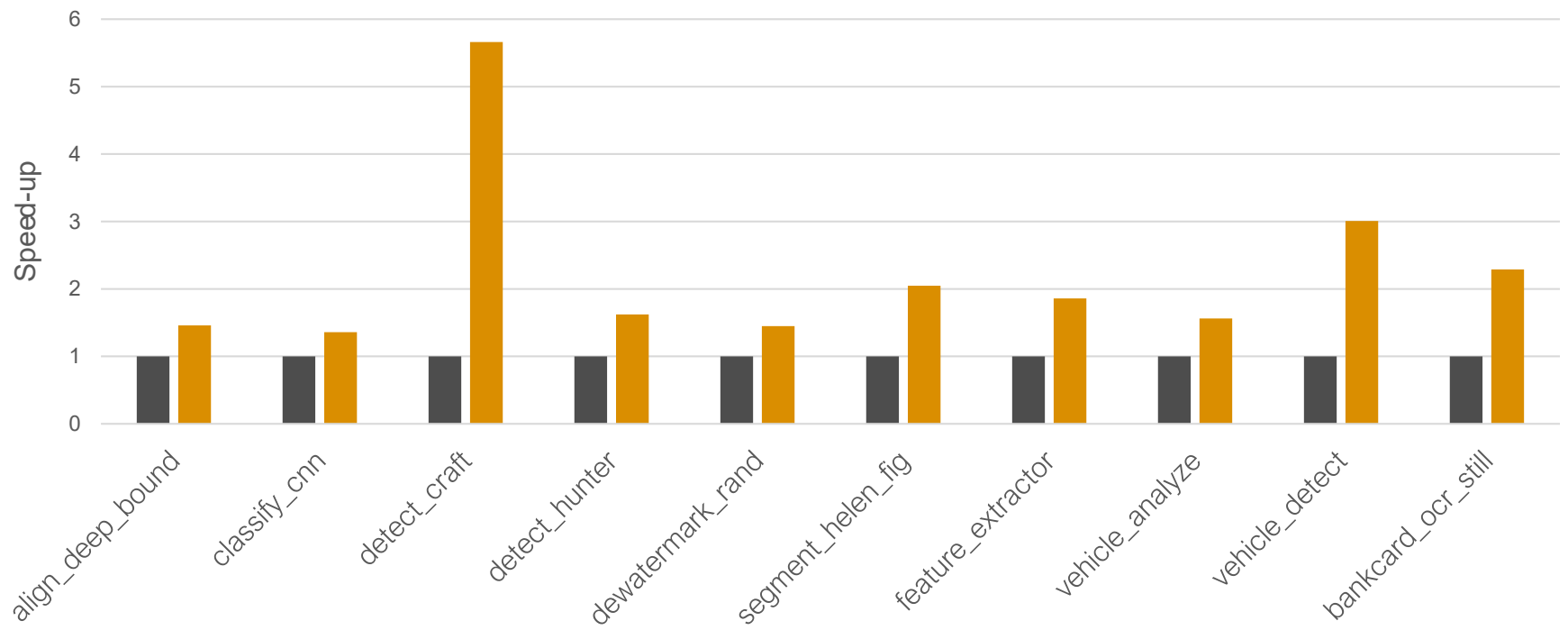
