



**DISCLAIMER: More detailed information on the results and/or performance obtained and their use is available in the Subproject's subsequent Deliverable (D7.7.5) and/or Periodic Report.**

Project Number:	604102	Project Title:	Human Brain Project
Document Title:	High Performance Computing Platform v1 - preliminary release for internal Consortium use (prototype)		
Document Filename:	SP7 D7.7.4 FINAL		
Deliverable Number:	D7.7.4		
Deliverable Type:	Prototype		
Work Package(s):	WPs 7.2 to 7.7		
Dissemination Level:	PU = Public		
Planned Delivery Date:	Month 19 / 31 May 2015		
Actual Delivery Date:	Month 21 / 24 Jun 2015		
SP Leader(s):	Thomas LIPPERT, JUELICH (P17), Germany (WP7.1, WP7.6, WP7.7) Thomas SCHULTHESS, ETHZ (P15), Switzerland (WP7.5)		
SP Manager(s):	Boris ORTH, Anna LÜHRS, JUELICH (P17), Germany		
Authors:	Boris ORTH, Anna LÜHRS, JUELICH (P17), Germany		
Contributors:	Javier BARTOLOME, BSC (P4) Stefan EILEMANN, EPFL (P1) Minos GAROFALAKIS, TUC (P51) Diana GUDU, KIT (P30) Carsten KARBACH, JUELICH (P17) Martin KERSTEN, CWI (P48) Xuesong LU, EPFL (P1) Bernd MOHR, JUELICH (P17) Sergi MORE, BSC (P4) Roberto MUCCI, CINECA (P10) Bernd SCHULLER, JUELICH (P17) Raúl SIRVENT, BSC (P4) Benjamin WEYERS, RWTH (P42)		
STO Review:	Björn KINDLER, Martina SCHMALHOLZ, Sabine SCHNEIDER UHEI (P45), Germany		
Editorial Review:	Richard WALKER, Guy WILLIS, Lauren ORWIN EPFL (P1), Switzerland		
Abstract:	<p>This will be the first release of the platform, which will be available for use by members of the HBP Consortium.</p> <p>This document is dedicated to the users of the HPC Platform and explains how to access the hardware, software and services provided by SP7.</p> <p>The document is a living document and will updated when new components or new features of existing pieces of the HPC Platform are released.</p>		
Keywords:	HPC Platform Release		



## Table of Contents

1. The Aim of this Document .....	4
2. How to Access the HBP High Performance Computing Platform .....	4
2.1 Immediate Access:.....	4
2.2 Access via the HBP Collaboratory: .....	5
3. Platform User Instructions.....	5
4. Help and User Feedback .....	6
5. Annexes .....	6
<b>Annex 1: Tools &amp; Features Available in this Platform Release.....</b>	<b>7</b>
Software applications.....	7
Programming models, libraries and toolkits .....	10
Other software.....	12
<b>Annex 2: Descriptions of Tools &amp; Features available in this HPC Platform Release .....</b>	<b>13</b>
Software applications.....	13
Cube: Score-P / Scalasca.....	13
DisplayCluster .....	15
Extrac 16	
FLAT 16	
Livre 17	
Paraver 18	
Remote Connection Manager (RCM) .....	19
RTNeuron .....	20
Scalasca: HPC Performance Trace Analyzer .....	21
Score-P: HPC Performance Instrumentation and Measurement Tool.....	22
SCOUT 22	
Programming models, libraries and toolkits .....	23
Deflect client library.....	23
Equalizer .....	23
OmpSs 24	
PyCOMPSs.....	24
ViSTA Virtual Reality Toolkit .....	24
ZeroEQ 25	
Other software.....	26
MonetDB .....	26
T-Storm 26	
<b>Annex 3: Infrastructure Components of the HPC Platform .....</b>	<b>27</b>
<b>Annex 4: Access to the supercomputers of the HPC Platform .....</b>	<b>28</b>



## List of Figures

Figure 1 Screenshot of the UNICORE Portal as an interim access point to the HPC Platform 4

Figure 2 Screenshot of the HBP Collaboratory as an access point to the HPC Platform ..... 5

Figure 3 Screenshot of Cube ..... 14

Figure 4 Using DisplayCluster - Picture 1..... 15

Figure 5 Using DisplayCluster - Picture 2..... 15

Figure 6 Screenshot 1 of Livre ..... 17

Figure 7 Screenshot 2 of Livre ..... 17

Figure 8 Screenshot of Paraver ..... 18

Figure 9 Screenshot 1 of RCM: graphical interface ..... 19

Figure 10 Screenshot 2 of RCM..... 19

Figure 11 Using RTNeuron - Picture 1 ..... 20

Figure 12 Using RTNeuron - Picture 2..... 20

Figure 13 Equalizer on a display wall ..... 23

Figure 14 Screenshot of PyCOMPSs ..... 24

## List of Tables

Table 1 Software applications available in the HPC Platform ..... 9

Table 2 Programming models, libraries and toolkits availale in the HPC Platform..... 11

Table 3 Other software available in the HPC Platform ..... 12



## 1. The Aim of this Document

This document provides the information needed to access the High Performance Computing Platform v1 - preliminary release for internal Consortium use (Deliverable D7.7.4, Milestone 147). It explains how to request a login, how to access the HPC Platform and whom to contact for support and feedback. The document introduces the Platform, its hardware and software components, its services and its goals. The first two annexes list the software applications, programming models, libraries and tools available for the users of the HPC Platform and how to access them. The third annex describes the integration of the infrastructure components into the Platform and it is dedicated to the developers of the HPC Platform. Annex 4 describes the mechanisms to get access to the supercomputers integrated in the HPC Platform.

## 2. How to Access the HBP High Performance Computing Platform

### 2.1 Immediate Access:

Until the Collaboratory (see below) is fully operational, the supercomputing resources of the HPC Platform can be accessed through the UNICORE Portal:

<https://collaboration.humanbrainproject.eu/web/collab/wiki/-/wiki/Main/UNICORE-Portal>

The screenshot shows the UNICORE Portal interface. The browser address bar displays `https://hbp-portal.fz-juelich.de:9091/portal#!/Sites`. The page title is "UNICORE" and the user is logged in as "Bernd Schuller". A sidebar on the left contains navigation links: Home, New Job, My Jobs, My Sites, Data Manager, and Grid Browser. The main content area is titled "Sites Browser" and contains a table with the following data:

NAME	STATE	TOTAL NUMBER OF PROCESSORS	ACTIONS
HBP	Ready		
BSC-MareNostrum	Access denied	48912	
CINECA-FERMI	Unknown, please refresh		
CINECA-GALILEO	Unknown, please refresh		
CINECA-PICO	Unknown, please refresh		
HBP-KIT	Access denied	1	
HBP_JUQUEEN	Ready	458752	
VIS-CSCS	Ready	65535	

Figure 1 Screenshot of the UNICORE Portal as an interim access point to the HPC Platform

The current v1 Platforms are a preliminary release for internal Consortium use. If you do not have the necessary login and password, please contact the HPC Platform Management Team:



[hbp-hpc-platform@fz-juelich.de](mailto:hbp-hpc-platform@fz-juelich.de)

Please note that applications from non-HBP staff may take a couple of days to process.

