Research towards Future Supercomputer for Everybody

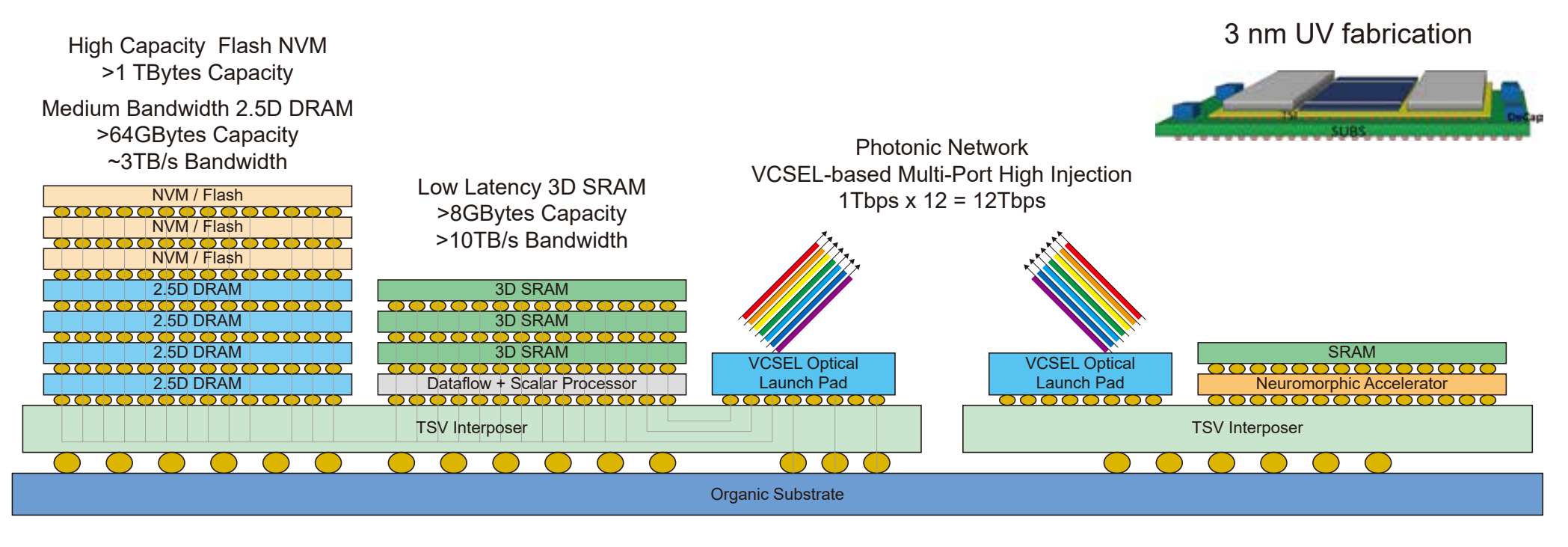
Exploring Next-Gen Architecture with Complex Memory Hierarchy

To achieve higher performance and larger capacity on recent and future architectures, we need to explore next-gen memory hierarchy, including heterogeneous devices. Also placement of many-cores and memory devices can be reconsidered; 3D stacking of cores and memory chips may go mainstream in HPC/Big-data area.

