大阪大学
THE UNIVERSITY OF OSAKA

OCTOPUS

Introduction to OCTOPUS

OCTOPUS orientation session
November 5, 2025

Keichi Takahashi
D3 Center, University of Osaka

D3 Center, University of Osaka

Supercomputer (2021/5–)

Compute and storage platform
(–2024/3, 2025/9–)

Cloud platform (2024/11–)

Data aggregation platform (2021/5–)

We manage the computing infrastructure, including supercomputers, that underpins research and education both within and outside UOsaka. We also spearhead advanced research and development in large-scale computing, telecommunications, and ICT-driven education.

Agenda

- We provide on overview of our latest computing platform OCTOPUS:
  - **System overview**: overall architecture, performance, and available software
  - **Distinctive features**: provenance tracking, simulated annealing, etc.
  - **Comparison to SQUID**: differences in performance and usage
- We detail the official service and usage fees, which will be available from December.

13:00–13:25	System and usage overview (Keichi Takahashi, Associate Professor, D3 Center)
13:25–13:50	Usage application and usage fees (Naohiro Haraguchi, Technical Staff, D3 Center)
13:50–14:00	Q&A

OCTOPUS core goals

Offering latest processors

OCTOPUS offers the latest high-performance processors (Intel Granite Rapids, announced in Sep. 2024) to deliver state-of-the-art computing performance.

Alleviating congestion

As workloads will be distributed between SQUID and OCTOPUS, OCTOPUS will reduce congestion and shorten wait times during busy periods (Dec. to Mar. every year).

Supporting open science

OCTOPUS integrates middleware to track, manage, and share the input data and programs used to generate computational results (*provenance*) on the system.


NEC
OCTOPUS
Osaka university Compute
& sTOrage Platform Urging
open Science

System overview

Front-end network (10/40/100 GbE)

Interconnect (InfiniBand NDR)

NEC LX201 Ein-1 x140
- CPU: Intel Xeon 6980P (2.0GHz/128core) x2
- Memory: DDR5-6400 768 GB
- Network: InfiniBand NDR200 x1

35,840 cores
Peak FP64 perf. **2.293 PFLOPS**

General-purpose CPU nodes

DDN ES400NVX2 x1
- Enclosure x4
- DDN monitoring server x1
- DDN NFS server x2
- DDN metadata backup server x2

Effective capacity **3.58 PB**
Peak bandwidth **100 GB/s**

File servers

HPC front-end x2
- CPU: Intel Xeon 6980P x2
- GPU: NVIDIA L4x1
- Memory: 768 GB

Dedicated front-end x1
- CPU: Intel Xeon 6980P x2
- Memory: 1,152 GB

Front-end servers

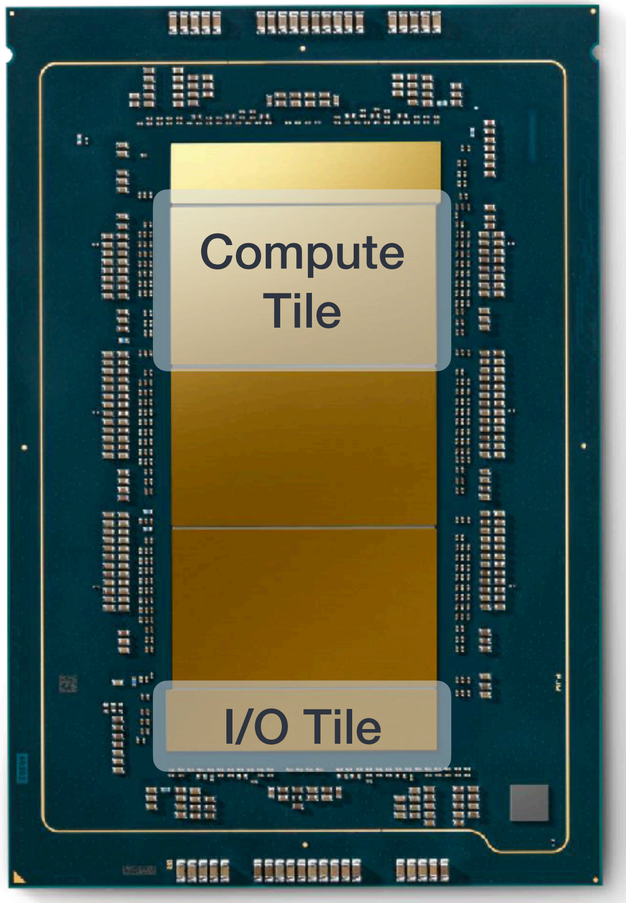
- Mgmt. network
- Mgmt. servers
- VPN for cloud interoperability
- Virtual host servers
- Virtualization mgmt. servers
- Shared storage for virtual hosts
- etc.