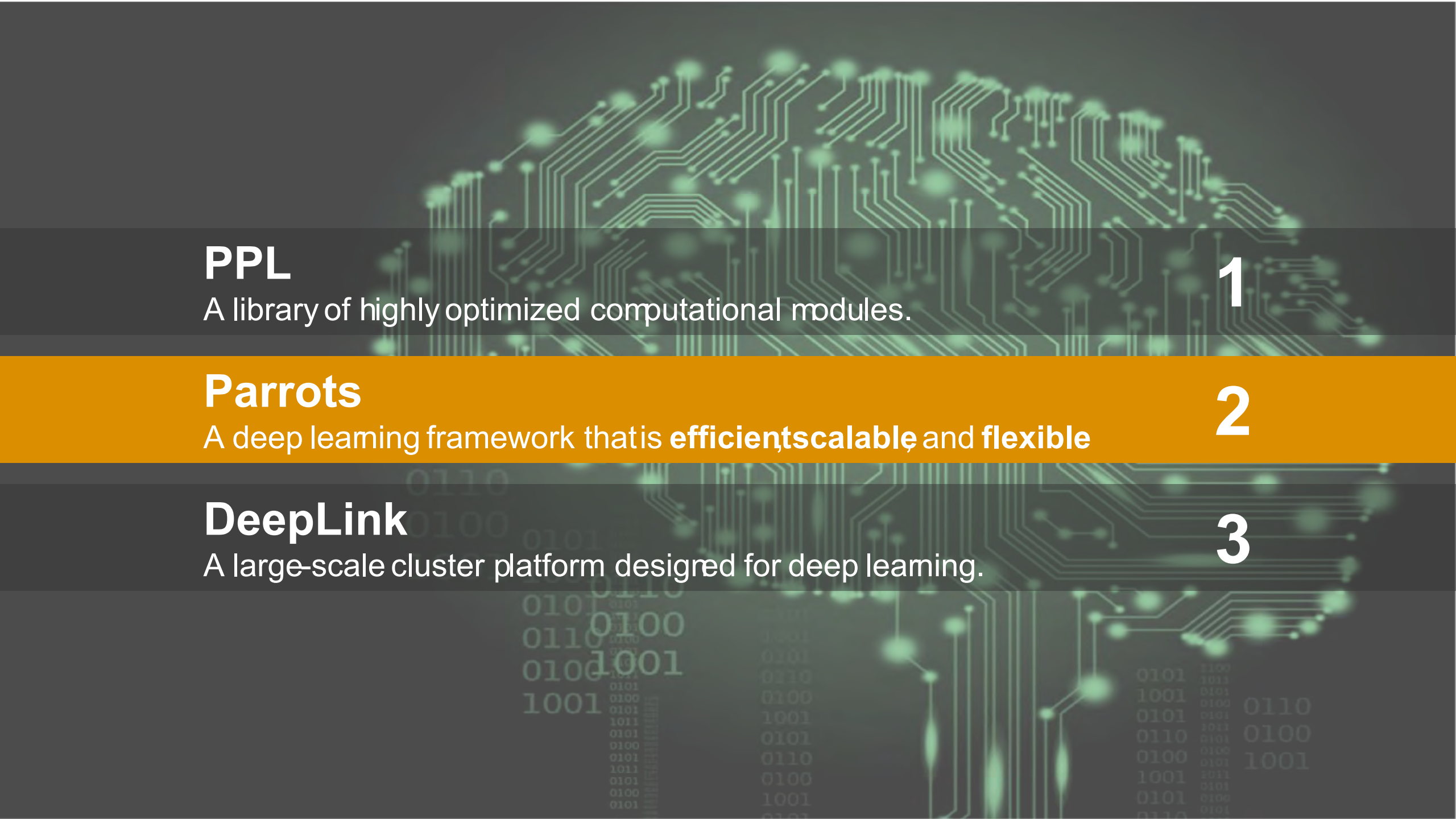
PPL ARM vs. Caffe ARM





