

# Einsatz von UNICORE in der Hirnforschung

2017-03-16 | Björn Hagemeyer

## Activities in Neuroscience

- SMHB - Supercomputing and Modeling for the Human Brain
- HBP - Human Brain Project

# UNICORE



Web    Command line    GUI    API

## Clients



Workflows    Jobs    Data Management    Discovery

## Services



Compute    Storage

## Resources



Users

Federations

Policies

## Security

# Polarized Light Imaging (PLI)

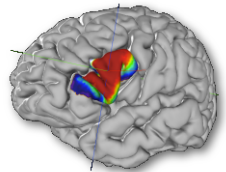
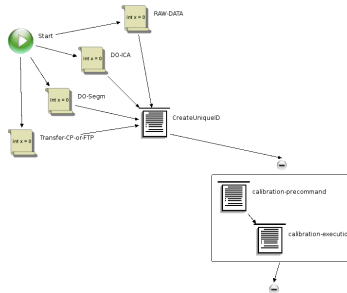
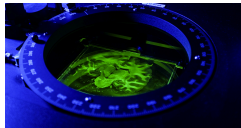
## The Method

- Developed at Institute for Neuroscience and Medicine, Forschungszentrum Jülich
- Aim: extract the course of single nerve fibers
- Sections of postmortem human brain, each  $70\mu m$
- Imaging the sections under linearly polarized light
  - $\sim 1500$  sections/brain
  - $30 \times 25$  tiles/section
  - $2048 \times 2048$  pixels/tile



# Polarized Light Imaging (PLI)

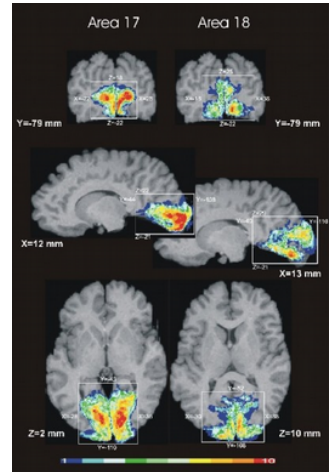
## The Workflow



# High-throughput brain scans

Jülich/HHU collaboration

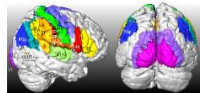
- Goal is to create a 3D brain atlas
- Data acquisition
  - Brain section scans (ex vivo)
    - (~2000 slices, 500GB per slice → 1PB)
  - MRT scans (in vivo)
- Processing: image registrytion, calibration, segmentation, etc.
- Image processing using HPC
- Raw data often re-processed (new algorithms, new software versions)
- Additionally: workflows, metadata, sharing with external partners



## Part: Human Brain Project

# Human Brain Project

- FET Flagship
- ~10 years, ~1 Billion EURO
- Coordinated by EPFL (Lausanne)
- Huge, multidisciplinary consortium
  - Neuroscience, medicine, physics, IT, philosophy
- <https://www.humanbrainproject.eu>



# HBP

## Platforms

- The *Neuroinformatics* Platform
- The *Brain Simulation* Platform
- The *High Performance Analytics and Computing* Platform
- The *Medical Informatics* Platform
- The *Neuromorphic Computing* Platform
- The *Neurorobotics* Platform

# System Access

## User questions

Typical SSH access scenario, multiple machines and sites

- What's my login/password on machine X?
- What LRMS is running on machine X and what are its commands?
- How do I set cores, nodes, memory, etc.?
- Where is my application located?
- Where is my workgroup's data or mine located?

⇒ Particularly difficult for domain scientists

# System Access

## Integration

How can I

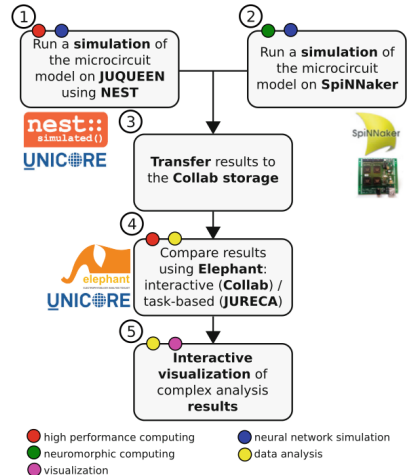
- ... use multiple, heterogeneous systems seamlessly and securely?
- ... manage my job input data and results?
- ... do the above across system boundaries?
- ... integrate HPC and data resources into applications/portals?