## 2.2 Access via the HBP Collaboratory:

All HBP Platforms will be accessible via the HBP Collaboratory (formerly Unified Portal). Collaboratory:

<https://collab.humanbrainproject.eu/%23/collab/19/nav/403>

HPC Platform on the Collaboratory:

<https://collab.humanbrainproject.eu/#/collab/49/nav/240>

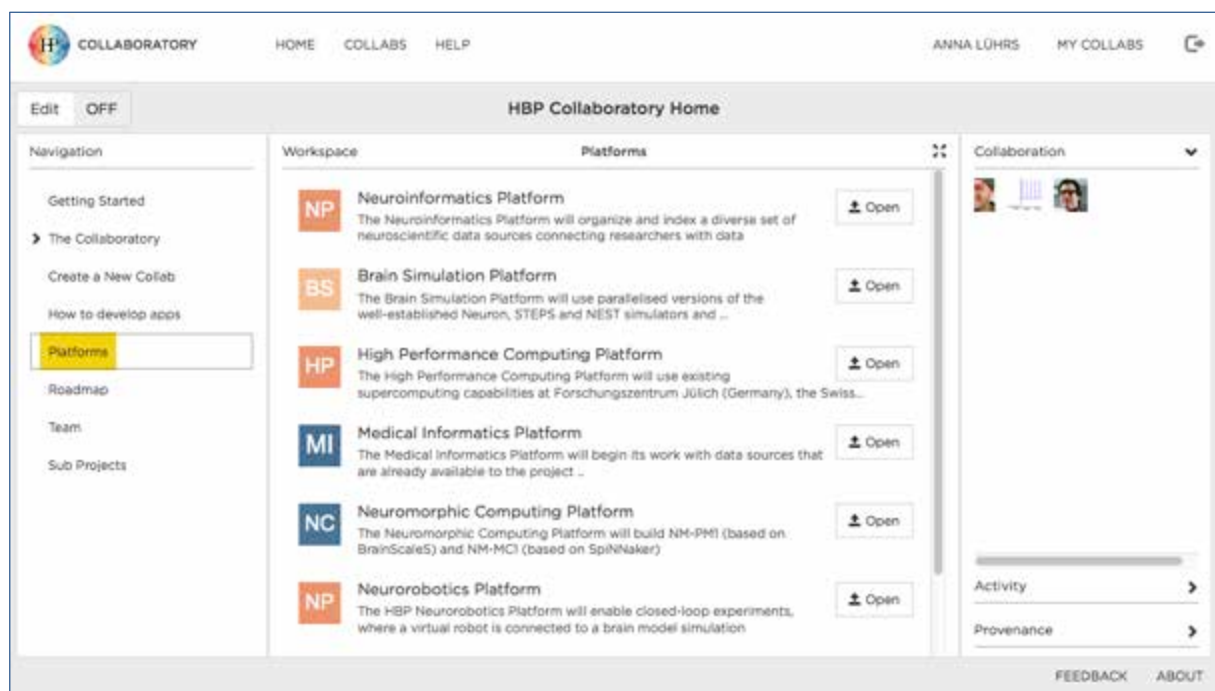


Figure 2 Screenshot of the HBP Collaboratory as an access point to the HPC Platform

## 3. Platform User Instructions

All versions of this document are available in the HBP Collaboration Portal in the section *Documents / SP7 - High Performance Computing / HPC Platform Release / D7.7.4 Deliverable*:

[https://collaboration.humanbrainproject.eu/web/collab/documents?p\\_p\\_id=20&p\\_p\\_lifecycle=0&p\\_p\\_state=normal&p\\_p\\_mode=view&p\\_p\\_col\\_id=column-1&p\\_p\\_col\\_pos=1&p\\_p\\_col\\_count=2&\\_20\\_struts\\_action=%2Fdocument\\_library%2Fview&\\_20\\_folderId=531000&\\_20\\_viewEntries=1&\\_20\\_viewFolders=1&\\_20\\_action=browseFolder&\\_20\\_entryStart=0&\\_20\\_entryEnd=50&\\_20\\_folderStart=0&\\_20\\_folderEnd=20](https://collaboration.humanbrainproject.eu/web/collab/documents?p_p_id=20&p_p_lifecycle=0&p_p_state=normal&p_p_mode=view&p_p_col_id=column-1&p_p_col_pos=1&p_p_col_count=2&_20_struts_action=%2Fdocument_library%2Fview&_20_folderId=531000&_20_viewEntries=1&_20_viewFolders=1&_20_action=browseFolder&_20_entryStart=0&_20_entryEnd=50&_20_folderStart=0&_20_folderEnd=20)

The HPC Platform Guidebook, a collection of all user documentation, can also be found there under *Documents / SP7 - High Performance Computing / HPC Platform Release / HPC Platform Guidebook*:

[https://collaboration.humanbrainproject.eu/web/collab/documents?p\\_p\\_id=20&p\\_p\\_lifecycle=0&p\\_p\\_state=normal&p\\_p\\_mode=view&p\\_p\\_col\\_id=column-1&p\\_p\\_col\\_pos=1&p\\_p\\_col\\_count=2&\\_20\\_struts\\_action=%2Fdocument\\_library%2Fview&\\_20](https://collaboration.humanbrainproject.eu/web/collab/documents?p_p_id=20&p_p_lifecycle=0&p_p_state=normal&p_p_mode=view&p_p_col_id=column-1&p_p_col_pos=1&p_p_col_count=2&_20_struts_action=%2Fdocument_library%2Fview&_20)



[\\_folderId=531003&\\_20\\_viewEntries=1&\\_20\\_viewFolders=1&\\_20\\_action=browseFolder&\\_20\\_entryStart=0&\\_20\\_entryEnd=50&\\_20\\_folderStart=0&\\_20\\_folderEnd=20](#)

## 4. Help and User Feedback

For help or to provide feedback, please contact the HPC Platform Management Team:

[HBP-HPC-platform@fz-juelich.de](mailto:HBP-HPC-platform@fz-juelich.de)

## 5. Annexes

1. Tools & Features Available in this Platform Release
2. Descriptions of Tools & Features available in this HPC Platform Release
3. Infrastructure Components of the HPC Platform
4. Access to the supercomputers of the HPC Platform



## Annex 1: Tools & Features Available in this Platform Release

The intended user groups for the software applications, programming models, libraries toolkits developed by and deployed in the HPC Platform were defined in [Deliverable D7.7.2 High Performance Computing Platform v1 - specification document](#):