PPL

A library of highly optimized computational modules.

1

Parrots

A deep learning framework that is **efficient**, **scalable**, and **flexible**

2

DeepLink

A large-scale cluster platform designed for deep learning.

3

Popular Frameworks

- Limited support of multi-GPU.
- No support of distributed training.
- Too restricted, difficult to support novel models.

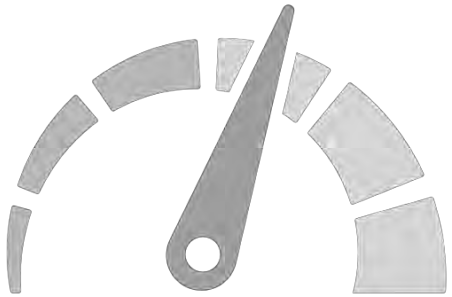
- Limited support of distributed training.
- Very flexible but you have to write low-level codes – poor productivity.

- By Google
- Designed for high scalability
- Too resource-demanding and performance remains sub-optimal.

- Designed for improved user experience & flexibility.
- Pure python.
- Limited efficiency & scalability.

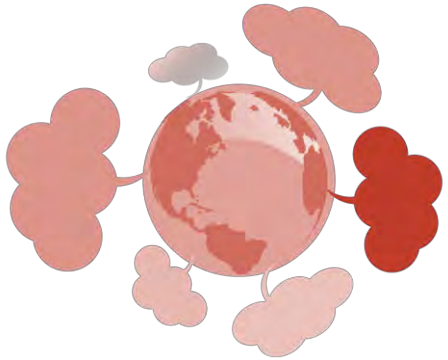
Parrots

A New Deep Learning Framework for the Future



Efficiency

- Automatic parallelize operations at different layers by dependency analysis
- I/O and communication latency hidden via careful coordination
- Highly optimized memory reuse



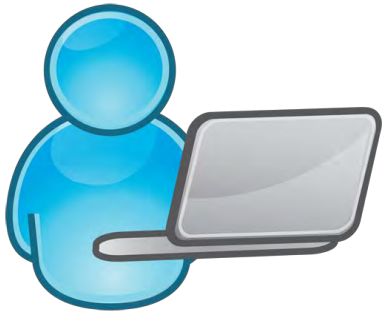
Scalability

- Devised for distributed learning on heterogeneous architectures from the very beginning
- Various communication schemes optimized for different environments
- Builtin support of RDMA

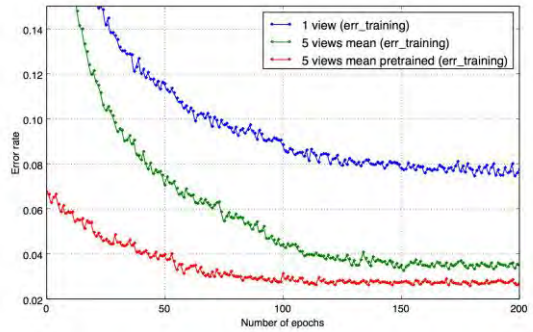
Productivity

- Battery included.
- Mainstream computer vision frameworks and pre-trained models are provided out of box.
- One network, multiple flows.
- Modularized design: compose large frameworks with model-wares.