Misc.

Node specification

	Specification
CPU	Intel Xeon 6980P (2.0 GHz, 128 cores) x2
Theoretical perf.	16.384 TFLOP/s (=8.192 TFLOP/s x2)
Memory	768 GB (32 GB DDR5-6400 DIMM x24)
Memory bandwidth	1,228.8 GB/s (=614.4 GB/s x2)
Network	InfiniBand NDR200 x1, Gigabit Ethernet
Cooling	Water-cooled

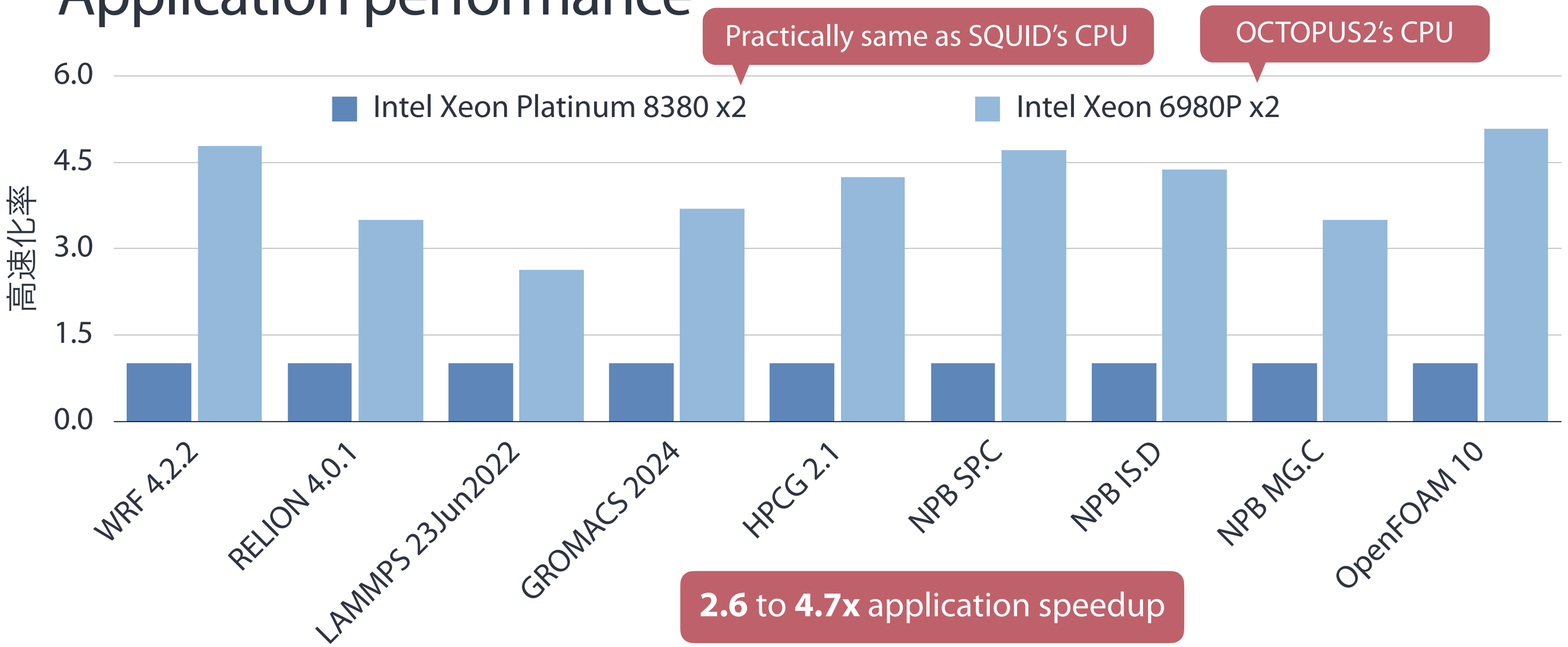
Intel Xeon 6980P CPU



Comparison to previous OCTOPUS and SQUID

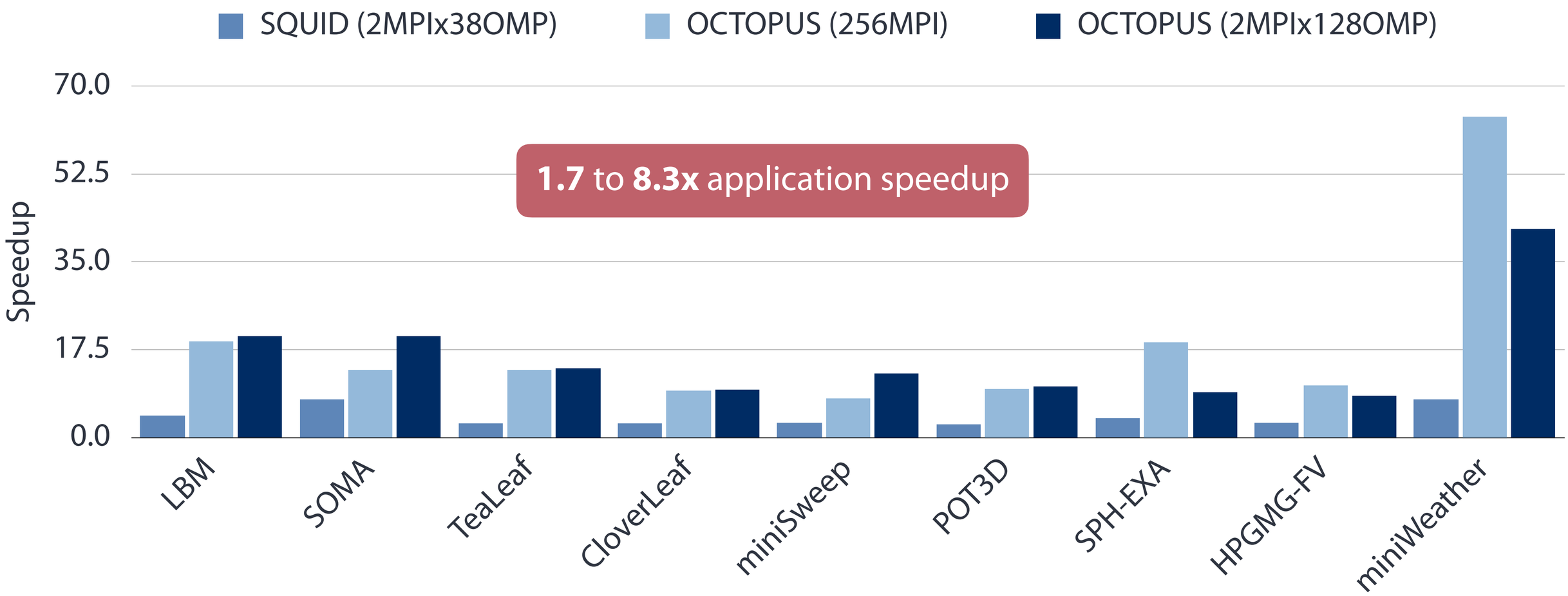
	OCTOPUS1	SQUID	OCTOPUS2	OCT2/ SQUID
CPU	Intel Xeon Gold 6126 (Skylake-SP) x2	Intel Xeon Platinum 8368 (IceLake-SP) x2	Intel Xeon 6980P (Granite Rapids) x2	-
# of cores/CPU	12 cores	38 cores	128 cores	**3.3x**
Performance/CPU	0.998 GFLOP/s	2.918 TFLOP/s	8.192 TFLOP/s	**2.8x**
Memory BW/CPU	127 GB/s	204 GB/s	614 GB/s	**3.0x**
Memory capacity	192 GB	256 GB	768 GB	**3.0x**
LLC capacity/CPU	19 MB	57 MB	504 MB	**8.8x**
Power/CPU	125 W	270 W	500 W	**1.8x**