- [Computational Scientific User \(CSU\) - A user with scientific development skills and comfort in launching command line HPC jobs.](#)
- [Biological Scientific User \(BSU\) - A user with scientific expertise, but limited technical computing skills.](#)
- [Scientific User \(SU\) - A scientific user, either a CSU or a BSU.](#)
- [Scientific Developer \(SCIDEV\) - A user who is developing software to directly realise the scientific objectives. This user is usually working in close collaboration with scientists, both CSUs and BSUs.](#)
- [Developer \(DEV\) - A user who is developing software to realise engineering, operational and/or scientific objectives.](#)
- [Portal User \(PU\) - A user who accesses Platform functions through the Web GUI, the Collaboratory.](#)
- [Service User \(ServU\) - A user who accesses Platform functions through a programmatic Service Client API.](#)
- [Infrastructure Personnel \(INFRA\) - An infrastructure system administrator or developer, typically responsible for deploying and monitoring Platform services that are offered directly to customers.](#)

### Software applications

Tool / Feature	Responsible	Milestone / Remarks	Intended User Groups	Keywords	Available at	Description Screenshot
<b>Cube: Score-P / Scalasca</b>	WP 7.2, T 7.2.4 Scalasca team: <a href="mailto:scalasca@fz-juelich.de">scalasca@fz-juelich.de</a>	MS137 (M18): Multiscale supercomputing: algorithms and models	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	Parallel Application, Performance Analysis	<a href="http://www.scalasca.org/software/cube-4.x/download.html">http://www.scalasca.org/software/cube-4.x/download.html</a>	Description Screenshot
<b>DisplayCluster</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a>	MS139 (M12): Software for interactive visualization MS141 (M30): Interactive visualization framework	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU), Service User (ServU),	Visualization	<a href="https://github.com/BlueBrain/DisplayCluster">https://github.com/BlueBrain/DisplayCluster</a>	Description Screenshots





			Infrastructure Personnel (INFRA)			
<b>Extrae</b>	WP 7.2, T 7.2.4 BSC Performance Tools Group: <a href="mailto:tools@bsc.es">tools@bsc.es</a>	MS135 (M6): Specifications and roadmap MS136 (M12): Interactive supercomputing: algorithms & models	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	High Performance Computing, Instrumentation and Measurement	<a href="http://www.bsc.es/computer-sciences/performance-tools/downloads">http://www.bsc.es/computer-sciences/performance-tools/downloads</a>	Description
<b>FLAT</b>	WP 7.4, T 7.4.1 EPFL-DIAS, Xuesong Lu ( <a href="mailto:xuesong.lu@epfl.ch">xuesong.lu@epfl.ch</a> ), Darius Sidlauskas ( <a href="mailto:darius.sidlauskas@epfl.ch">darius.sidlauskas@epfl.ch</a> )	MS143 (M24): Data analytics prototype	Scientific User (SU), Portal User (PU)	Data management, query and analysis	Collaboratory, integrated and part of BBP SDK tool set	Description
<b>Livre</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann ( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS139 (M12): Software for interactive visualization	Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV)	Visualization	<a href="https://github.com/BlueBrain/Livre">https://github.com/BlueBrain/Livre</a>	Description Screenshots
<b>Paraver</b>	WP 7.2, T 7.2.4 BSC Performance Tools Group, <a href="mailto:tools@bsc.es">tools@bsc.es</a>	MS135 (M6): Specifications and roadmap MS136 (M12): Interactive supercomputing: algorithms & models	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	High Performance Computing, Performance Analysis and Performance Visualization	<a href="http://www.bsc.es/computer-sciences/performance-tools/downloads">http://www.bsc.es/computer-sciences/performance-tools/downloads</a>	Description Screenshot
<b>Remote Connection Manager (RCM)</b>	WP 7.4, T 7.4.4 Roberto Mucci ( <a href="mailto:superc@cineca.it">superc@cineca.it</a> )	MS147 (M18): Platform ready for internal release	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU)	Remote visualization, Scientific Visualization, Data management, query and analysis, High Performance Computing	<a href="http://www.hpc.cineca.it/content/remote-visualization-rcm">http://www.hpc.cineca.it/content/remote-visualization-rcm</a>	Description Screenshots
<b>RTNeuron</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann ( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS139 (M12): Software for interactive visualization MS140 (M18):	Computational Scientific User (CSU), Biological Scientific User (BSU),	Visualization	<a href="https://bbpcode.epfl.ch/code/#/admin/projects/viz/RTNeuron">https://bbpcode.epfl.ch/code/#/admin/projects/viz/RTNeuron</a>	Description Screenshots





		Neuroscience visualisation methods	Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU)			
<b>Scalasca: HPC Performance Trace Analyzer</b>	WP 7.2, T 7.2.4 Scalasca team: <a href="mailto:scalasca@fz-juelich.de">scalasca@fz-juelich.de</a>	MS137 (M18): Multiscale supercomputing: algorithms and models	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	Parallel Application Performance Analysis	<a href="http://www.scalasca.org/software/scalasca-2.x/download.html">http://www.scalasca.org/software/scalasca-2.x/download.html</a>	Description
<b>Score-P: HPC Performance Instrumentation and Measurement Tool</b>	WP 7.2, T 7.2.4 Score-P consortium: <a href="mailto:support@score-p.org">support@score-p.org</a>	MS137 (M18): Multiscale supercomputing: algorithms and models	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	Parallel Application Performance Analysis	<a href="http://www.score-p.org">http://www.score-p.org</a> , Section “Download section”	Description
<b>SCOUT</b>	WP 7.4, T 7.4.1 EPFL-DIAS, Xuesong Lu ( <a href="mailto:xuesong.lu@epfl.ch">xuesong.lu@epfl.ch</a> ), Darius Sidlauskas ( <a href="mailto:darius.sidlauskas@epfl.ch">darius.sidlauskas@epfl.ch</a> )	MS143 (M24): Data analytics prototype	Scientific User (SU), Portal User (PU)	Data management, query and analysis	Collaboratory, integrated and part of BBP SDK tool set	Description

Table 1 Software applications available in the HPC Platform



## Programming models, libraries and toolkits

Tool / Feature	Responsible	Milestone / Remarks	Intended User Groups	Keywords	Available at	Description Screenshot
<b>Deflect client library</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann ( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS139 (M12): Software for interactive visualization MS141 (M30): Interactive visualization framework	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU), Service User (ServU), Infrastructure Personnel (INFRA)	Visualization	<a href="https://github.com/BlueBrain/Deflect">https://github.com/BlueBrain/Deflect</a>	Description Screenshot
<b>Equalizer</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann ( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS141 (M30): Interactive visualization framework	Scientific Developer (SCIDEV), Developer (DEV)	Visualization	<a href="https://github.com/Eyescale/Equalizer">https://github.com/Eyescale/Equalizer</a>	Description Screenshot
<b>OmpSs</b>	WP 7.2, T 7.2.4 BSC Programming Models Group: <a href="mailto:pm-tools@bsc.es">pm-tools@bsc.es</a>	MS135 (M6): Specifications and roadmap MS136 (M12): Interactive supercomputing: algorithms & models	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV)	High Performance Computing	<a href="https://pm.bsc.es/ompps-downloads">https://pm.bsc.es/ompps-downloads</a>	Description Screenshot
<b>PyCOMPSs</b>	WP 7.2, T 7.2.4 BSC Workflows and Distributed Computing Group: <a href="mailto:support-compss@bsc.es">support-compss@bsc.es</a>	MS135 (M6): Specifications and roadmap MS136 (M12): Interactive supercomputing: algorithms & models	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU), Service User (ServU)	High Performance Computing	<a href="http://compss.bsc.es">http://compss.bsc.es</a>	Description Screenshot
<b>RTNeuron</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann	MS139 (M12): Software for interactive visualization	Computational Scientific User (CSU), Biological Scientific User	Visualization	<a href="https://bbpcode.epfl.ch/code/#/admin/projects/viz/RTNeur">https://bbpcode.epfl.ch/code/#/admin/projects/viz/RTNeur</a>	Description Screenshot



	( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS140 (M18): Neuroscience visualisation methods	(BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU)		<a href="#">on</a>	
<b>ViSTA Virtual Reality Toolkit</b>	WP 7.3, T 7.3.4 RWTH, Torsten Kuhlen ( <a href="mailto:kuhlen@vr.rwth-aachen.de">kuhlen@vr.rwth-aachen.de</a> ), Benjamin Weyers ( <a href="mailto:weyers@vr.rwth-aachen.de">weyers@vr.rwth-aachen.de</a> )	MS139 (M12): Software for interactive visualization	Developer (DEV)	Visualization Immersive Environments	<a href="http://sourceforge.net/projects/vistavr-toolkit/">http://sourceforge.net/projects/vistavr-toolkit/</a> , <a href="https://github.com/HBPVIS/Vista">https://github.com/HBPVIS/Vista</a>	Description
<b>ZeroEQ</b>	WP 7.3, T 7.3.1 EPFL, Stefan Eilemann ( <a href="mailto:stefan.eilemann@epfl.ch">stefan.eilemann@epfl.ch</a> )	MS141 (M30): Interactive visualization framework	Scientific Developer (SCIDEV), Developer (DEV)	High Performance Computing, Visualization	<a href="https://github.com/HBPVis/zeq">https://github.com/HBPVis/zeq</a>	Description

Table 2 Programming models, libraries and toolkits available in the HPC Platform



## Other software

Tool / Feature	Responsible	Milestone / Remarks	Intended User Groups	Keywords	Available at	Description Screenshot
MonetDB	WP 7.4, T 7.4.4 CWI	MS143 (M24): Data analytics prototype MS144 (M30): Query tools	Computational Scientific User (CSU), Biological Scientific User (BSU), Scientific User (SU), Scientific Developer (SCIDEV), Developer (DEV), Portal User (PU), Service User (ServU)	Data management, query and analysis	<a href="http://www.monetdb.org">http://www.monetdb.org</a>	Description
T-Storm	WP 7.4, T 7.4.2 Minos Garofalakis ( <a href="mailto:minos@acm.org">minos@acm.org</a> )	MS143 (M24): Data analytics prototype	Computational Scientific User (CSU), Scientific Developer (SCIDEV), Developer (DEV)	Data management, query and analysis, Time-series analytics	Collaboratory	Description

Table 3 Other software available in the HPC Platform



## Annex 2: Descriptions of Tools & Features available in this HPC Platform Release

### Software applications

#### *Cube: Score-P / Scalasca*

##### Description

Cube, which is used as performance report explorer for Scalasca and Score-P, is a generic tool for displaying a multi-dimensional performance space consisting of the dimensions

- (i) Performance metric,
- (ii) Call path, and
- (iii) System resource.

Each dimension can be represented as a tree, where non-leaf nodes of the tree can be collapsed or expanded to achieve the desired level of granularity. In addition, Cube can display multi-dimensional Cartesian process topologies.

The Cube 4.x series report explorer and the associated Cube4 data format is provided for Cube files produced with the Score-P performance instrumentation and measurement infrastructure or with Scalasca version 2.x trace analyzer (and other compatible tools). However, for backwards compatibility, Cube 4.x can also read and display Cube 3.x data.

Cube is part of a larger set of tools for parallel performance analysis and debugging developed by the “Virtual Institute - High Productivity Supercomputing” consortium. Further documentation, training and support are available through VI-HPS:

- High-level Tool Descriptions: [VI-HPS Tools Guide](#) and <http://www.vi-hps.org/tools/>
- Index to VI-HPS tool documentation: <http://www.vi-hps.org/training/doc/>
- For training classes: [VI-HPS Tuning Workshops](#), course material available at <http://www.vi-hps.org/training/material/>



