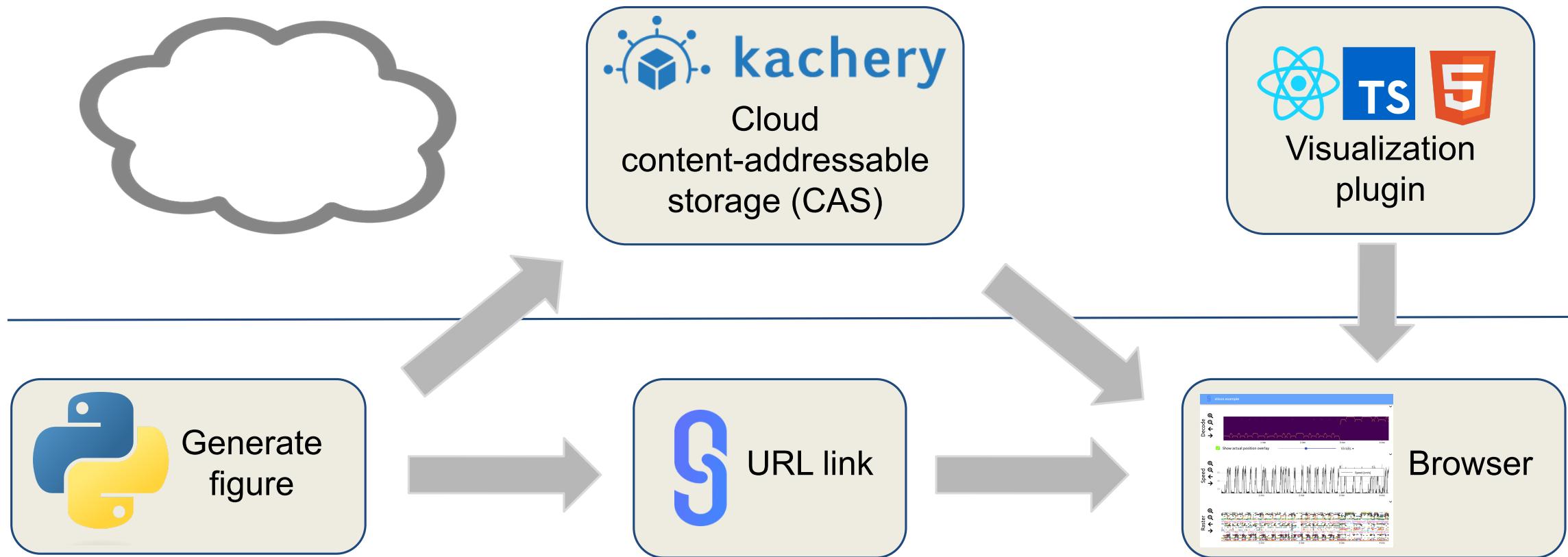
# Output:
# https://figurl.org/f?v=gs://figurl/plotly-1
# &d=sha1://5c6ec276ce9a3b20b208aaff911b037ce4052e51
# &label=plotly%20example%20-%20iris%203d

```

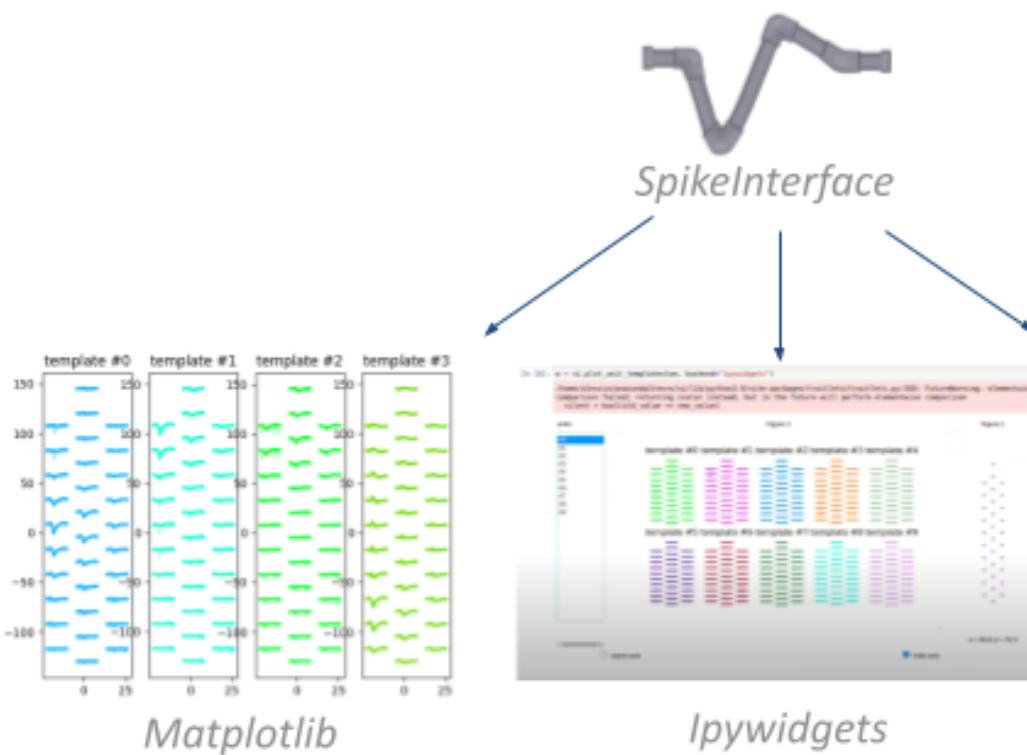


[Figurl link](#)

Figurl architecture

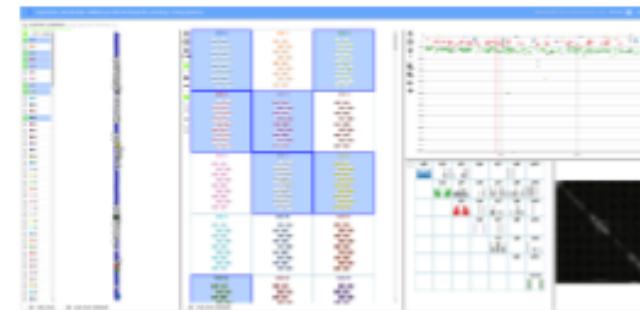


Figurl / SpikeInterface integration (with Alessio Buccino)



SpiketInterface can produce ephys / spike sorting visualizations in various formats.

The Figurl format is highly interactive and can be [shared](#) without any software installation.



Figurl / SortingView

Figurl / SpikeInterface integration

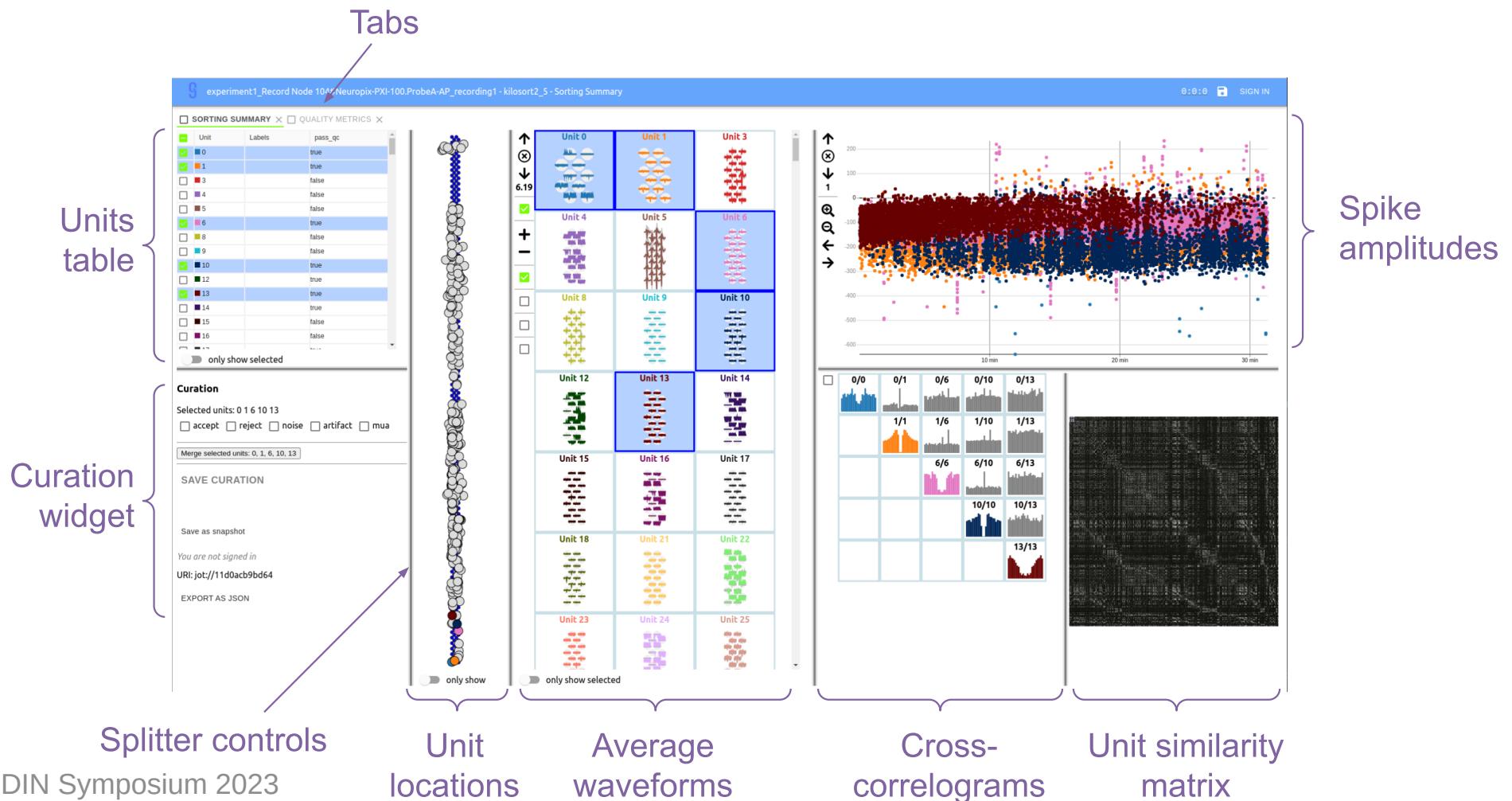
```
import spikeextractors as se

# Load the recording and sorting
recording, sorting = ...