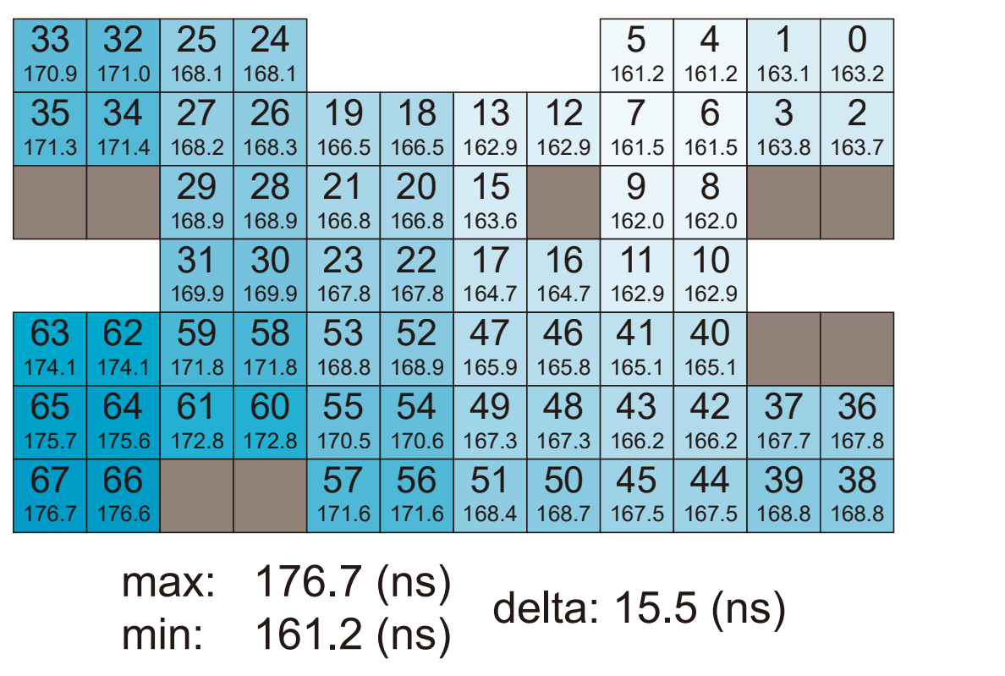
Understanding Memory Performance

Under the above-mentioned assumptions, understanding memory performance will be more challenging for data and address traffic routed among many-cores, which may often conform 2D mesh.

To analyze its effects, we have conducted preliminary measurements of memory latency from every core of Xeon Phi. We see 9% difference in the current architecture; the effect will be expanded in future architecture where chip cores are also stacked.



An Example of Future Architecture with 3D Stacking Technology and Heterogeneous Memory



33 170.9	32 171.0	25 168.1	24 168.1					5 161.2	4 161.2	1 163.1	0 163.2
35 171.3	34 171.4	27 168.2	26 168.3	19 166.5	18 166.5	13 162.9	12 162.9	7 161.5	6 161.5	3 163.8	2 163.7
		29 168.9	28 168.9	21 166.8	20 166.8	15 163.6		9 162.0	8 162.0		
		31 169.9	30 169.9	23 167.8	22 167.8	17 164.7	16 164.7	11 162.9	10 162.9		
63 174.1	62 174.1	59 171.8	58 171.8	53 168.8	52 168.9	47 165.9	46 165.8	41 165.1	40 165.1		
65 175.7	64 175.6	61 172.8	60 172.8	55 170.5	54 170.6	49 167.3	48 167.3	43 166.2	42 166.2	37 167.7	36 167.8
67 176.7	66 176.6			57 171.6	56 171.6	51 168.4	50 168.7	45 167.5	44 167.5	39 168.8	38 168.8

max: 176.7 (ns)
min: 161.2 (ns)
delta: 15.5 (ns)

Memory access latency from every core of Xeon Phi 7285. As the target of access, one of MCDRAM modules, placed close to core 4, 5 is used.