## Screenshot

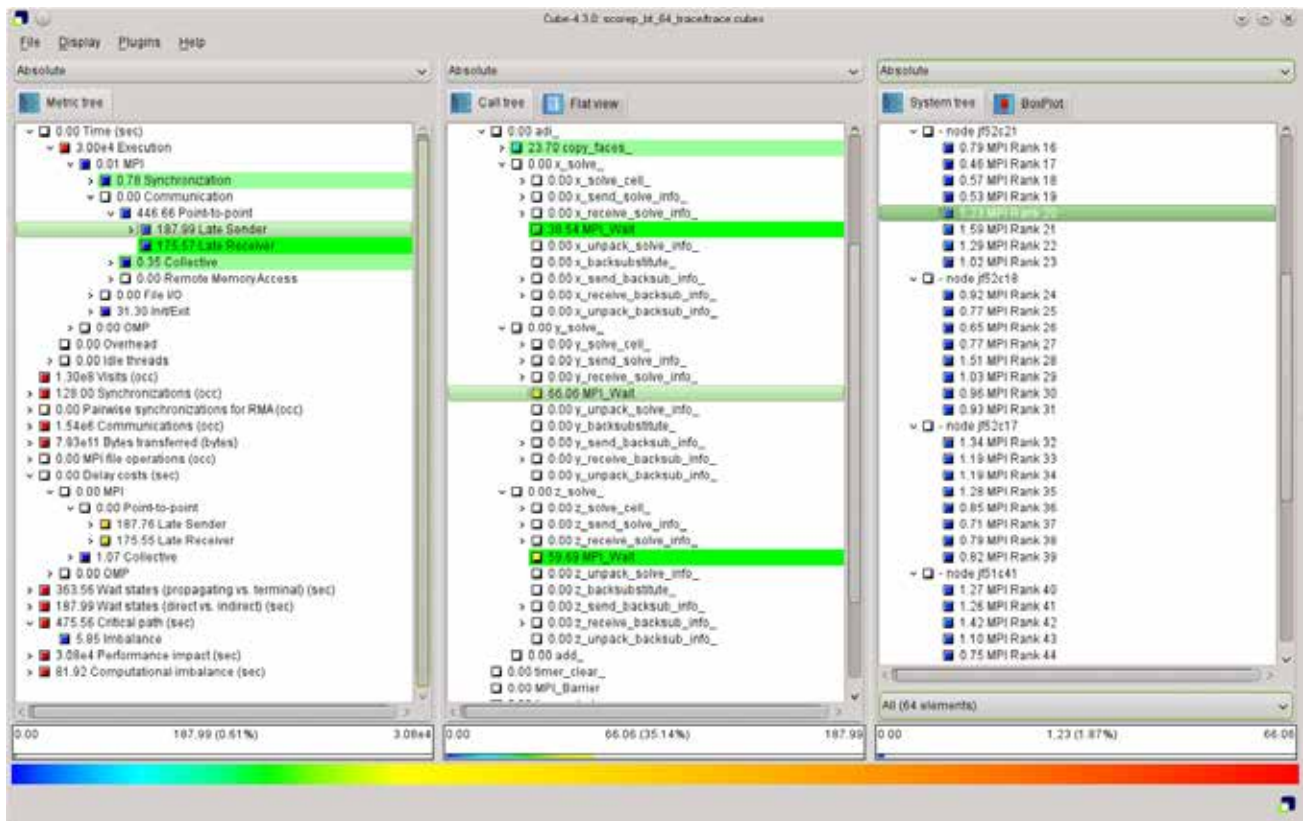


Figure 3 Screenshot of Cube



## DisplayCluster

### Description

DisplayCluster is a software environment for interactively driving large-scale tiled displays. It provides the following functionality:

- View media interactively such as high-resolution imagery, PDFs and video.
- Receive content from remote sources such as laptops, desktops or parallel remote visualization machines using the [Deflect library](#).

### Screenshots

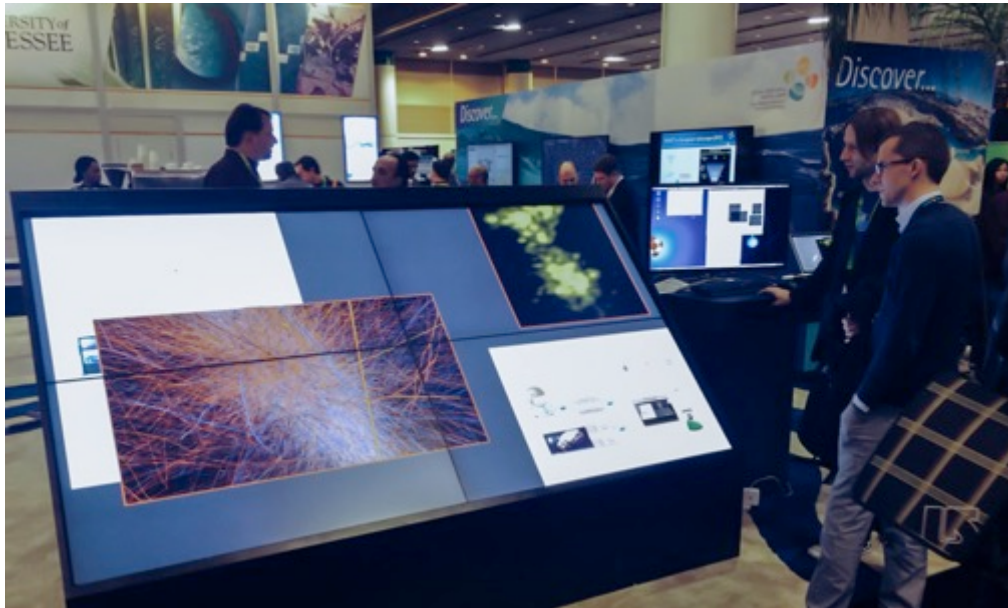


Figure 4 Using DisplayCluster - Picture 1



Figure 5 Using DisplayCluster - Picture 2





## ***Extræe***

### **Description**

Extræe is an instrumentation and measurement system gathering time stamped information of the events of an application. It is the package devoted to generate Paraver trace files for a post-mortem analysis of a code run. It uses different interposition mechanisms to inject probes into the target application in order to gather information about the application performance.

## ***FLAT***

### **Description**

FLAT is a spatial indexing tool, which enables scalable range queries on (3D) spatial datasets. Given the user input, which should be a query range, and the dataset to be queried, FLAT returns all the objects that intersect with the query range.

In particular, both the query ranges and the spatial objects should be represented using minimum bounding rectangle (MBR), which is the geometry approximation bounding the underlying spatial object.

FLAT outperforms the state-of-the-art spatial indexing techniques (e.g., R-trees, grid file) on extremely dense datasets.



## *Livre*

### Description

Livre is an out-of-core rendering engine that has the following features:

- Distributed rendering using Equalizer parallel rendering framework
- Octree based out-of-core rendering.
- Visualisation of pre-processed UVF format volume data sets.
- Real-time voxelisation and visualisation of surface meshes using OpenGL 4.2 extensions.
- Real-time voxelisation and visualisation of Blue Brain Project (BBP) morphologies.
- Real-time voxelisation and visualisation of local-field potentials in BBP circuit.
- Multi-node, multi-GPU rendering.

### Screenshots



Figure 6 Screenshot 1 of Livre

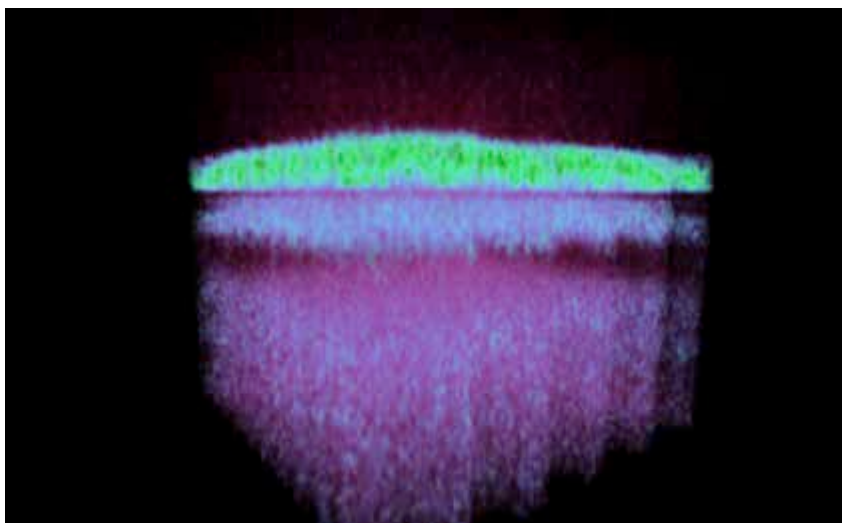


Figure 7 Screenshot 2 of Livre



## Paraver

### Description

Paraver is a very flexible data browser. The metrics used are not hardwired on the tool but can be programmed. To compute them, the tool offers a large set of time functions, a filter module, and a mechanism to combine two timelines. This approach allows displaying a huge number of metrics with the available data. The analysis display allows computing statistics over any timeline and selected region, what allows correlating the information of up to three different time functions. To capture the expert's knowledge, any view or set of views can be saved as a Paraver configuration file. Therefore, re-computing the same view with new data is as simple as loading the saved file. The tool has been demonstrated to be very useful for performance analysis studies, giving much more details about the applications behaviour than most performance tools available.