Application performance



* Data obtained from https://www.phoronix.com/review/intel-xeon-6980p-performance
* The data shown does not represent a performance guarantee.

Application performance (comparison to SQUID)



* Measured on SQUID and OCTOPUS using the SPEChpc 2021 benchmark suite (tiny size).
* y-axis indicates speedup over a baseline system(TU Dresden's Taurus system)

Provided software

Category	Software
Programming language	Intel oneAPI Base/HPC Toolkit, GNU Compiler Collection, Python, R, Octave, Julia
Library	Intel MPI, Open MPI, Intel oneMKL, GNU Scientific Library, netCDF, PnetCDF, HDF5, TensorFlow, Keras, PyTorch, pbdR
Container runtime	Podman, Apptainer, Docker
ISV apps	Gaussian, IDL, AVS/Express, Amazon DCV
OSS apps	GROMACS, OpenFOAM, LAMMPS, GAMESS, ABINIT-MP, RELION, ADIOS, VisIt, CTFFIND, Flash Code, FreeFem++, GENESIS, MotionCor3, SMASH, Quantum Espresso, ResMap, NEC Vector Annealing, ParaView, Gnuplot, ImageMagick, NcView

* See https://www.hpc.cmc.osaka-u.ac.jp/system/manual/octopus2-use/software/ for complete list and version info.

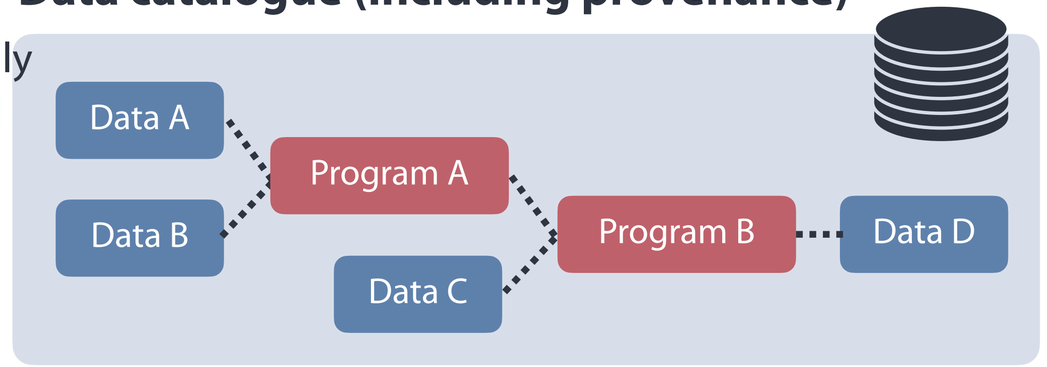
Provenance tracking system (SCUP-HPC)

- OCTOPUS can track and manage what data and programs were used and how the results were generated (i.e., *computational provenance*).
- Provenance information will be accessible to non-OCTOPUS users as well, contributing to improved reproducibility of research outcomes.