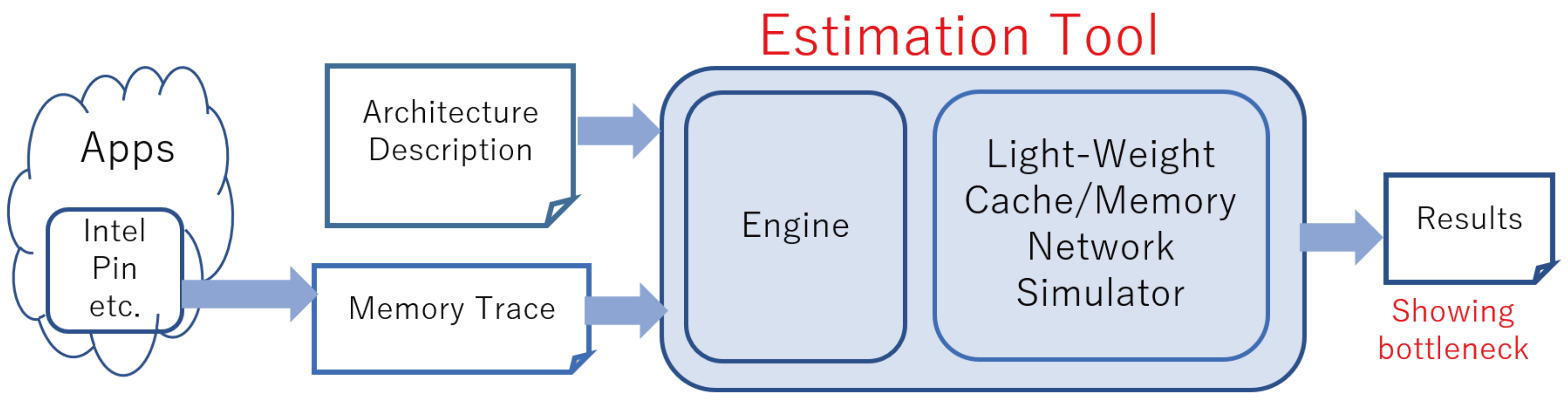
Memory Performance Estimation Tool

Towards high performance node architecture in future, it is important to understand application performance, not only synthetic benchmark.

For this purpose, we are developing an estimation tool of memory performance of future architecture, whose features are:

Light weight: it takes history of the entire application execution as input, which can be too heavy for cycle-wise simulators.

To support complex structure: it considers placement of many cores and topology including chip-lets and memory chips.



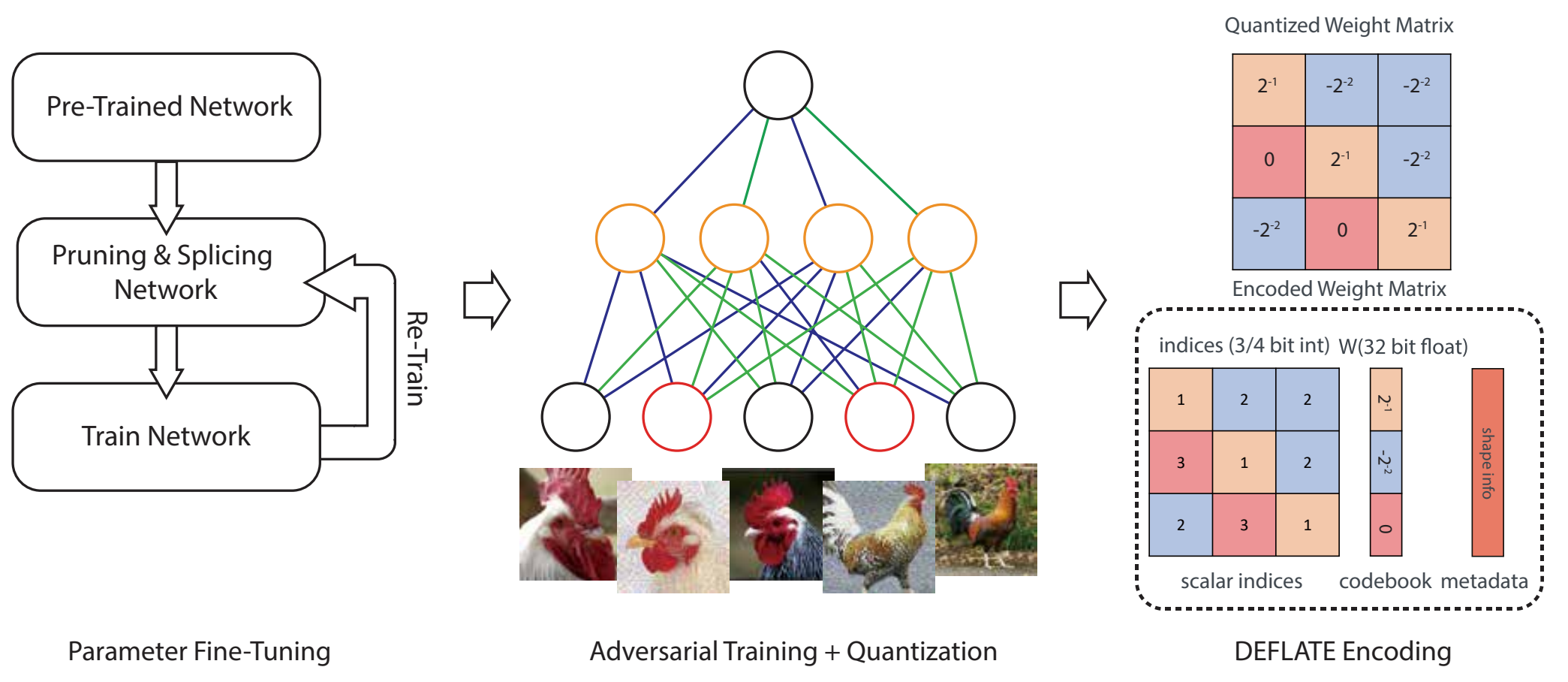
Acknowledgments: This research is supported by the New Energy and Industrial Technology Development Organization (NEDO), Japan.