# NEST

## The Neural Simulation Technology Initiative

### NEST Simulator

- simulator for spiking neural network models
- focus on the dynamics, size and structure of neural systems



Source: Senk et. al (2017)

## Running nest Without UNICORE

- Login via SSH to JUQUEEN
- Manage working directory, code, and input parameters
- Create and submit a LoadLeveler script

### Example

```
#job_name = slns_demo
#...
#@bg_size = 32
#@wall_clock_limit = 00:10:00

module load python3/3.4.2
export TMPDIR=$WORK/tmp
export PYTHONPATH=/homeb/slns/slns007/local/opt/...

runjob --ranks-per-node 1 --exp-env ... : /bgsys/.../python3 microcircuit.py
```

## Running nest Using UNICORE

- Complexity is now hidden by UNICORE
- User can use a UNICORE Application “NEST”
- User only needs to invoke application and provide relevant data

### Example

```
ApplicationName: NEST,  
  
Parameters: [  
  NESTCODE: microcircuit.py, PARAMETERS: parameters.py, ],  
  
Imports: [ ... ],  
  
Resources: { Nodes: 32, Runtime: 1200 }
```

## Running nest Using UNICORE

Admin defines UNICORE Application “NEST” for JUQUEEN

### Example

```
<idb:IDBApplication>

  <idb:ApplicationName>NEST</idb:ApplicationName>

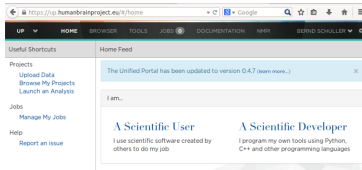
  <jsdl:POSIXApplication>
    <jsdl:Executable>runjob --ranks-per-node 1 --exp-env ... : .../python3</jsdl:Executable>
    <jsdl:Argument Type="filename">$NESTCODE?</jsdl:Argument>
    <jsdl:Argument Type="filename"># $PARAMETERS?</jsdl:Argument>
  </jsdl:POSIXApplication>

  <idb:PreCommand>#@environment = COPY_ALL</idb:PreCommand>
  <idb:PreCommand>module load python3/3.4.2</idb:PreCommand>
  <idb:PreCommand>export TMPDIR=$WORK/tmp</idb:PreCommand>
  <idb:PreCommand>export PYTHONPATH=/usr/local/...:$PYTHONPATH</idb:PreCommand>

  <idb:PostCommand>find -name *gdf | xargs zip output.zip</idb:PostCommand>

</idb:IDBApplication>
```

# HBP's HPC Platform



Unified Portal

2. access REST APIs  
pass OIDC Bearer token

REST API

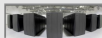
**UNICORE**

BSC



HPC site

CINECA



HPC site

CSCS



HPC site

JSC

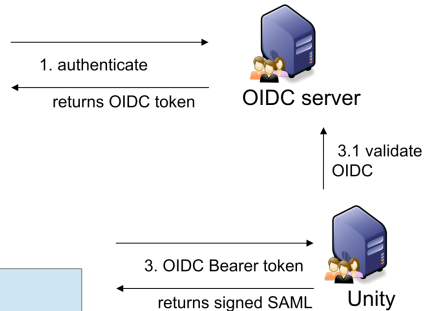


HPC site

KIT



S3 storage



## Single Sign-On

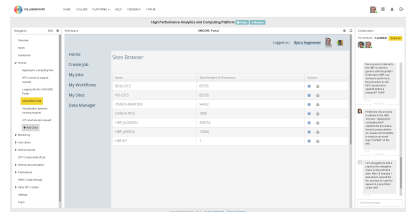
### Unity

- Bridges UNICORE to HBP OIDC infrastructure
- Supports REST and Web clients
- Support for SOAP/WS clients (legacy)