Scientist executes programs on OCTPUS and generates research data

Automatically genearted

Data catalogue (including provenance)



Research productivity

Data utilization

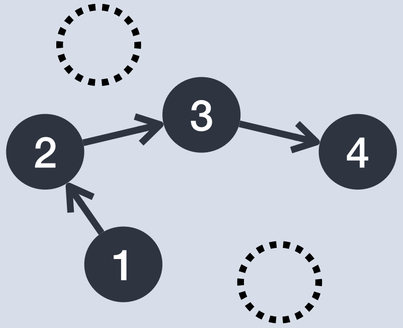
Data reliability and reproducibility

* Provenance tracking disabled by default and is enabled via an option in the job script.
* See https://www.nri.cmc.osaka-u.ac.jp/ja/2025/09/12/pr-scup-hpc/ for details about SCUP.

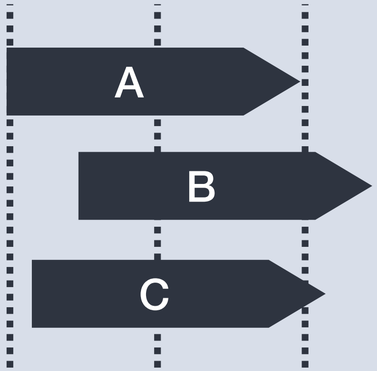
Simulated annealing (NEC Vector Annealing)

- We offer NEC Vector Annealing, a software to quickly solve **combinatorial optimization problems** free of charge to OCTOPUS users.
- Vector Annealing can solve Quadratic unconstrained binary optimization (QUBO) problems given through a Python/C++ API.

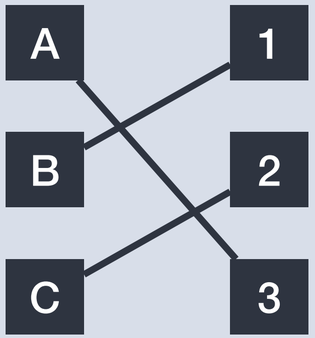
Ordering



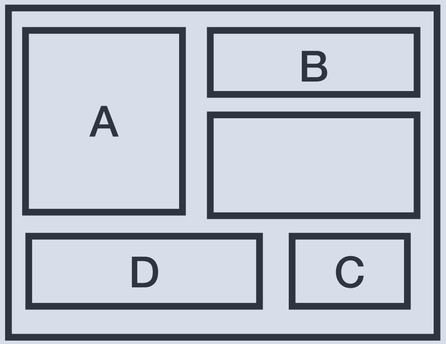
Scheduling



Matching



Resource allocation

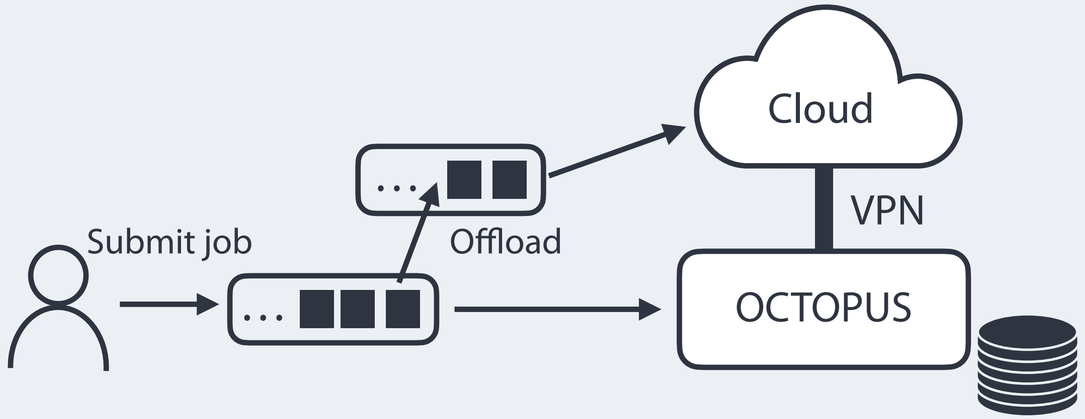


* See https://www.hpc.cmc.osaka-u.ac.jp/system/manual/octopus2-use/vector-annealing/ for usage.