# prepare SpikeInterface widget
widget = ...

# Prepare and print the figURL
url = widget.url(label='example')
print(url)
```

Figurl / SpikeInterface integration (figurl)

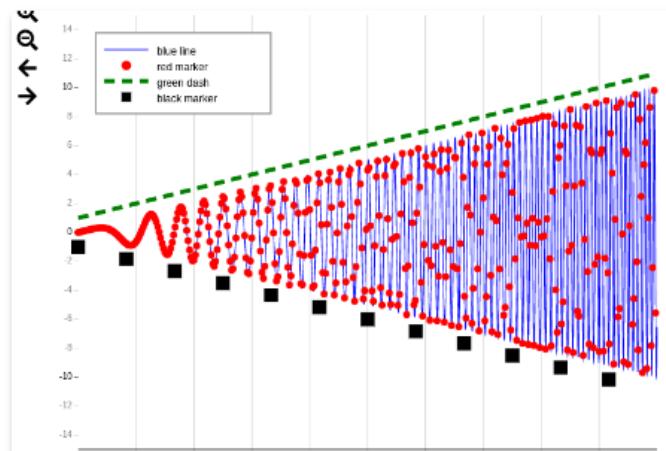


Figurl: Other examples - gallery



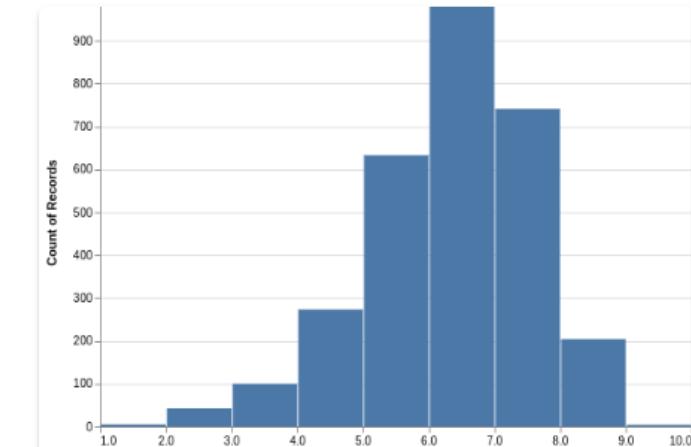
Plotly

The Figurl Plotly interface allows any Plotly figure created in Python to be shared as a Figurl figure.

[LEARN MORE](#)

TimeseriesGraph

Scrollable view of timeseries elements, synchronized with other timeseries views.

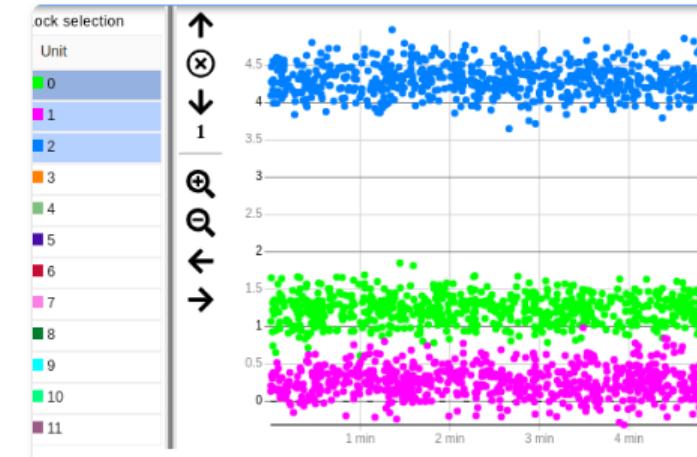
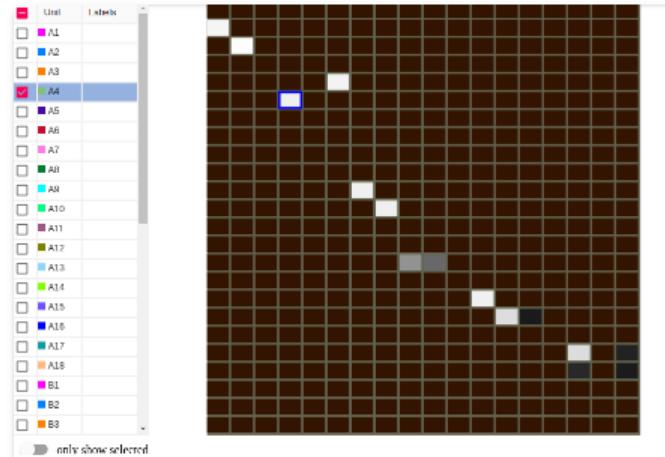
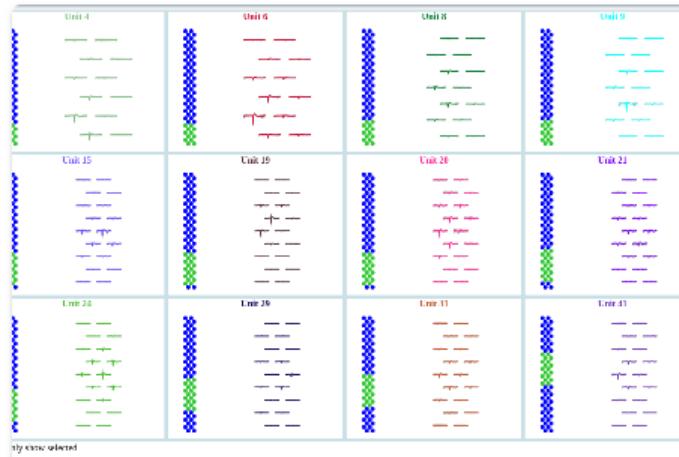
[LEARN MORE](#)

Altair

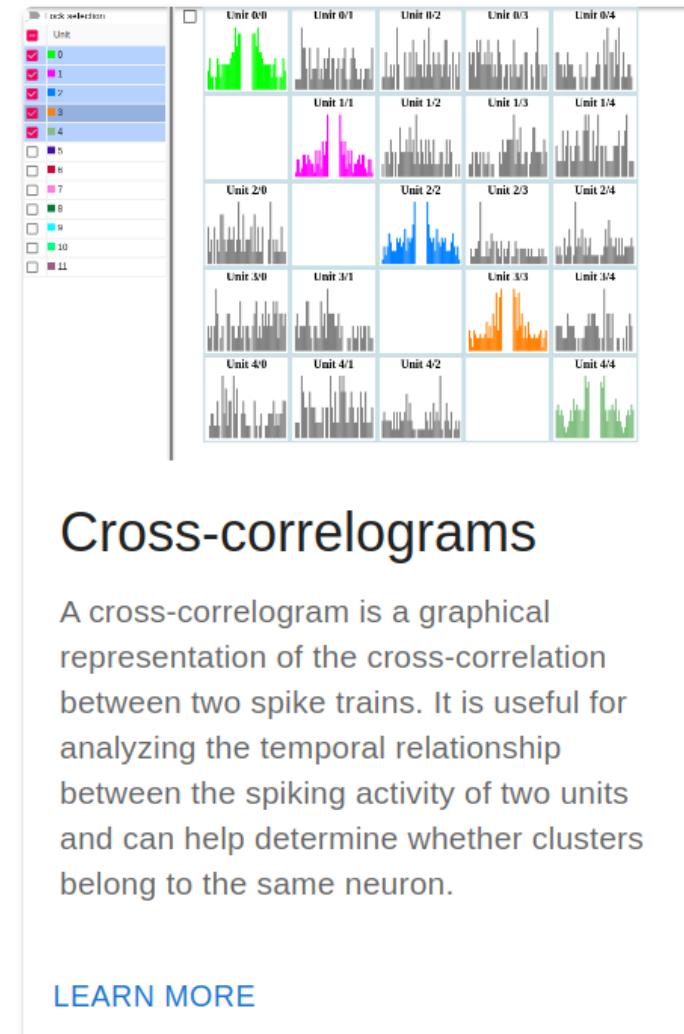
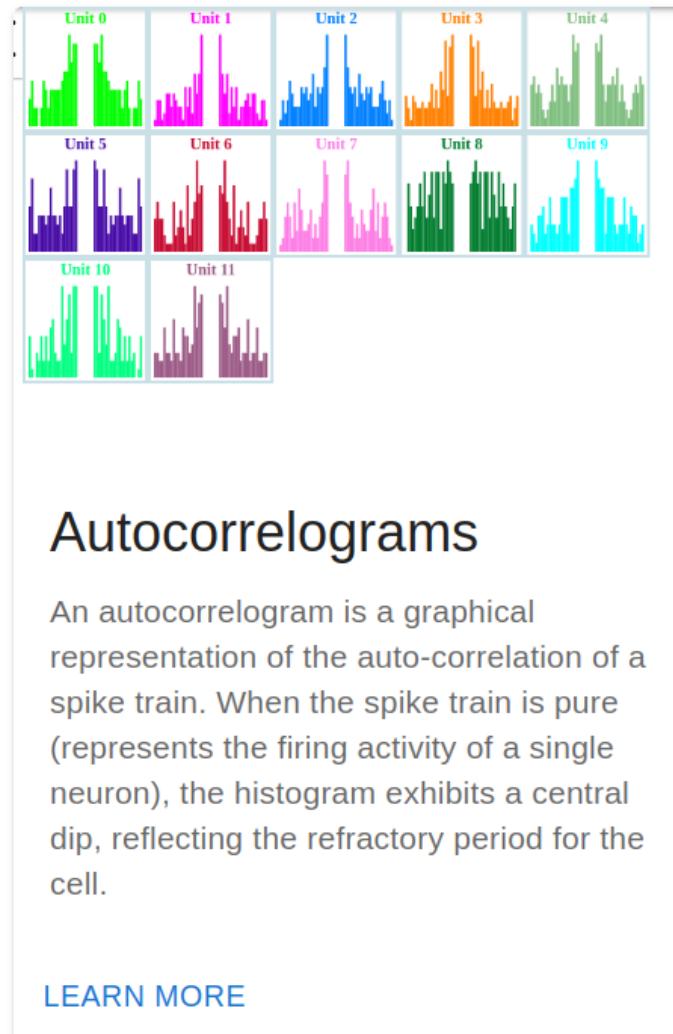
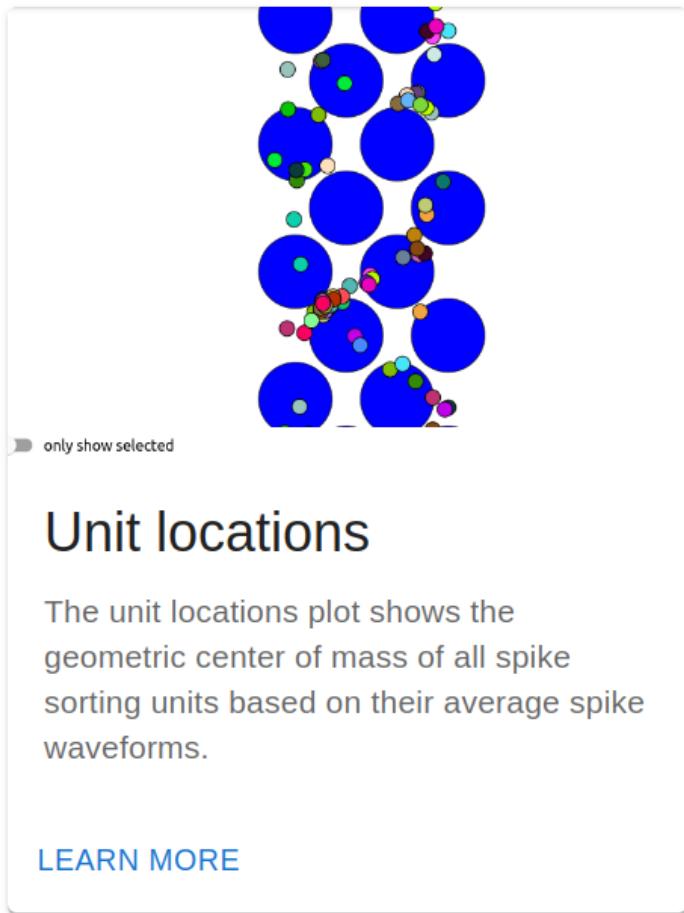
The Figurl Altair interface allows any Altair chart created in Python to be shared as a Figurl figure.

[LEARN MORE](#)

Figurl: Other examples - gallery



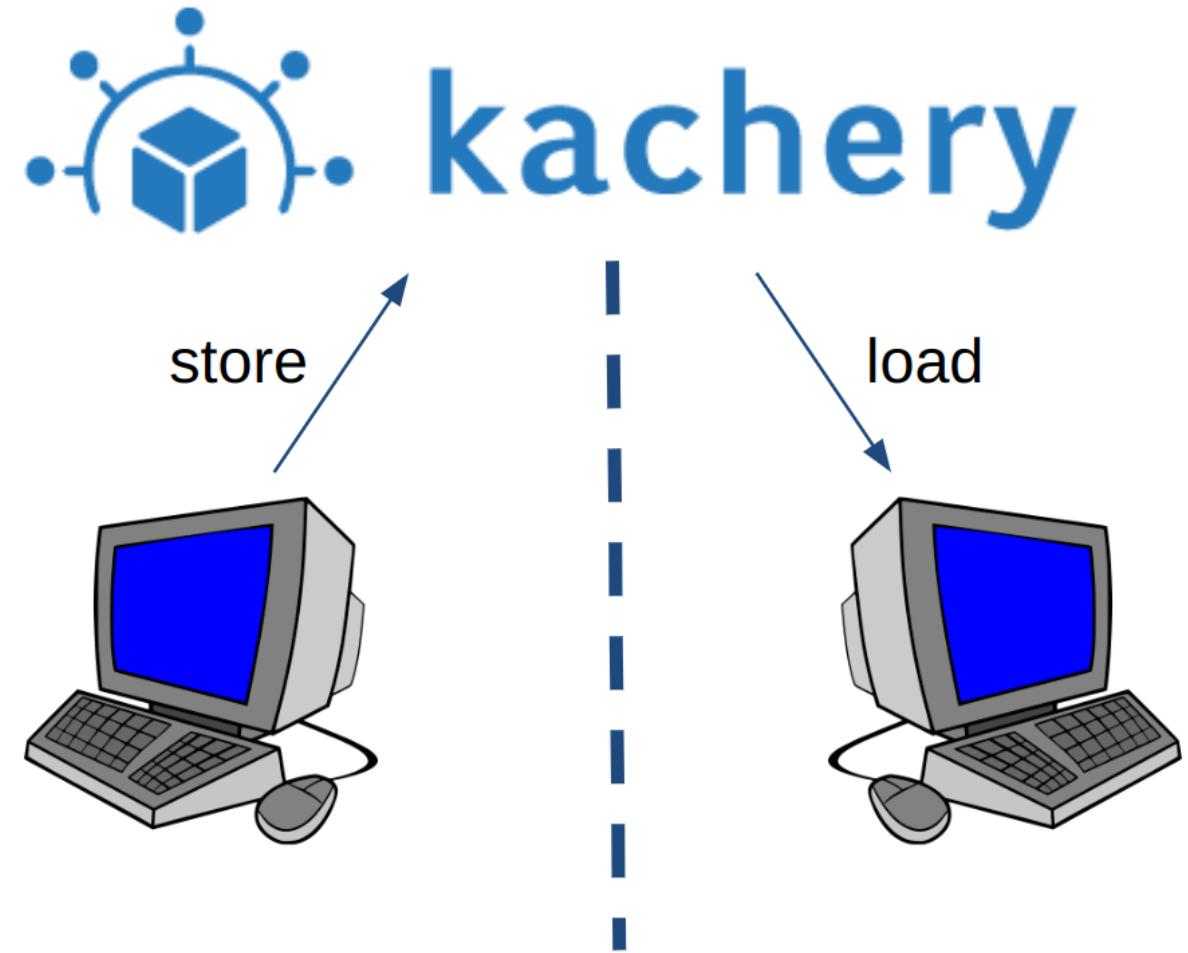
Figurl: Other examples - gallery



Figurl uses Kachery

Kachery is a Content Addressable Storage database in the cloud

- Minimal configuration for upload
- Download from anywhere
- Python client or Command-line client
- Serverless infrastructure
- Organized into zones (labs can host zones / pay for storage)



Storing kachery data

```
echo "test-content" > test_content.txt
kachery-cloud-store test_content.txt
# output:
# sha1://b971c6ef19b1d70ae8f0feb989b106c319b36230?label=test_content.txt
```

From Python

```
uri = kcl.store_text('example text', label='example.txt')
# uri = "sha1://d9e989f651cdd269d7f9bb8a215d024d8d283688?label=example.txt"
```

Retrieving kachery data

```