### User Management

- Users are granted resources (→ review process)
- User IDs and groups are mirrored to HPC sites (LDAP)
- Access via UNICORE is configured automatically

*The HBP Collaboratory collects tools from the HBP Platforms in one place and allows you to organize them into your own collaborative workspace or collab.*



# The Collaboratory Applications

- Applications can be added to *collabs*
- One such application is the UNICORE Storage Browser

Navigation

ADD

Overview

Team

Storage

Settings

Task 2.2

My Profile

HPC Storage JURECA

HPC Storage KIT

Job Manager

Workspace

bp000027-Home

HPC-JURECA

bp000027-Home

| Name                                  | Content | Created on | Created by |
|---------------------------------------|---------|------------|------------|
| .cache                                |         |            |            |
| .config                               |         |            |            |
| .ssh                                  |         |            |            |
| 2012-06-04_Private_Cloud.pdf.metadata | -       | -          | -          |
| .bashrc                               | -       | -          | -          |
| .kshrc                                | -       | -          | -          |
| .news_time                            | -       | -          | -          |
| .profile                              | -       | -          | -          |
| .sh_history                           | -       | -          | -          |
| .Xauthority                           | -       | -          | -          |

Collaboration

1 member - 1 available



This is the very beginning of the current collab's chat. Start using it to collaborate with your team!

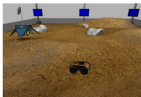
## Part: Neurorobotics

# Current Goals

## Neurorobotics

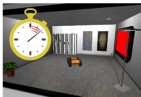
Workspace

Example experiments



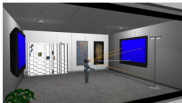
### Husky Braitenberg experiment in the SpaceBotCup 2013 arena

This experiment loads the Husky robot from Clearpath Robotics and the arena from the SpaceBotCup 2013. If the user starts the experiment, the Braitenberg vehicle network is executed and the robot will...



### Husky Braitenberg experiment with automatically switching screens

This experiment is similar to the Husky Braitenberg one (Husky robot detecting red colour and driving towards it). In this experiment the right screen is turned red automatically after...



### iCub Visual Tracking experiment

In this experiment the iCub robot performs a Visual Tracking task in the virtual room environment.

Timeout: 00 00:14:00

Backend availability: 13 / 14

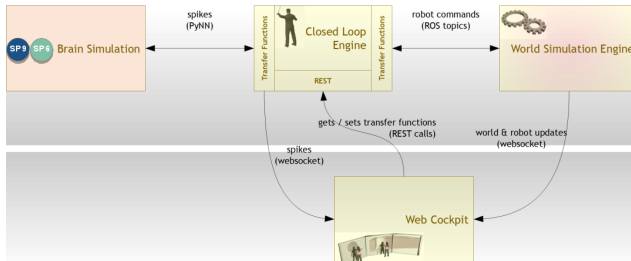
Cluster availability: 6 / 34

+ Launch

Upload custom environment

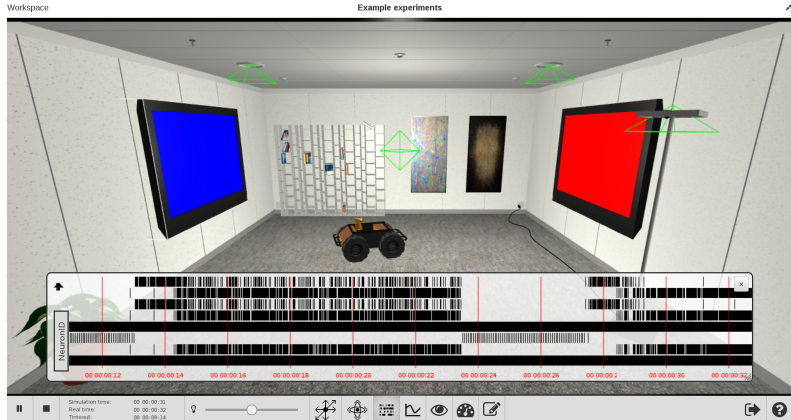
## Screenshot of HBP SP10 NRP Example Experiments