Other notable features

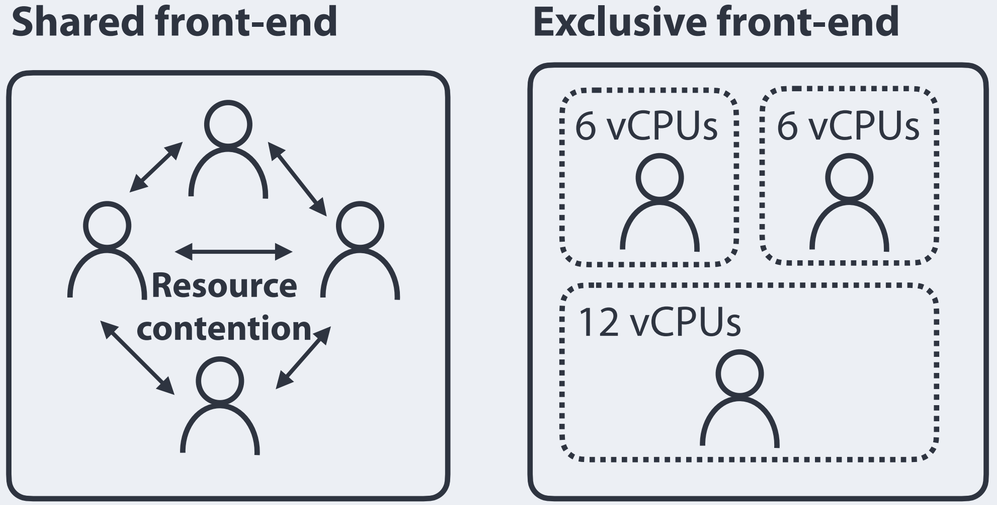
Cloud bursting

- Reduces wait times by offloading a subset of the jobs submitted to OCTOPUS to public clouds for execution.
- Only jobs that are explicitly allowed by users in the job scripts for cloud bursting will be forwarded to the cloud.