kachery-cloud-load sha1://b971c6ef19b1d70ae8f0feb989b106c319b36230
```

```
w = kcl.load_text('sha1://d9e989f651cdd269d7f9bb8a215d024d8d283688?label=example.txt')  
x = kcl.load_json('sha1://d0d9555e376ff13a08c6d56072808e27ca32d54a?label=example.json')  
y = kcl.load_npy("sha1://bb55205a2482c6db2ace544fc7d8397551110701?label=example.npy")  
z = kcl.load_pkl("sha1://20d178d5a1264fc3267e38ca238c23f3e2dcd5d2?label=example.pkl")
```

Neurosift: overview

- Visualize / browse NWB files hosted on DANDI (and elsewhere)
- Also view local NWB files
- No installation required
- No server backend required (client-only)
- Efficiently read data lazily from HDF5 files (h5wasm fork)
- Similar to NWBWidgets
- Developed in collaboration with CatalystNeuro

Neurosift / DANDI integration

Browse to a DANDI NWB file and click to open in Neurosift

	000463 / sub-BH395	Size
..		
 sub-BH395_ses-20221213T174039_ecophys.nwb	 EXTERNAL SERVICES  Neurosift	4.5 GB
 sub-BH395_ses-20221213T180853_ecophys.nwb		4.5 GB
 sub-BH395_ses-20221213T183140_ecophys.nwb	   	4.5 GB
 sub-BH395_ses-20221213T185547_ecophys.nwb	   	4.4 GB
 sub-BH395_ses-20221213T195834_ecophys.nwb	   	1.8 GB

Neurosift: NWB file browser

Neurosift ★ This viewer is in alpha and is under active development ★

main

DANDISSET: [000003 0.230629.1955](#)

[sub-YutaMouse33](#)/[sub-YutaMouse33_ses-YutaMouse33-150222_behavior+ecephys.nwb](#)

Physiological Properties and Behavioral Correlates of Hippocampal Granule Cells and Mossy Cells

Session ID	YutaMouse33-150222
Experimenter	Yuta Senzai
Lab	Buzsaki
Institution	NYU
Related publications	DOI:10.1016/j.neuron.2016.12.011
Session description	Mouse in open exploration and theta maze.