### Screenshot

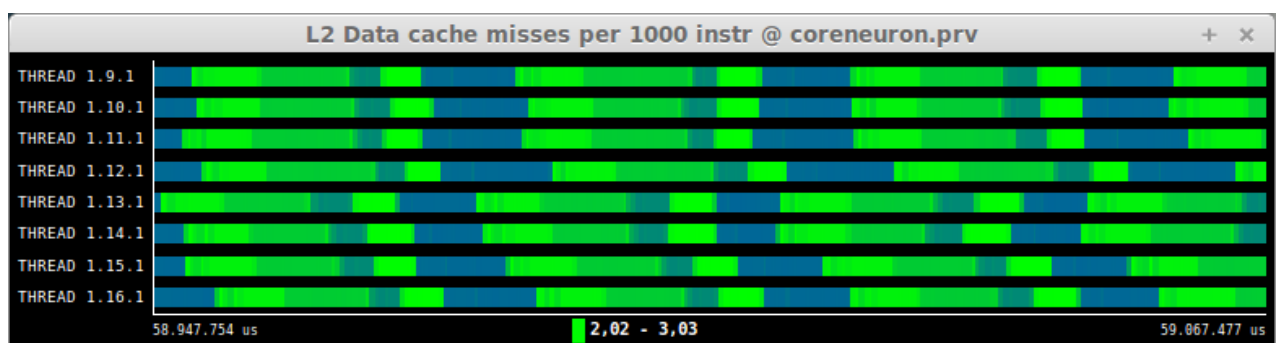


Figure 8 Screenshot of Paraver



## Remote Connection Manager (RCM)

### Description

The Remote Connection Manager (RCM) is an application that allows HPC users to perform remote visualisation on Cineca HPC clusters.

The tool offers to

- Visualize the data produced on Cineca's HPC systems (scientific visualization);
- Analyse and inspect data directly on the systems;
- Debug and profile parallel codes running on the HPC clusters.

The graphical interface of RCM allows the HPC users to easily create remote displays and to manage them (connect, kill, refresh).

### Screenshots



Figure 9 Screenshot 1 of RCM: graphical interface

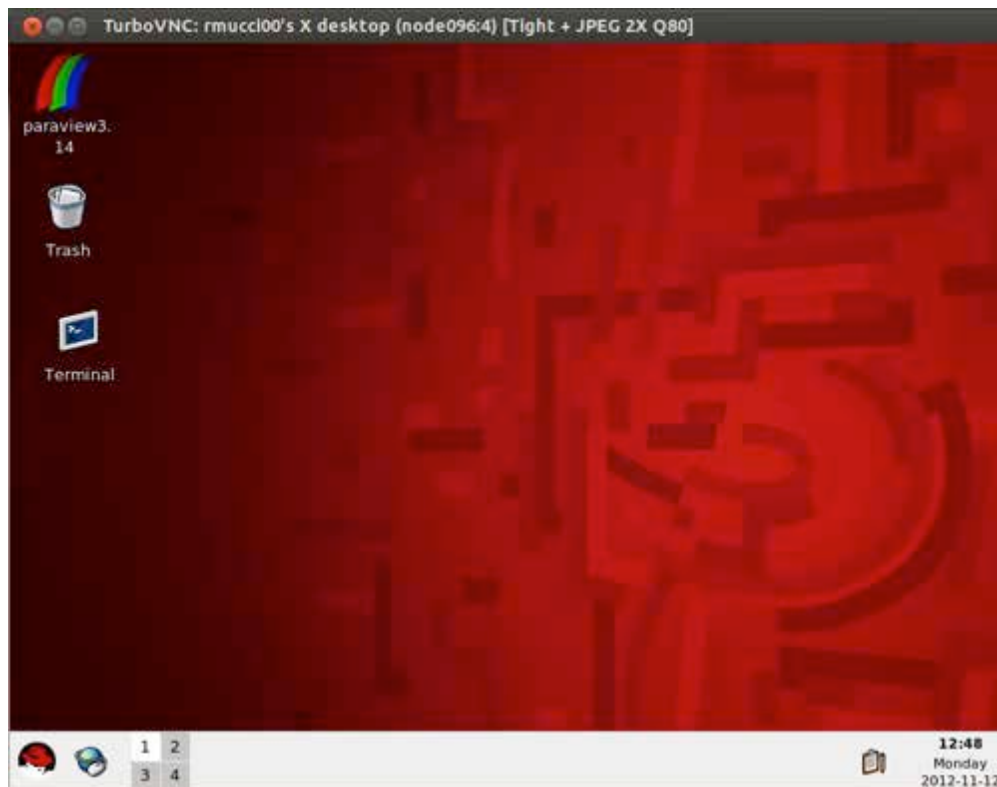


Figure 10 Screenshot 2 of RCM



## ***RTNeuron***

### **Description**

RTNeuron is a scalable real-time rendering tool for the visualisation of neuronal simulations based on cable models. Its main utility is twofold: the interactive visual inspection of structural and functional features of the cortical column model and the generation of high quality movies and images for presentations and publications. The package provides three main components:

- A high level C++ library.
- A Python module that wraps the C++ library and provides additional tools.
- The Python application script `rtneuron-app.py`

A wide variety of scenarios is covered by `rtneuron-app.py`. In case the user needs a finer control of the rendering, such as in movie production or to speed up the exploration of different data sets, the Python wrapping is the way to go. The Python wrapping can be used through an IPython shell started directly from `rtneuron-app.py` or importing the module `rtneuron` into own Python programs.

### **Screenshots**



Figure 11 Using RTNeuron - Picture 1

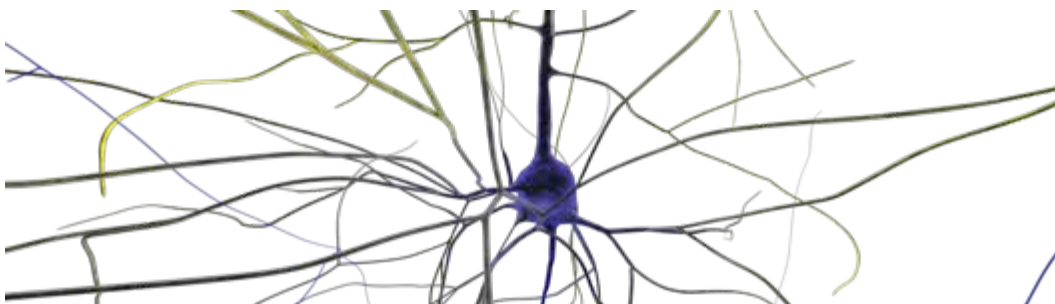


Figure 12 Using RTNeuron - Picture 2



## ***Scalasca: HPC Performance Trace Analyzer***

### **Description**

Scalasca performance analysis of large-scale parallel applications

Scalasca is a software tool that supports the performance optimisation of parallel programs by measuring and analysing their runtime behaviour. The analysis identifies potential performance bottlenecks - in particular those concerning communication and synchronization - and offers guidance in exploring their causes.