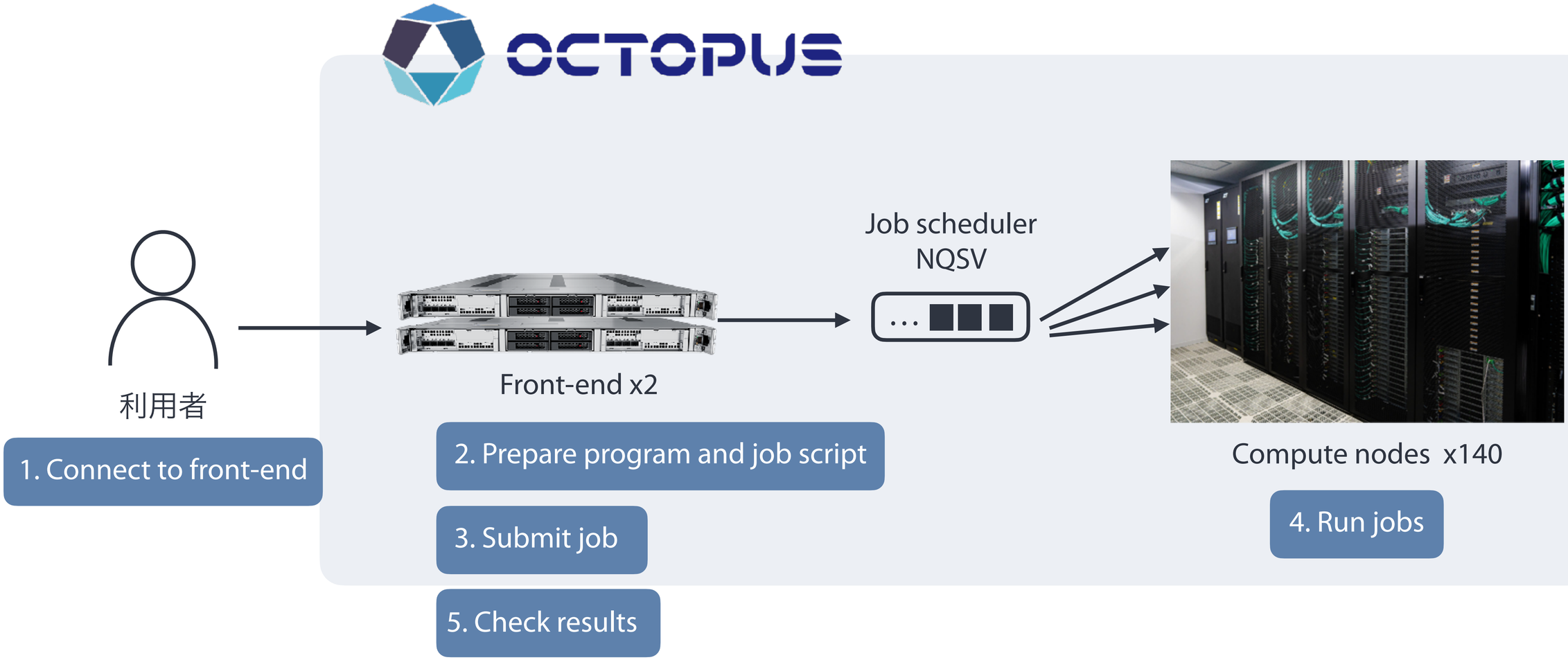


Exclusive front-end nodes

- OCTOPUS offers dedicated virtual front-end servers for each group (additional fees apply).
- These can be used for resource-intensive tasks such as pre- and post-processing.



OCTOPUS usage flow



Logging in

- Users connect to front-end node via Secure SHell (SSH).
- Two-factor authentication using password and one-time password (OTP) is required.
  - Please install a 2FA app on your PC or smartphone.
  - Public key authentication is *not* supported.
  - Password and OTP are different from those of SQUID.

```
ssh <user ID>@octopus.hpc.osaka-u.ac.jp
```

SSH client
Windows: Windows Terminal, PuTTY, ConEmu, etc.
macOS: Terminal, iTerm2, Ghostty, etc.

2FA app
Google Authenticator
Microsoft Authenticator

Submitting jobs

Example of a job script

```
#!/bin/bash

#PBS -q OCT
#PBS --group=G12345
#PBS -l elapstim_req=00:01:00
#PBS -T intmpi
#PBS -b 4

module load BaseCPU

cd $PBS_O_WORKDIR

mpirun ${NQSV_MPIOPTS} -np 1024 ./a.out
```

Job classes

Job class	Max. runtime	Max. # of cores	Max. memory	Max. # of nodes	Notes
OCT	120 h	32,768	92.75 TiB	128	
OCT-H	120 h	32,768	92.75 TiB	128	High priority
OCT-S	120 h	128	371 GiB	1	Node-sharing
DBG	10 m	512	1,484 GiB	2	Debug
INT	10 m	512	1,484 GiB	2	Interactive

Resource request

- `-q OCT`: Job class (see table above)
- `--group=...`: Group name
- `-l elapstim_req...`: Max runtime

Commands to execute

- `module load...`: Load Intel compiler environment
- `cd ...`: Move to directory where job was submitted
- `mpirun...`: Launch MPI application

Submit with `qsub job.sh`

See https://www.hpc.cmc.osaka-u.ac.jp/system/manual/octopus2-use/jobscript/ for details.