Identifier	0210547d-d252-429b-a39c-6ee1a4e8ea3e
Session start	2015-02-22T00:00:05.00
Timestamps ref.	2015-02-22T00:00:05.00
File creation	2021-01-16T23:45:47.998974-05:00
NWB version	2.2.5

No views selected

DEFAULT RAW

Item	Neurodata type	Description	Comments	Data
ch_SsolR	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_arm	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_dig1	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_dig2	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_entL	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_entR	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_soiL	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_soiR	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
ch_wait	TimeSeries	Environmental electrode recorded inline with neural data.	no comments	<h [76293625]
X/Y position_sensor0	SpatialSeries	raw sensor data from sensor 0	no comments	<d [2384179, 2]
X/Y position_sensor1	SpatialSeries	raw sensor data from sensor 1	no comments	<d [2384179, 2]

▶ analysis (0)

▶ general (8)

▼ processing/behavior (1)

Item	Neurodata type	Description	Comments	Data
PSTH	states	TimeIntervals	sleep states of animal	

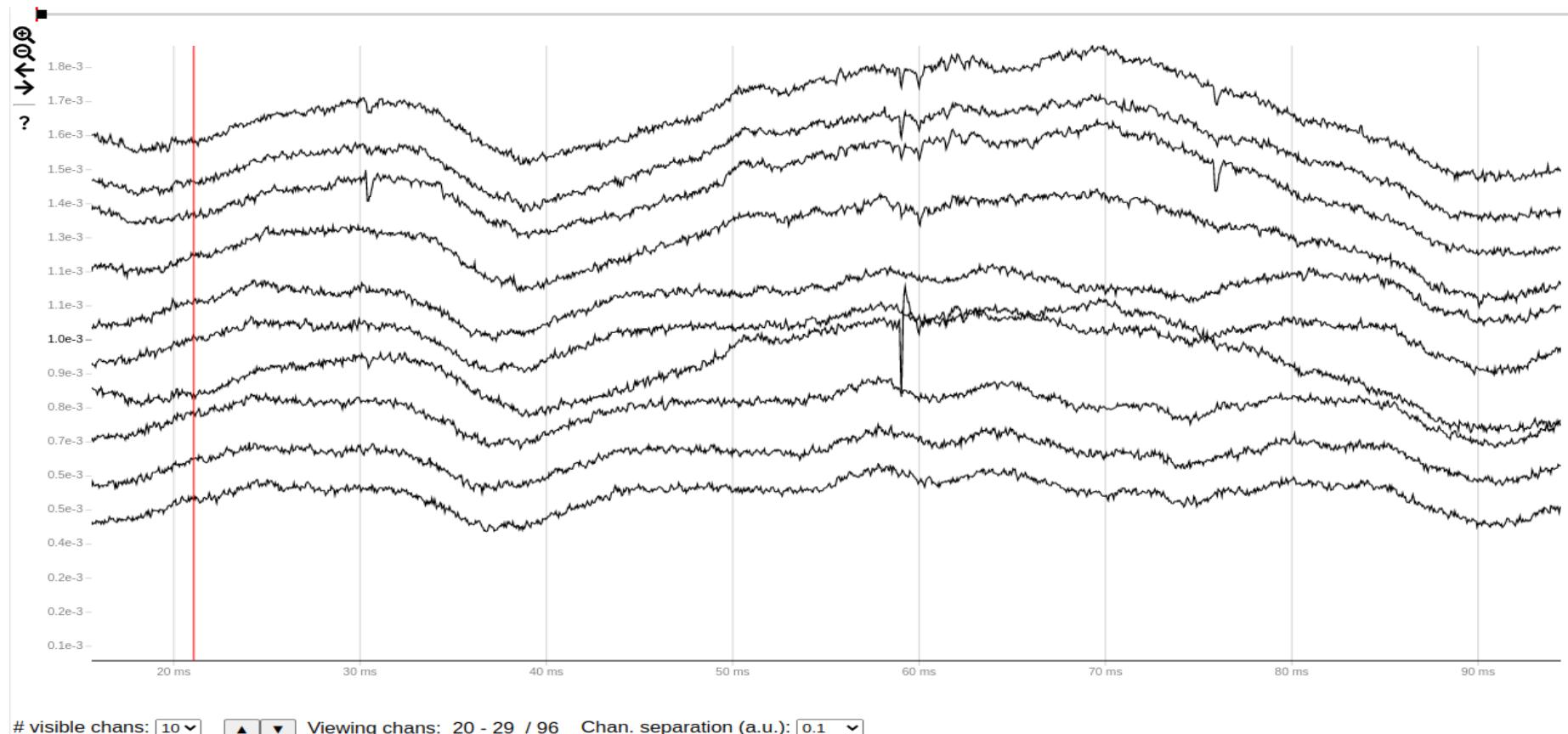
▼ processing/ecephys (2)

Item	Neurodata type	Description	Comments	Data
LFP/LFP	ElectricalSeries	Local field potential signal.	Generated from SpikeInterface::NwbRecordingExtract or	<h [76293625, 96]
LFPDecompositionSeries	DecompositionSeries	Theta and Gamma phase for reference LFP	no comments	<d [76293625, 1, 2]

▶ specifications (2)

[View timeseries alignment](#)

Neurosift: ElectricalSeries



Neurosift: lazy loading from remote HDF5 files

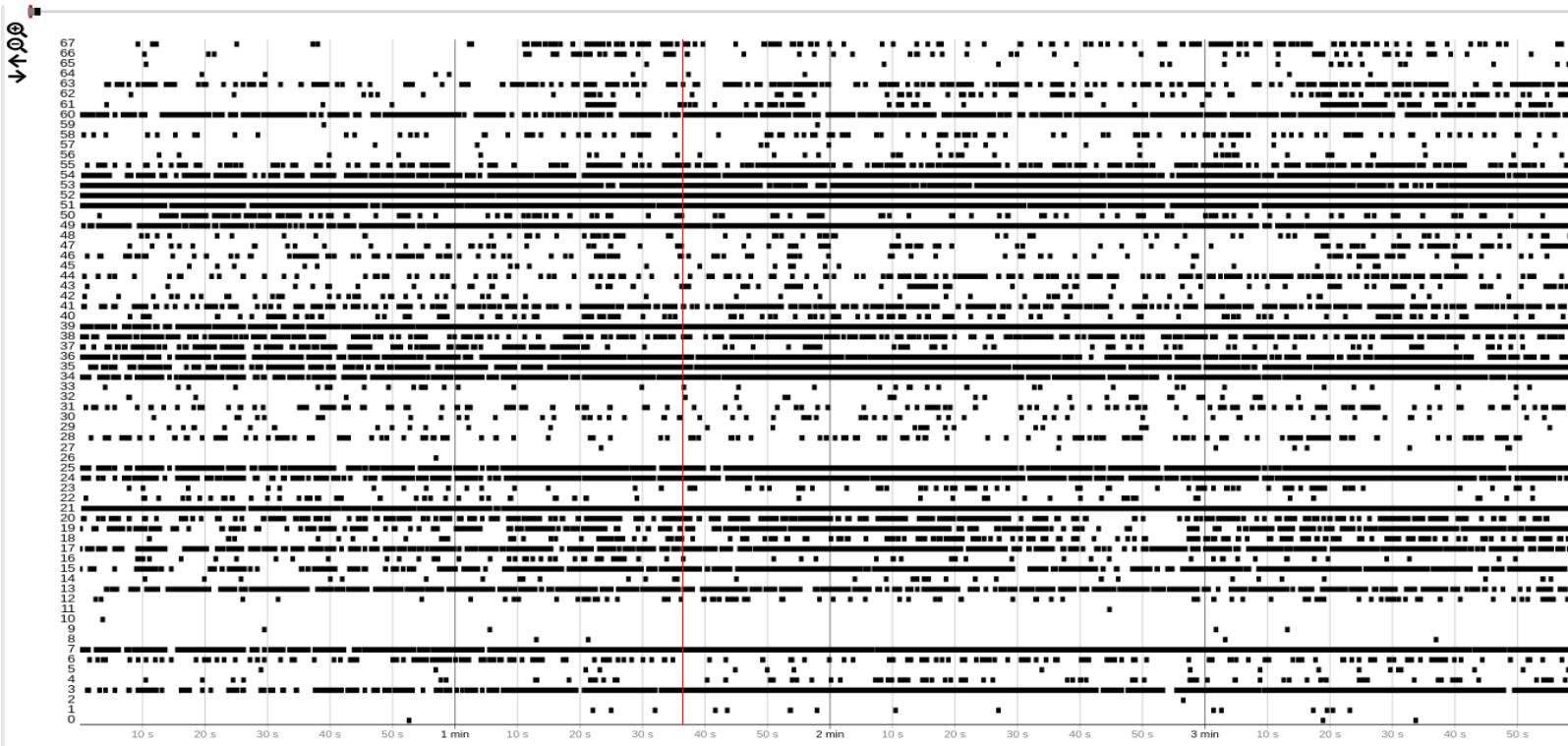
Forked version of h5wasm that uses efficient/smart chunking optimized for reading HDF5 using WebAssembly and web workers.