Scalasca targets mainly scientific and engineering applications based on the programming interfaces MPI and OpenMP, including hybrid applications based on a combination of the two. The tool has been specifically designed for use on large-scale systems including IBM Blue Gene and Cray XT, but is also well suited for small- and medium-scale HPC platforms. The software is available for free download under the New BSD open-source license.

Scalasca is part of a larger set of tools for parallel performance analysis and debugging developed by the “Virtual Institute - High Productivity Supercomputing” consortium. Further documentation, training and support are available through VI-HPS.

- High-level Tool Descriptions: [VI-HPS Tools Guide](#) and <http://www.vi-hps.org/tools/>
- Index to VI-HPS tool documentation: <http://www.vi-hps.org/training/doc/>
- For training classes: [VI-HPS Tuning Workshops](#), course material available at <http://www.vi-hps.org/training/material/>





## **Score-P: HPC Performance Instrumentation and Measurement Tool**

### **Description**

Scalable Performance Measurement Infrastructure for Parallel Codes

The Score-P measurement infrastructure is a highly scalable and easy-to-use tool suite for profiling, event tracing, and online analysis of HPC applications. Score-P is developed under a BSD 3-Clause (Open Source) License and governed by a meritocratic governance model.



Score-P offers the user a maximum of convenience by supporting a number of analysis tools. Currently, it works with Periscope, Scalasca, Vampir, and Tau and is open for other tools. Score-P comes together with the new Open Trace Format Version 2, the Cube4 profiling format and the Opari2 instrumenter.

Score-P is part of a larger set of tools for parallel performance analysis and debugging developed by the “Virtual Institute - High Productivity Supercomputing” consortium. Further documentation, training and support are available through VI-HPS.

- High-level Tool Descriptions: [VI-HPS Tools Guide](#) and <http://www.vi-hps.org/tools/>
- Index to VI-HPS tool documentation: <http://www.vi-hps.org/training/doc/>
- For training classes: [VI-HPS Tuning Workshops](#), course material available at <http://www.vi-hps.org/training/material/>

## **SCOUT**

### **Description**

SCOUT is a structure-aware method for prefetching data along interactive spatial query sequences. Given the user input, which is a spatial range query sequence representing the structure explored (interactively) by the user, and the spatial dataset to be queried, SCOUT reduces the query response time by prefetching the data along the query sequence.

Similarly to FLAT, both the query ranges in the query sequence and the spatial objects should be represented using a minimum bounding rectangle (MBR).

SCOUT outperforms the related prefetching techniques (e.g., Straight Line Extrapolation or Hilbert prefetching) with high prefetching accuracy, which is translated to one order of magnitude speedup.





## Programming models, libraries and toolkits

### *Deflect client library*

#### Description

Deflect is a C++ library to develop applications that can send and receive pixel streams from other Deflect-based applications, for example DisplayCluster. The following applications are provided which make use of the streaming API:

- DesktopStreamer: A small utility that allows the user to stream the desktop.
- SimpleStreamer: A simple example to demonstrate streaming of an OpenGL application.

### *Equalizer*

#### Description

Equalizer is a parallel rendering framework to create and deploy parallel, scalable OpenGL applications. It provides the following major features to facilitate the development and deployment of scalable OpenGL applications:



- Runtime Configurability: An Equalizer application is configured automatically or manually at runtime and can be deployed on laptops, multi-GPU workstations and large-scale visualization clusters without recompilation.
- Runtime Scalability: An Equalizer application can benefit from multiple graphics cards, processors and computers to scale rendering performance, visual quality and display size.
- Distributed Execution: Equalizer applications can be written to support cluster-based execution. Equalizer uses the [Collage network library](#), a cross-platform C++ library for building heterogeneous, distributed applications.

Support for Stereo and Immersive Environments: Equalizer supports stereo rendering head tracking, head-mounted displays and other advanced features for immersive Virtual Reality installations.

#### Screenshot

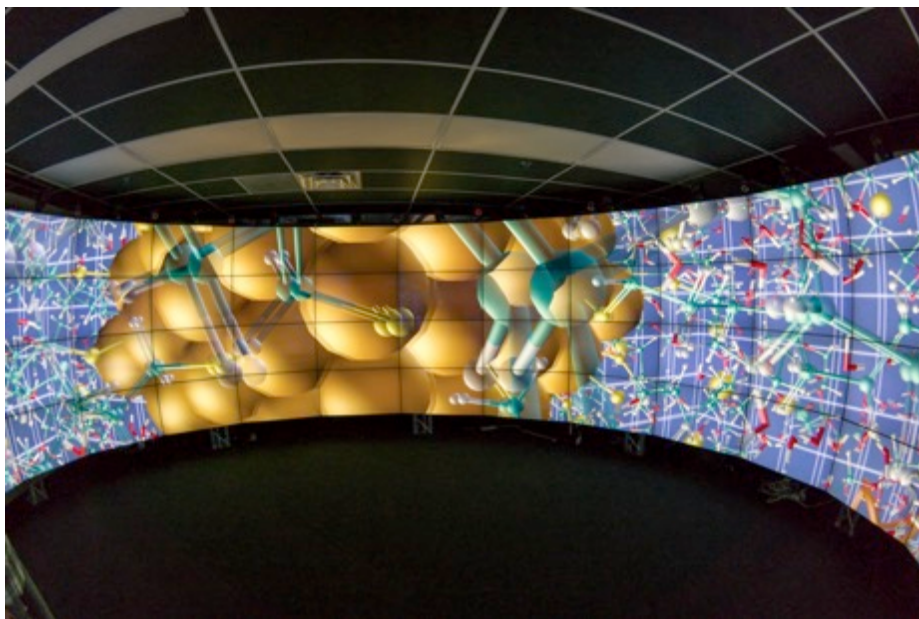


Figure 13 Equalizer on a display wall



## OmpSs

### Description

OmpSs is a fine-grained programming model oriented to shared memory environments, with a powerful runtime that leverages low-level APIs (e.g. CUDA/OpenCL) and manages data dependencies (memory regions). It exploits task level parallelism and supports asynchronicity, heterogeneity and data movement.



## PyCOMPSs

### Description

PyCOMPSs is a coarse-grained programming model oriented to distributed environments, with a powerful runtime that leverages low-level APIs (e.g. Amazon EC2) and manages data dependencies (objects and files). From a sequential Python code, it is able to run in parallel and distributed.