Checking the status of a job

QUE: Queueing, **RUN**: Running

CPU: CPU time, **Elapse**: Elapsed time

Job status

```
$ qstat
RequestID       ReqName  UserName Queue     Pri STT S   Memory      CPU   Elapse R H M Jobs
--------------- -------- -------- -------- ---- --- - -------- -------- -------- - - - ----
22038.oct       job.sh   x12345   O8         0 QUE -    0.00B     0.00        0 Y Y Y    4
```

Planned start time

```
$ sstat
RequestID       ReqName  UserName Queue     Pri STT Date(PLANNED START)
--------------- -------- -------- -------- ---- --- -------------------
22038.oct       job.sh   x12345   O8         0 QUE 2025-11-06 01:25:48
```

Standard output and error are written to files after job completion

Stdout: `<job script name>.o<request ID>`

Stderr: `<job script name>.e<request ID>`

* You can use qcat to monitor the outputs of jobs while they are running.

Differences from SQUID

- **Login**
  - Same as SQUID (password + OTP), but credentials and front-end node are different.
- **Job submission**
  - Since OCTOPUS uses the same scheduler as SQUID, there are no major differences.
  - However, the queue configurations (e.g., queue names, number of nodes, job submission/execution limits) are different.
- **File system**
  - Group quota is 5TB on both systems, home quota is 100GB per user on OCTOPUS.
  - SQUID's file system is accessible from OCTOPUS (from the front-end nodes only).

See https://www.hpc.cmc.osaka-u.ac.jp/system/manual/octopus2-use/ for details.

Summary

- The new OCTOPUS system will begin full-scale operation in **December 2025**.
- OCTOPUS is expected to deliver a **several-fold improvement in real-world application performance** compared to the old OCTOPUS and SQUID.
- OCTOPUS will also offer unique and advanced features, such as support for open science and simulated annealing.
- We encourage you to take this opportunity to try OCTOPUS for your research.
- We are ready to provide **comprehensive user support**,
  including answering questions via email,
  offering in-person consultations, holding various training sessions,
  and providing performance tuning assistance.

FAQ (1/3)

- **Will the Intel classic compilers (icc/icpc/ifort) be provided?**
  - The development of the Intel classic compilers has already ended, and they will receive no further updates, including bug fixes. Therefore, we do not plan to provide them on OCTOPUS.
- **I want to use auto-parallelization.**
  - The new Intel compilers (icx/icpx/ifx) do not have an auto-parallelization feature. Please use manual hybrid parallelization with OpenMP.
- **How can I exchange files between SQUID and OCTOPUS?**
  - SQUID and OCTOPUS mount each other's file systems. Please use /octfs on SQUID and /sqfs on OCTOPUS.
- **How do I use the simulated quantum annealing feature?**
  - Please refer to this page.

FAQ (2/3)

- **Is OCTOPUS available for use through HPCI/JHPCN/open calls for proposals?**
  - Starting with the 2026 fiscal year projects, we award computing hours through HPCI and JHPCN. It is also available for use through our open call for research proposals program.
- **What is the official name of OCTOPUS?**
  - It is "*Osaka university Compute & sTOrage Platform Urging open Science*."
- **Will GPUs be available on OCTOPUS in the future?**
  - OCTOPUS is a CPU-only system. If you wish to use GPUs, please consider the GPU nodes on SQUID or mdxII.

FAQ (3/3)

- **Is it OK to run programs on the front-end node?**
  - It is fine for lightweight and short-running programs. However, high-load and long-running programs must be executed as jobs. Programs with high loads running for extended periods on the front-end nodes may be terminated without notice.
- **Which has a shorter wait time, SQUID or OCTOPUS?**
  - It depends on the demand, so it is unknown at this time.
- **I want to use OTP on multiple devices.**
  - Please use the export feature of your two-factor authenticator or a password manager.
- **Is the usage fee the same even if I use fewer cores than a full node?**
  - Fewer OCTOPUS points will be consumed if you request resources equivalent to less than 1/2 or 1/4 of a node. Please specify the OCT-S job class and set the cpunum_job and memsz_job options.