<https://github.com/usnistgov/h5wasm>

[https://github.com/flatironinstitute/neurosift/tree/main/gui/src/pages/
NwbPage/RemoteH5File/h5wasm](https://github.com/flatironinstitute/neurosift/tree/main/gui/src/pages/NwbPage/RemoteH5File/h5wasm)



Neurosift: Raster plot (spike trains)

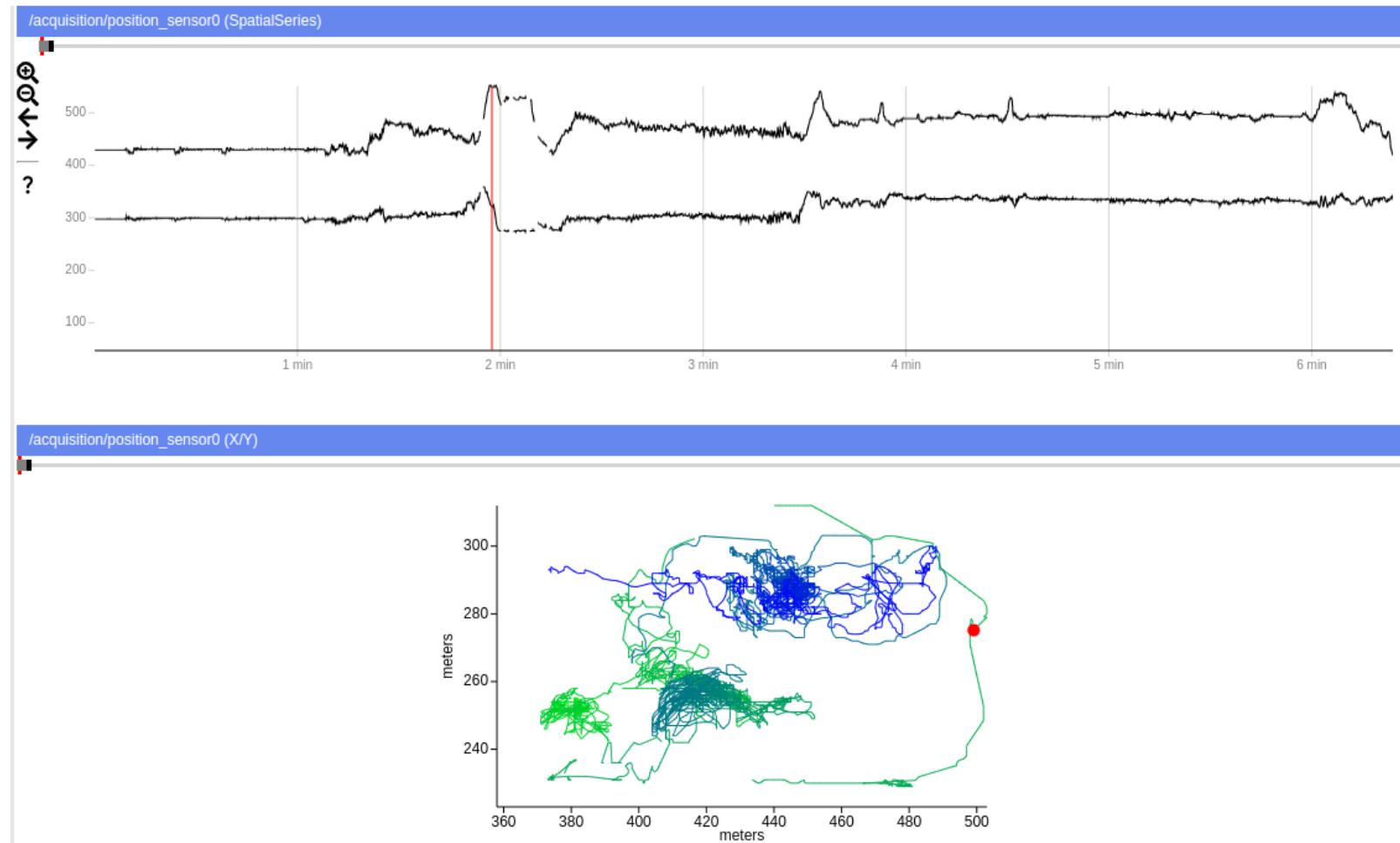


[link](#)

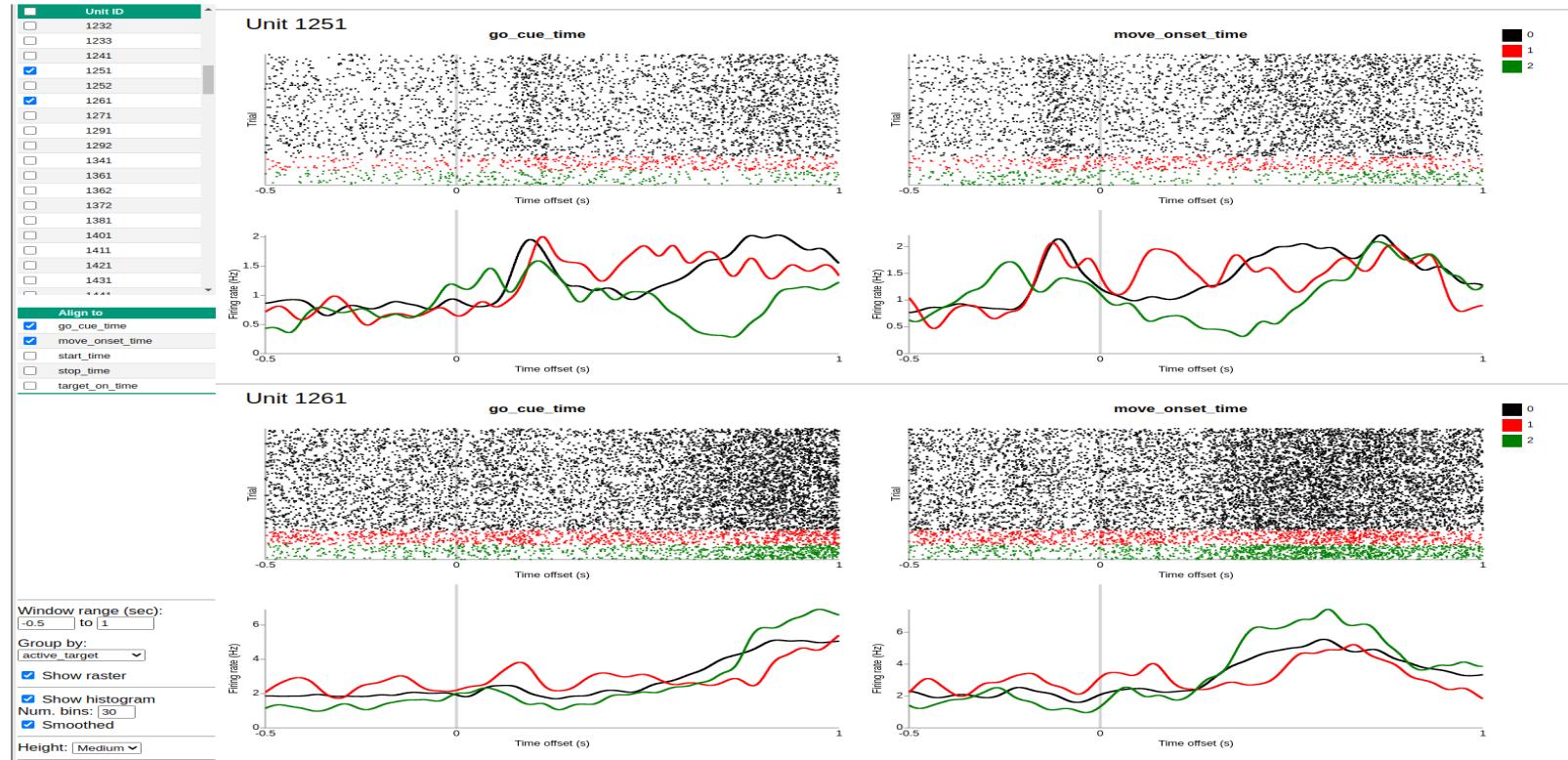
Neurosift: Synchronized views

▼ acquisition (13)		
Item	Neurodata type	
<input type="checkbox"/> ElectricalSeries	ElectricalSeries	
<input type="checkbox"/> ch_SsoL	TimeSeries	
<input type="checkbox"/> ch_SsoR	TimeSeries	
<input type="checkbox"/> ch_arm	TimeSeries	
<input type="checkbox"/> ch_dig1	TimeSeries	
<input type="checkbox"/> ch_dig2	TimeSeries	
<input type="checkbox"/> ch_entL	TimeSeries	
<input type="checkbox"/> ch_entR	TimeSeries	
<input type="checkbox"/> ch_soiL	TimeSeries	
<input type="checkbox"/> ch_soiR	TimeSeries	
<input type="checkbox"/> ch_wait	TimeSeries	
<input checked="" type="checkbox"/> X/Y	position_sensor0	SpatialSeries
<input type="checkbox"/> X/Y	position_sensor1	SpatialSeries

► analysis (0)
► general (8)
► processing/behavior (1)
► processing/ecephys (2)
► specifications (2)
► stimulus (2)
► units (68 units)



Neurosift: Peri-stimulus time histogram (PSTH)



[link](#)

Neurosift examples

<https://github.com/flatironinstitute/neurosift/wiki/Neurosift-DANDI-Examples>

Neurodata types: [ImageSegmentation](#) | [SpatialSeries](#) | [TwoPhotonSeries](#) | [TimeSeries](#) |
[TimeIntervals](#) | [PSTH](#) | [ElectricalSeries](#) | [LabeledEvents](#) | [ImageSeries](#) | [RasterPlot](#) |
[Autocorrelograms](#) | [Images](#) | [BehavioralEvents](#) | ...