### Screenshot

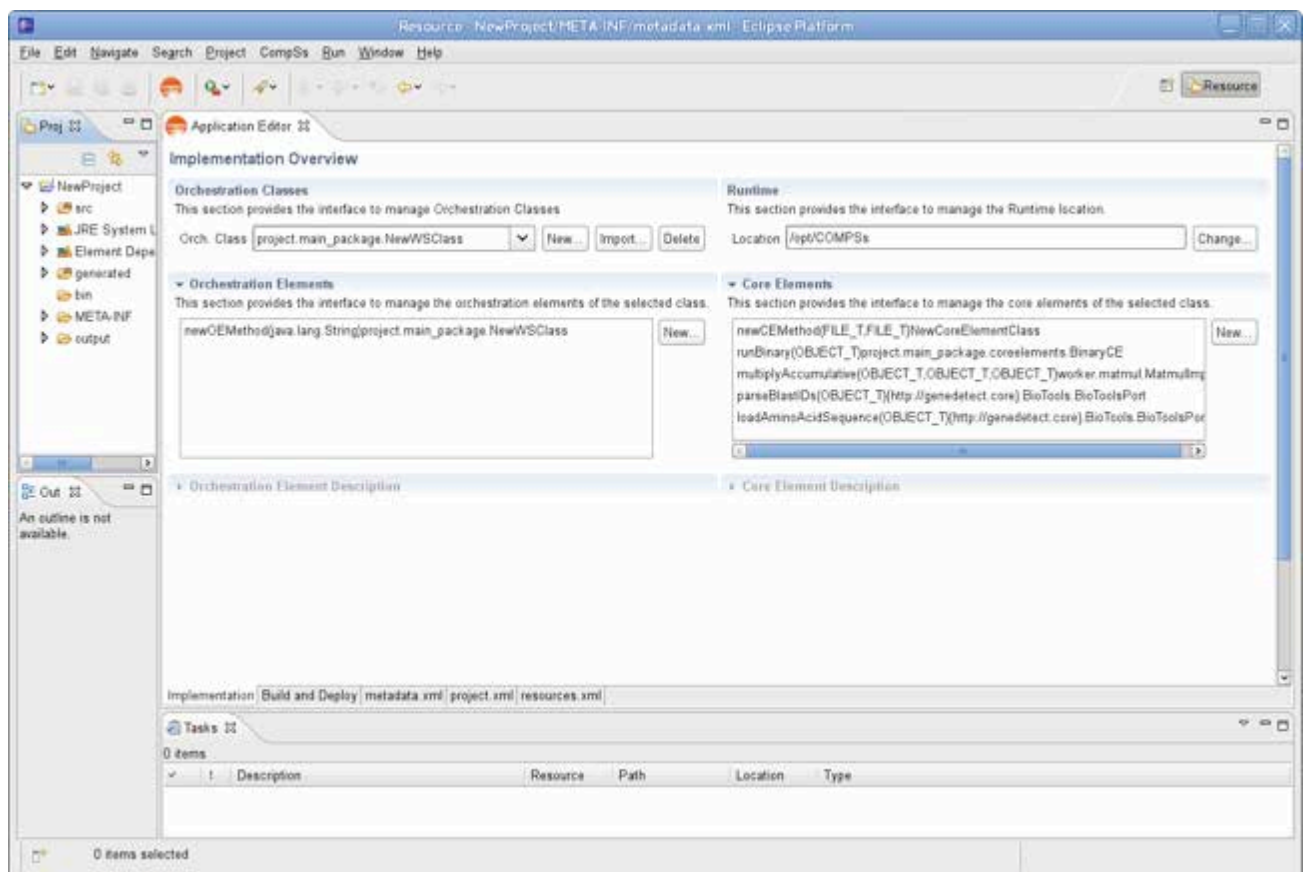


Figure 14 Screenshot of PyCOMPSs

## ViSTA Virtual Reality Toolkit

### Description

The ViSTA Virtual Reality Toolkit allows the integration of virtual reality (VR) technology and interactive, 3D visualisation into technical and scientific applications. The toolkit aims to enhance scientific applications with methods and techniques of VR and immersive visualization, thus enabling researchers from multiple disciplines to interactively analyse and explore their data in virtual environments. ViSTA is designed to work on multiple target platforms and



operating systems, across various display devices (desktop workstations, powerwalls, tiled displays, CAVEs, etc.) and with various interaction devices.

## **ZeroEQ**

### **Description**

ZeroEQ is a cross-platform C++ library to publish and subscribe for events. It provides the following major features:

- Publish events using [zeq::Publisher](#)
- Subscribe to events using [zeq::Subscriber](#)
- Asynchronous, reliable transport using ZeroMQ
- Automatic publisher discovery using Zeroconf
- Efficient serialization of events using flatbuffers



## Other software

### *MonetDB*

#### Description

When a database grows into millions of records spread over many tables and business intelligence or science becomes the prevalent application domain, a column-store database management system is called for. Unlike traditional row-stores, such as MySQL and PostgreSQL, a column-store provides a modern and scalable solution without calling for substantial hardware investments.



MonetDB pioneered column-store solutions for high-performance data warehouses for business intelligence and eScience since 1993. It achieves its goal by innovations at all layers of a DBMS, e.g. a storage model based on vertical fragmentation, modern CPU-tuned query execution architecture, automatic and adaptive indices, run-time query optimization, and a modular software architecture. It is based on the SQL 2003 standard with full support of foreign keys, joins, views, triggers, and stored procedures. It is fully ACID compliant and supports a rich spectrum of programming interfaces (JDBC, ODBC, PHP, Python, RoR, C/C++, Perl).

### *T-Storm*

#### Description

T-Storm is a platform for supporting scalable real-time analytics of massive sets of voluminous time-series. The platform is constructed over the Storm parallel dataflow engine, and supports both vertical scalability (fully utilizing high-end servers and multi-core systems) and horizontal scalability (scaling across a cluster of physical machines or even incorporating virtual cloud resources).

The current version of T-Storm enables efficient maintenance of the highly correlated time-series in linear space and near-linear computational complexity. In practice, computational complexity depends on the input time-series. This functionality is, for example, useful to identify the pairs of neurons that fire in a correlated manner. In future versions, T-Storm will enable a wide range of continuous time-series analytics. These will include single-stream analytics, such as maintaining sliding-window summaries and statistics, but also cross-stream analytics, such as continuous clustering and classification of the time series. Future versions will also support an Application Programming Interface and a high-level user interface, e.g., via R language for statistical computing.



---

## Annex 3: Infrastructure Components of the HPC Platform

The infrastructure components of the HPC Platform are described in detail in section 3 of the HPC Platform Guidebook. It provides technical details of the supercomputers, of the high-fidelity visualisation systems and of the network. The UNICORE infrastructure used as middleware for the HPC Platform services is as well documented, including the installation of UNICORE at the sites and the UNICORE Portal. The section also describes the setup of the LDAP servers used for the user management and accounting.



---

## Annex 4: Access to the supercomputers of the HPC Platform

The HPC Platform Guidebook describes in section 4 how users can get access to the supercomputers of the HPC Platform. Significant compute resources can only be provided based on proposals undergoing a scientific and technical peer-review. For some systems, small amounts of compute time can also be granted with less effort for testing purposes.