Contact: Toshio Endo (endo@is.ttiech.ac.jp)

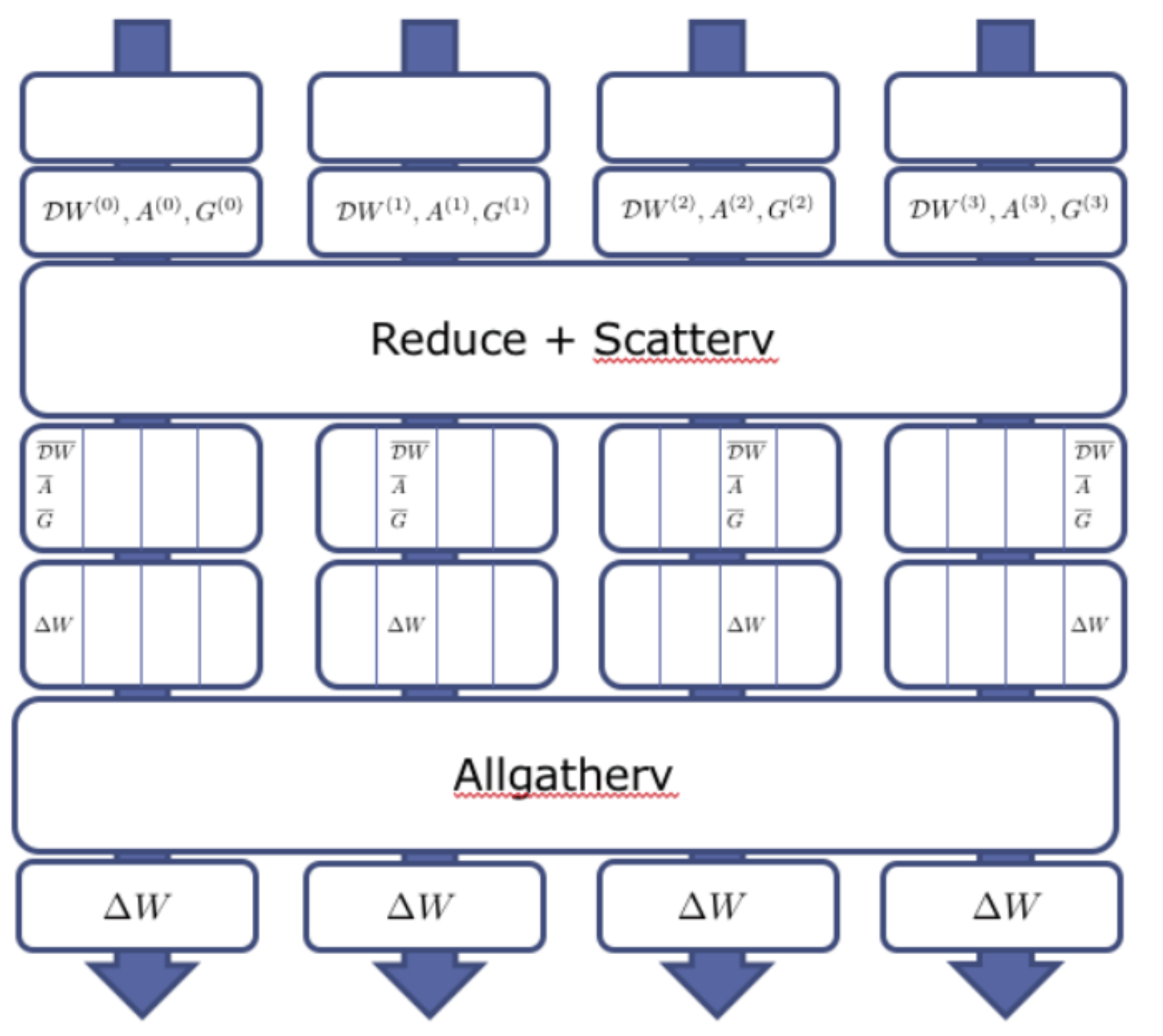
Deep Learning Technology Dealing with Deeper Memory Hierarchy