Neurosift: viewing local files

- Prerequisite: [NodeJS v16 or higher](#)
- Install the neurosift Python package: `pip install --upgrade neurosift`
- Run the following command in a terminal window:

```
neurosift view-nwb /path/to/file.nwb
```

Will open Neurosift in a browser.

ProtocaaS: overview

ProtocaaS is a **prototype** web-based tool for analyzing neurophysiology data in the cloud (or with cluster/local compute).

- See also: [NeuroCAAS](#), [SpikeInterface-cloud](#)
- Under heavy development with Ben Dichter and Luiz Tauffer (last couple of months)
- Enable labs to use cloud resources to run analysis pipelines on their data
- Also allow using local machines or compute clusters
- Initial focus is spike sorting, but will expand
- Tight integration with NWB and DANDI
- Uses [SpikeInterface](#) and [spikeinterface-dockerfiles](#)

Protocaa~~s~~: spike sorting steps

- Step 1: Prepare NWB files, **create a dandiset**, upload raw data (or use existing)
- Step 2: Create a Protocaa~~s~~ project and **import the dandiset** (does not create a copy)
- Step 3: **Launch spike sorting** from the web GUI (configure compute resource)
- Step 4: Browse outputs using Neurosift
- Step 5: **Upload outputs to dandiset** (existing or new)

Protocaaas spike sorting: Create a project

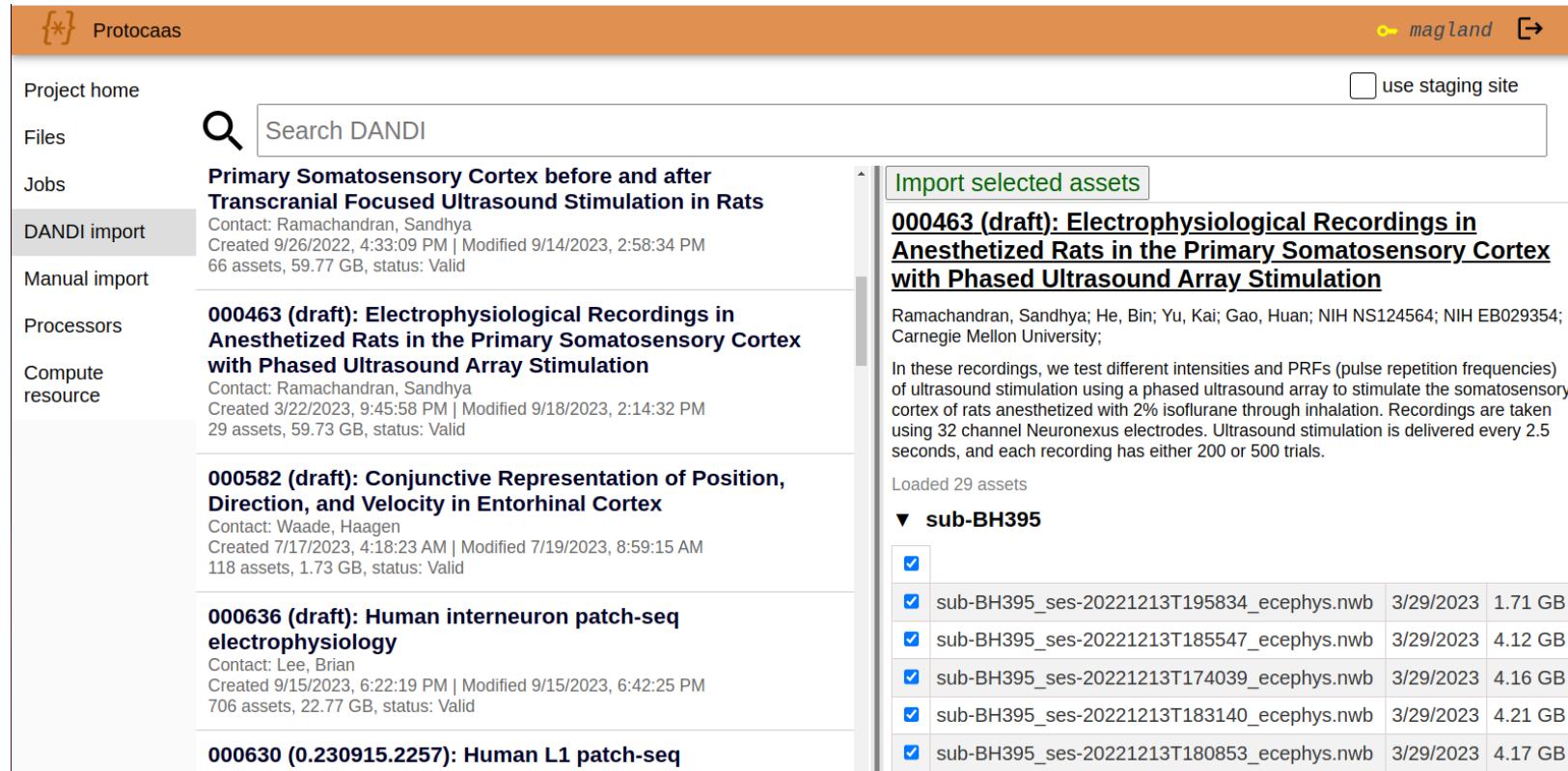


Create a project

Project name:

- Log in with GitHub
- [Give it a try](#) - just a prototype

Protocas spike sorting: Import the raw NWB files from DANDI



The screenshot shows the Protocas software interface. The left sidebar has a navigation menu with options: Project home, Files, Jobs, DANDI import (which is selected), Manual import, Processors, and Compute resource. The main area displays a search bar with "Search DANDI" and a checkbox for "use staging site". A list of imported assets is shown:

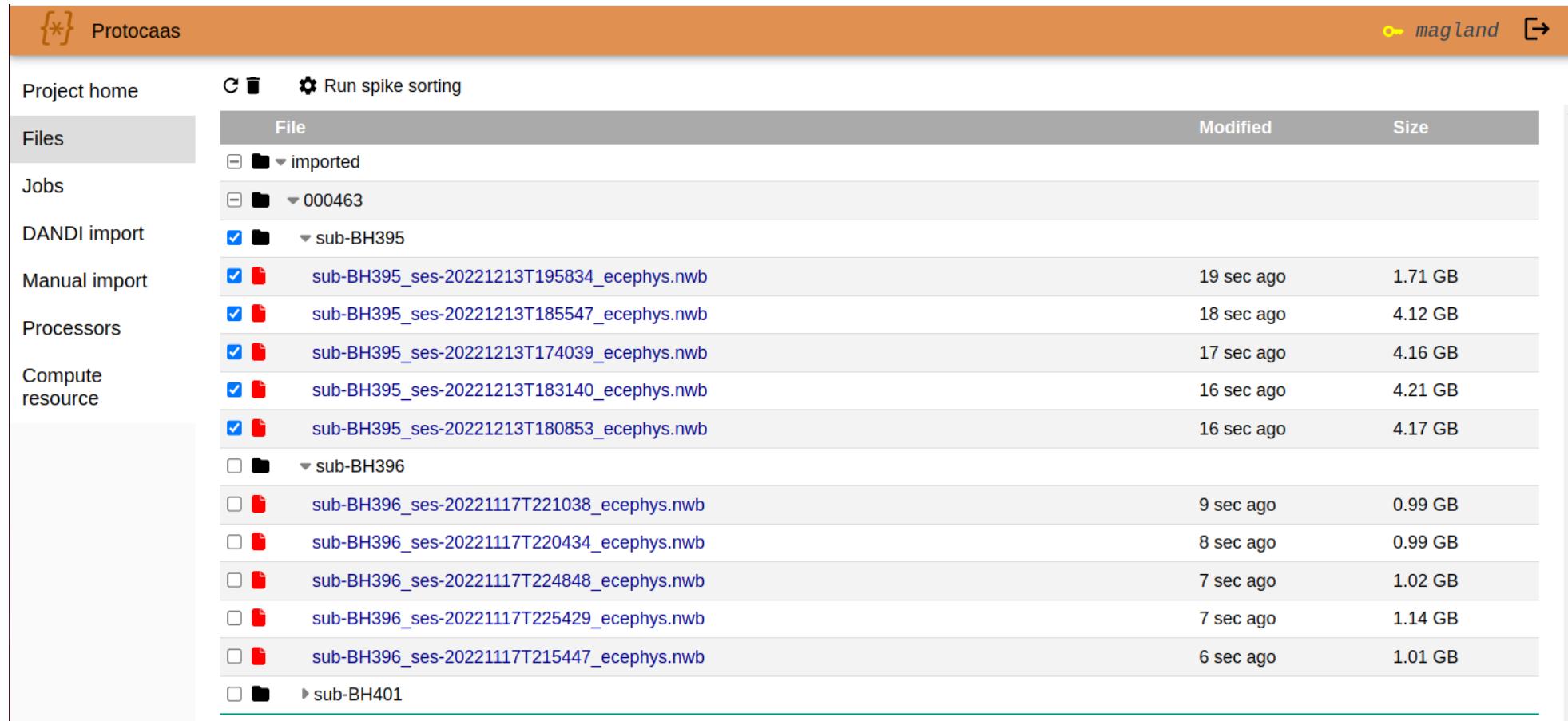
- Primary Somatosensory Cortex before and after Transcranial Focused Ultrasound Stimulation in Rats**
 Contact: Ramachandran, Sandhya
 Created 9/26/2022, 4:33:09 PM | Modified 9/14/2023, 2:58:34 PM
 66 assets, 59.77 GB, status: Valid
- 000463 (draft): Electrophysiological Recordings in Anesthetized Rats in the Primary Somatosensory Cortex with Phased Ultrasound Array Stimulation**
 Contact: Ramachandran, Sandhya; He, Bin; Yu, Kai; Gao, Huan; NIH NS124564; NIH EB029354; Carnegie Mellon University;
 In these recordings, we test different intensities and PRFs (pulse repetition frequencies) of ultrasound stimulation using a phased ultrasound array to stimulate the somatosensory cortex of rats anesthetized with 2% isoflurane through inhalation. Recordings are taken using 32 channel Neuronexus electrodes. Ultrasound stimulation is delivered every 2.5 seconds, and each recording has either 200 or 500 trials.