# NRP Architecture



Axel von Arnim, HBP SP10

# NRP Web Cockpit



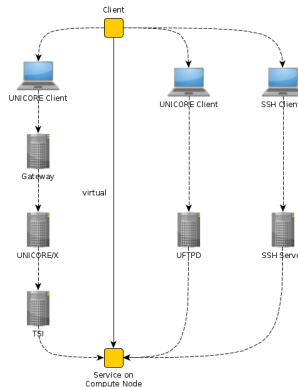
Screenshot of HBP SP10 NRP Husky Braitenberg Experiment

## Requirements

- Network tunnel into HPC resources
- Forwarding VNC connections
- Low latency
- Bandwidth?

# Architecture Options

- Many hops
- Higher latency expected
- Implementation difficult



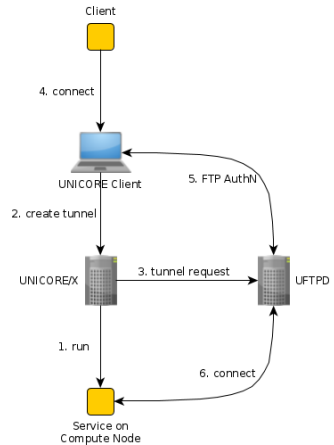
- Fewer hops
- Proven components
- Easier implementation

Own

# Implementation

## Tunnel Setup

- Same flow as for *ordinary* UFTP connections
- Firewall friendly



## Tunnel Setup

### UCC

- New create-tunnel command
- Parameter -L resembling SSH syntax
- Source Address can be set via -i
  - Important for NATing or multiple interfaces
- Example:

```
$ ucc run iperf.u -a  
720208ac-7015-49c3-b0c1-9e047b4287be.job  
$ ucc create-tunnel 720208ac-*.job -L 5001:localhost:5001  
Listening on localhost:5001
```

## Results

### Some (superficial) measurements

#### Throughput/Bandwidth

- Measured using iperf (submitted as job)
- No noticeable difference to native performance or SSH tunnels

#### Latency

- Usable for VNC, no noticeable lag

# Results

## IPerf

- 6 consecutive tests, 2 each using UFTP, SSH, and direct connections
- Server side log

```
-----  
Server listening on TCP port 5001  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 5] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 41246  
[ ID] Interval Transfer Bandwidth  
[ 5] 0.0-10.8 sec 120 MBytes 93.5 Mbits/sec  
[ 7] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 41252  
[ 7] 0.0-10.7 sec 119 MBytes 93.4 Mbits/sec  
[ 5] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 41253  
[ 5] 0.0-11.0 sec 119 MBytes 91.3 Mbits/sec  
[ 7] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 41254  
[ 7] 0.0-11.0 sec 123 MBytes 93.5 Mbits/sec  
[ 5] local 192.168.16.142 port 5001 connected with 134.94.168.32 port 52196  
[ 5] 0.0-10.1 sec 112 MBytes 93.6 Mbits/sec  
[ 7] local 192.168.16.142 port 5001 connected with 134.94.168.32 port 52200  
[ 7] 0.0-10.1 sec 112 MBytes 93.6 Mbits/sec
```

## Open Issues

### The small print

- Need to deploy full NRP scenario including VirtualGL over VNC
- Currently only a single connection possible after tunnel setup
- Tunnel tear-down needs to be implemented
- It is not as straightforward to implement SSH tunnels as it may seem
- More measurements under various conditions
  - Better abstraction is needed

## Summary

- Neuroscience involvement
  - SMHB
  - HBP
- Variety of services for supporting neuroscience
- New use cases lead to new features