Deep Compression

Deep Compression focuses on shrinking model size whilst retaining accuracy of trained models. In this work, pruning and splicing for fine-tuning and adversarial training as a regularizer resulting in better accuracy and up to 91x compression rate without compromising on any other performance loss.



Hybrid data and model parallelism


$DW^{(0)}, A^{(0)}, G^{(0)}$
$DW^{(1)}, A^{(1)}, G^{(1)}$
$DW^{(2)}, A^{(2)}, G^{(2)}$
$DW^{(3)}, A^{(3)}, G^{(3)}$
Reduce + Scatterv
$\overline{DW}$ $\overline{A}$ $\overline{G}$
ΔW
Allgatherv
ΔW

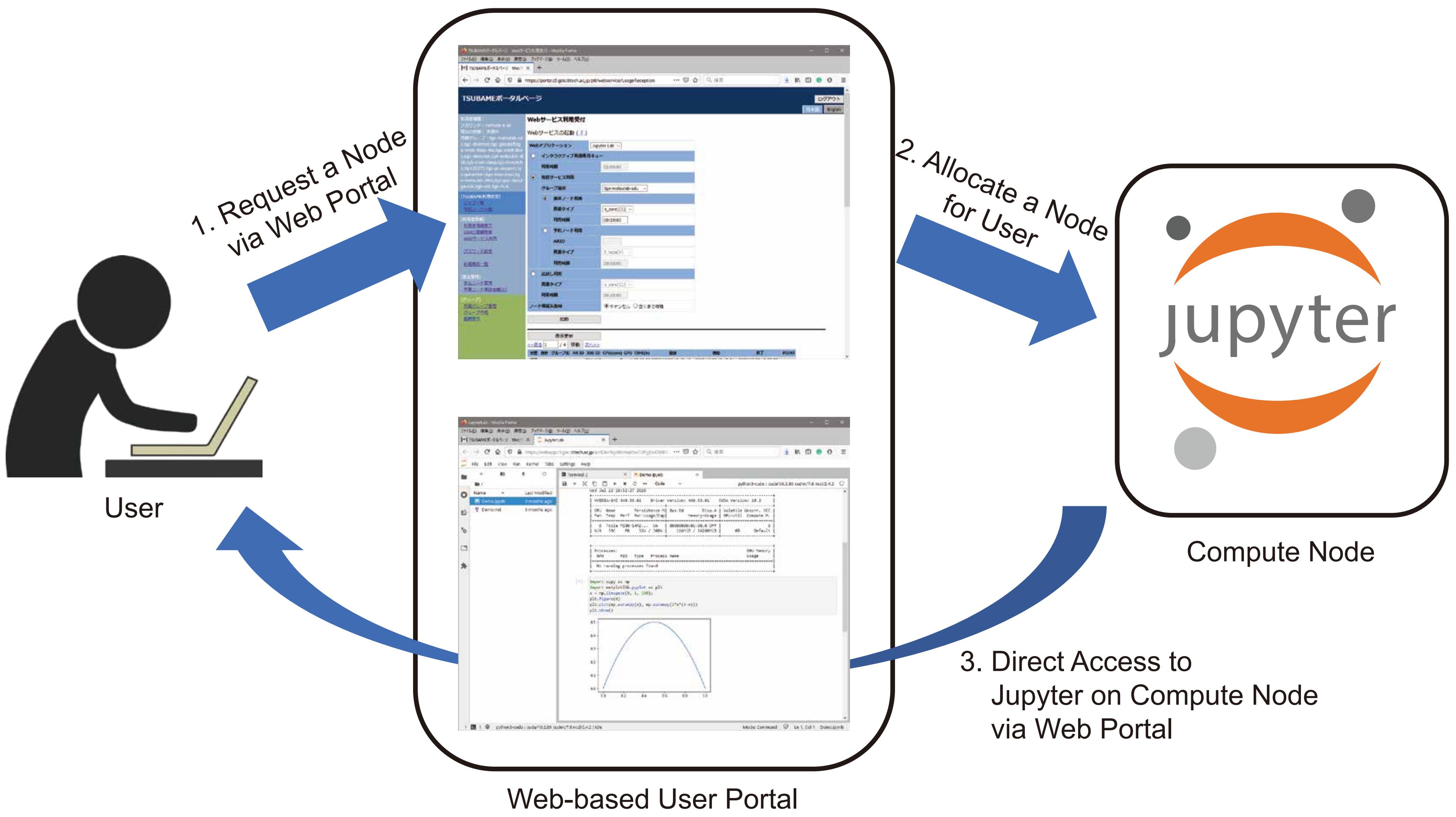
Second order optimization methods require the communication of the Hessian matrix. The present method reduces the communication significantly through Kronecker factorization and the use of reduce-scatter and allgather collectives to swtich between data parallel and model parallel execution. The optimal implementation of allreduce operations has the same communication pattern, so the present method simply inserts the computation of the Hessian.

Web-based Interactive Access to Supercomputer Nodes

Supercomputers are traditionally designed to execute large and non-interactive jobs with a batch scheduler. This style does not match the users who want to interact with compute nodes: debugging, visualization, and education of novice users in the classroom. In TSUBAME3.0 supercomputer, we introduced two new features to satisfy such demands:

Interactive use only nodes: We spare four nodes as dedicated nodes for shared interactive use. Users can run interactive jobs without waiting for job execution, even if the compute nodes are filled with batch jobs. Performance might not be optimal as the nodes are shared with multiple jobs, but still acceptable for such interactive usage.

Web-based access to Jupyter Lab on compute nodes: Novice users can use command-line shell and Python console running on high-performance TSUBAME nodes without any complicated knowledge of Linux, such as SSH key-pair authentication.


1. Request a Node via Web Portal
2. Allocate a Node for User
3. Direct Access to Jupyter on Compute Node via Web Portal
User
jupyter
Compute Node
TSUBAMEポータルページ
Web-based User Portal

https://www.gsic.titech.ac.jp/sc20