- 000582 (draft): Conjunctive Representation of Position, Direction, and Velocity in Entorhinal Cortex**
 Contact: Waade, Haagen
 Created 7/17/2023, 4:18:23 AM | Modified 7/19/2023, 8:59:15 AM
 118 assets, 1.73 GB, status: Valid
- 000636 (draft): Human interneuron patch-seq electrophysiology**
 Contact: Lee, Brian
 Created 9/15/2023, 6:22:19 PM | Modified 9/15/2023, 6:42:25 PM
 706 assets, 22.77 GB, status: Valid
- 000630 (0.230915.2257): Human L1 patch-seq**

A right-hand panel titled "Import selected assets" shows a list of checked items under "sub-BH395":

	sub-BH395_ses-20221213T195834_ecephys.nwb	3/29/2023	1.71 GB
<input checked="" type="checkbox"/>	sub-BH395_ses-20221213T185547_ecephys.nwb	3/29/2023	4.12 GB
<input checked="" type="checkbox"/>	sub-BH395_ses-20221213T174039_ecephys.nwb	3/29/2023	4.16 GB
<input checked="" type="checkbox"/>	sub-BH395_ses-20221213T183140_ecephys.nwb	3/29/2023	4.21 GB
<input checked="" type="checkbox"/>	sub-BH395_ses-20221213T180853_ecephys.nwb	3/29/2023	4.17 GB

Does not create a copy (links only)
 Magland ODIN Symposium 2023

Protocaastr spike sorting: Select files for sorting



The screenshot shows the Protocaastr web interface. The top navigation bar includes a user icon, the project name "Protocaastr", a user name "magland", and a sign-out button. On the left, a sidebar lists project sections: Project home, Files (selected), Jobs, DANDI import, Manual import, Processors, and Compute resource. The main content area displays a table of files under the "Files" section. The table has columns for "File", "Modified", and "Size". A "Run spike sorting" button is located at the top of the table. The file list includes several entries for sub-BH395 and sub-BH396 sessions, with some files checked for selection.

File	Modified	Size
imported		
000463		
sub-BH395		
sub-BH395_ses-20221213T195834_ecophys.nwb	19 sec ago	1.71 GB
sub-BH395_ses-20221213T185547_ecophys.nwb	18 sec ago	4.12 GB
sub-BH395_ses-20221213T174039_ecophys.nwb	17 sec ago	4.16 GB
sub-BH395_ses-20221213T183140_ecophys.nwb	16 sec ago	4.21 GB
sub-BH395_ses-20221213T180853_ecophys.nwb	16 sec ago	4.17 GB
sub-BH396		
sub-BH396_ses-20221117T221038_ecophys.nwb	9 sec ago	0.99 GB
sub-BH396_ses-20221117T220434_ecophys.nwb	8 sec ago	0.99 GB
sub-BH396_ses-20221117T224848_ecophys.nwb	7 sec ago	1.02 GB
sub-BH396_ses-20221117T225429_ecophys.nwb	7 sec ago	1.14 GB
sub-BH396_ses-20221117T215447_ecophys.nwb	6 sec ago	1.01 GB
sub-BH401		

Protocaaas spike sorting: Select sorting parameters

X

Batch spike sorting of 5 files using kilosort3

Overwrite existing outputs	<input type="checkbox"/>
Description string in output file name	kilosort3 /*

Submit

input	*	input .nwb file
output	*	output .nwb file
electrical_series_path	/acquisition/ElectricalSeries	Path to the electrical series in the NWB file, e.g., /acquisition/ElectricalSeries
detect_threshold	6	Threshold for spike detection
projection_threshold	9, 9	Threshold on projections
preclust_threshold	8	Threshold crossings for pre-clustering (in PCA projection space)
car	<input checked="" type="checkbox"/>	Enable or disable common reference
minFR	0.2	Minimum spike rate (Hz), if a cluster falls below this for too long it gets removed
minfr_goodchannels	0.2	Minimum firing rate on a "good" channel
nblocks	5	blocks for registration. 0 turns it off, 1 does rigid registration. Replaces "datashift" option.
sig	20	spatial smoothness constant for registration
freq_min	300	High-pass filter cutoff frequency

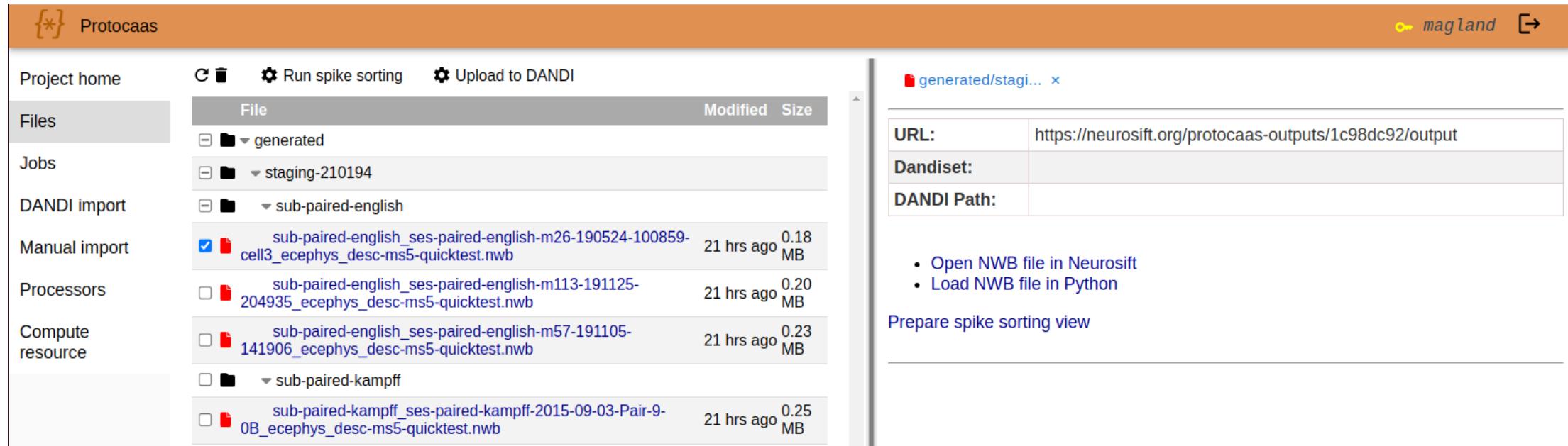
Protocaaas spike sorting: Monitor running jobs

C trash

		Job	Processor	Status	User	Created	Compute
<input type="checkbox"/>	●	Batch fepo5pf6	mountainsort5	running	magland	39 sec ago	ccmlin008
<input type="checkbox"/>	●	9165aa40	mountainsort5	pending	magland	37 sec ago	ccmlin008
<input type="checkbox"/>	●	75d420ed	mountainsort5	pending	magland	37 sec ago	ccmlin008
<input type="checkbox"/>	●	cc8ff08b	mountainsort5	pending	magland	38 sec ago	ccmlin008
<input type="checkbox"/>	●	4f972eea	mountainsort5	running	magland	38 sec ago	ccmlin008
<input type="checkbox"/>	●	10bddad3	mountainsort5	running	magland	39 sec ago	ccmlin008

- Jobs are running on the compute resource (AWS, Slurm, or local machine)
- Monitor their status through the web GUI

Protocaaas spike sorting: Browse results in Neurosift



The screenshot shows the Protocaaas project page in Neurosift. The left sidebar lists project sections: Project home, Files, Jobs, DANDI import, Manual import, Processors, and Compute resource. The 'Files' section is selected, displaying a table of uploaded files. The table columns are File, Modified, and Size. The table rows include:

File	Modified	Size
generated		
staging-210194		
sub-paired-english		
sub-paired-english_ses-paired-english-m26-190524-100859-cell3_ecophys_desc-ms5-quicktest.nwb	21 hrs ago	0.18 MB
sub-paired-english_ses-paired-english-m113-191125-204935_ecophys_desc-ms5-quicktest.nwb	21 hrs ago	0.20 MB
sub-paired-english_ses-paired-english-m57-191105-141906_ecophys_desc-ms5-quicktest.nwb	21 hrs ago	0.23 MB
sub-paired-kampff		
sub-paired-kampff_ses-paired-kampff-2015-09-03-Pair-9-0B_ecophys_desc-ms5-quicktest.nwb	21 hrs ago	0.25 MB

The right panel shows a detailed view of a specific file entry: 'generated/stagi...'. It includes fields for URL (https://neurosift.org/protocaaas-outputs/1c98dc92/output), Dandiset, and DANDI Path. Below these fields are two actions: 'Open NWB file in Neurosift' and 'Load NWB file in Python'. A link to 'Prepare spike sorting view' is also present.

- When jobs complete, they upload results to a cloud bucket
- Output files are added to the Protocaaas project (pointers to the cloud outputs)
- Protocaaas project contains all provenance information

Protocaaas spike sorting: Upload results to DANDI

Set API Keys

DANDI API

Key:

DANDI Staging

API Key:

DANDI Upload

Dandiset ID:	210194
Dandiset version:	draft
DANDI Instance:	dandi-staging
Files:	sub-paired-english/sub-paired-english_ses-paired-english-m26-190524-100859-cell3_ecephys_desc-ms5-quicktest.nwb sub-paired-english/sub-paired-english_ses-paired-english-m113-191125-204935_ecephys_desc-ms5-quicktest.nwb sub-paired-english/sub-paired-english_ses-paired-english-m57-191105-141906_ecephys_desc-ms5-quicktest.nwb sub-paired-kampff/sub-paired-kampff_ses-paired-kampff-2015-09-03-Pair-9-0B_ecephys_desc-ms5-quicktest.nwb sub-paired-kampff/sub-paired-kampff_ses-paired-kampff-2014-11-25-Pair-3-0_ecephys_desc-ms5-quicktest.nwb sub-paired-kampff/sub-paired-kampff_ses-paired-kampff-c14_ecephys_desc-ms5-quicktest.nwb

Protocaaas spike sorting: Provenance stored in DANDI metadata

Django REST framework jmagland@flatironinstitute.org

```
},
{
  "job": {
    "inputFiles": [
      {
        "name": "input",
        "fileName": "imported/staging-210194/sub-paired-english/sub-paired-english_ses-paired-english-m26-190524-100859-cell3_ecephys.nwb",
        "metadata": {
          "dandisetId": "210194",
          "dandiAssetId": "23e499b7-3aae-4eb2-ba40-a1bf30d35298",
          "dandiStaging": true,
          "dandiAssetPath": "sub-paired-english/sub-paired-english_ses-paired-english-m26-190524-100859-cell3_ecephys.nwb",
          "dandisetVersion": "draft"
        }
      }
    ],
    "outputFiles": [
      {
        "name": "output",
        "fileName": "generated/staging-210194/sub-paired-english/sub-paired-english_ses-paired-english-m26-190524-100859-cell3_ecephys_desc-ms.nwb"
      }
    ],
    "inputParameters": [
      {
        "name": "electrical_series_path",
        "value": "/acquisition/ElectricalSeries"
      },
      {
        "name": "test_duration_sec",
        "value": 60
      }
    ],
    "timestampCreated": 1696264273.9963758,
    "computeResourceId": "56f3e6f98806c31ceada2035daa809c84eaa42f8d551eb774e79a4c16b70b860",
    "timestampFinished": 1696264425.544048
  },
  "name": "protocaaas",
  "jobId": "lc98dc92",
  "projectId": "1d70eda8",
  "workspaceId": "nyxoglw",
  "processorName": "ms5_quicktest",
  "processorVersion": null
}
```

Protocaaas: advantages

- Scalability: you can spike sort many sessions at once
- Reproducibility: use AWS and get the exact same results
- Provenance: Jobs store all necessary metadata
- Compute flexibility: local, slurm cluster, cloud
- Access to sorters: not always easy to set up even when using SpikeInterface
- Collaboration: add other GitHub users to a workspace
- Integration with DANDI

Protocaaas: Initial experiments

Protocaaas

magland

Project home Run spike sorting

Files

File

		Modified	Size
<input type="checkbox"/>	.spike_sorting_figurl		
<input type="checkbox"/>	generated		
<input type="checkbox"/>	000409		
<input type="checkbox"/>	sub-CSH-ZAD-001		
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort3-10min.nwb	3 days ago	8.56 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort3-20min.nwb	3 days ago	16.86 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-mountainsort5-10min-training100-b.nwb	3 days ago	4.68 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort2-5-10min.nwb	3 days ago	7.45 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort3.nwb	2 days ago	13.86 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort2-5.nwb	2 days ago	29.27 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort2-5-keepall.nwb	2 days ago	42.26 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort3-keepall.nwb	2 days ago	17.68 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort2-5-40min.nwb	2 days ago	27.91 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-kilosort3-40min.nwb	2 days ago	34.91 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-mountainsort5-20min-training100.nwb	2 days ago	7.40 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-mountainsort5-40min-training100.nwb	1 day ago	14.87 MB
<input type="checkbox"/>	sub-CSH-ZAD-001_ses-3e7ae7c0-fe8b-487c-9354-036236fa1010_behavior+ecephys+image_desc-mountainsort5-training100.nwb	1 day ago	29.59 MB
<input type="checkbox"/>	000618		
<input type="checkbox"/>	sub-hybrid-janelia		
<input type="checkbox"/>	sub-hybrid-janelia_ses-hybrid-drift-sprobe-rec-16c-600s-12_ecephys_desc-kilosort3.nwb	12 hrs ago	0.52 MB
<input type="checkbox"/>	sub-hybrid-janelia_ses-hybrid-drift-sprobe-rec-16c-600s-12_ecephys_desc-mountainsort5.nwb	12 hrs ago	0.57 MB
<input type="checkbox"/>	sub-hybrid-janelia_ses-hybrid-drift-sprobe-rec-16c-600s-12_ecephys_desc-kilosort2-5.nwb (failed)		
<input type="checkbox"/>	sub-paired-english		
<input type="checkbox"/>	sub-paired-english_ses-paired-english-m108-191125-163508_ecephys_desc-kilosort2-5.nwb	12 hrs ago	2.55 MB

<https://protocaaas.vercel.app/project/4b4d3486?tab=project-files>

ProtocaaS: containerized processors

- Each spike sorter is a processor within a ProtocaaS app
- Existing apps: [Kilosort 2.5](#), [Kilosort 3](#) and [MountainSort 5](#)
 - SpikeInterface
 - SpikeInterface-dockerfiles
- Apps live in containers for production, or on file systems during development
- It's relatively easy to make custom processor apps

ProtocaaS: compute resources

- Users can use the default compute resources
- Or configure their own compute resources
- A compute resource is a daemon that runs on some server or machine
 - Watches for new jobs in the ProtocaaS database
 - Spawns the jobs based on the configuration (AWS, Slurm, local machine)
- Jobs are autonomous
 - Capable of reporting their own status to the central system
 - Does not need to communicate with the daemon

Compute resource: resource1-slurm

Compute resource name	ccmlin008-slurm
Compute resource ID	ccmlin008-slurm
Owner	magland
Created	2 wks ago

Full ID: 6c0089dd5abd8ea4815f4ecb7e83a8a2c02677f5dca8852b7c6b4396e37f8f97

Apps

+ Add app  

	App	Executable path	Container	AWS Batch	Slurm
<input type="checkbox"/>	mountainsort5	/app/main	magland/pc-mountainsort5		CPUs per task: 4 Partition: ccm Time: 2:00:00
<input type="checkbox"/>	spike_sorting_utils	/app/main	magland/pc-spike-sorting-utils		CPUs per task: 4 Partition: ccm Time: 1:00:00
<input type="checkbox"/>	kilosort3	/app/main	magland/pc-kilosort3		CPUs per task: 4 Partition: gpu Time: 2:00:00 Other options: --gpus=1
<input type="checkbox"/>	spikingcircus	/app/main	magland/pc-spiking-circus	Job queue: spike-sorting-job-queue Job definition: spike-sorting-job-def	

Jobs

	Job	Processor	Status	User	Created	Compute	Role
<input type="checkbox"/>	Batch 689lpqj9	kilosort3	completed	magland	14 hrs ago	ccmlin008-slurm	
<input type="checkbox"/>	Batch fhpqflqd	ks3_quicktest	completed	magland	15 hrs ago	ccmlin008-slurm	
<input type="checkbox"/>	Batch qy1gkdw2	ks3_quicktest	failed	magland	16 hrs ago	ccmlin008-slurm	

Summary

Some tools for visualization and analysis of neurophysiology data:

- Figurl
 - Stable
- Neurosift
 - Initial version is stable
- ProtocaaS
 - Prototype
 - Under active development, with CatalystNeuro
 - Almost ready for beta testers
 - Please submit your name ideas!

Thank you

Figurl

Jeff Soules, Alessio Buccino, Loren Frank, Eric denovellis, Kyu Hyun Lee, Alison Comrie, Michael Coulter

Neurosift

Jeff Soules, Ben Dichter, Cody Baker

Protocaa

Ben Dichter, Luiz Tauffer

Thanks to the NWB and DANDI teams!