User Scenario

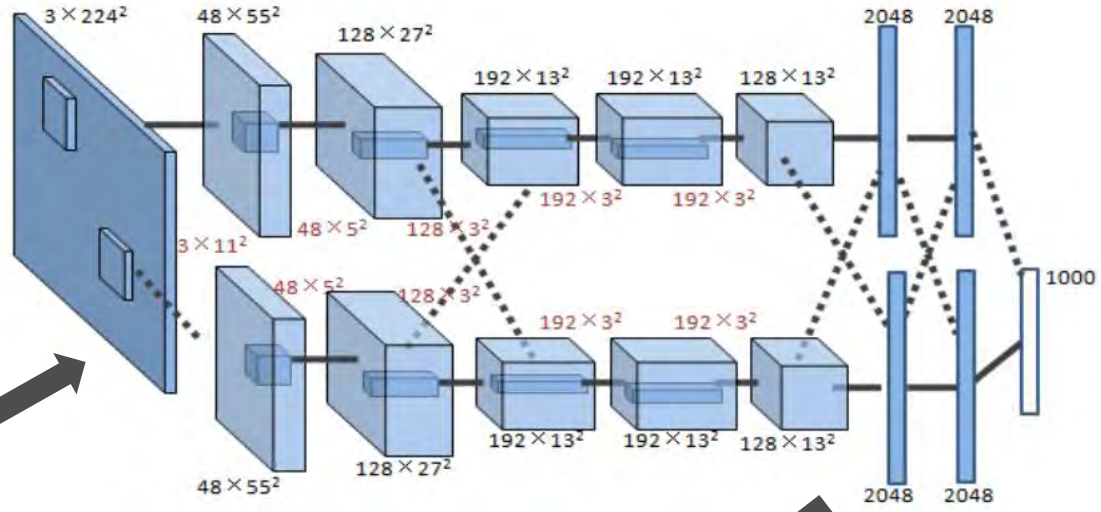


progress monitor



Model specification

training data



Session specification

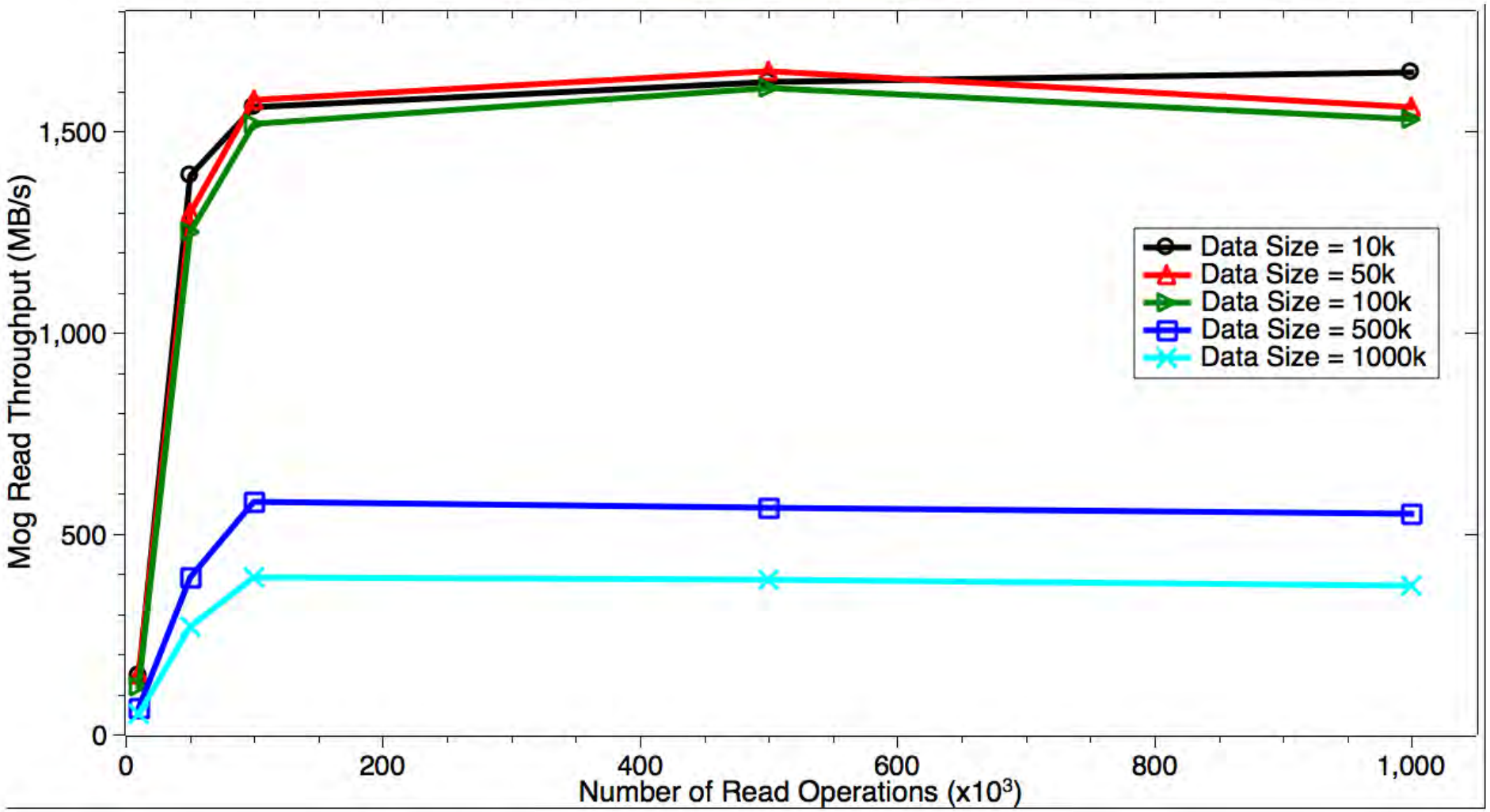
- data source
- iterations
- learning rate settings
- ……

Model params

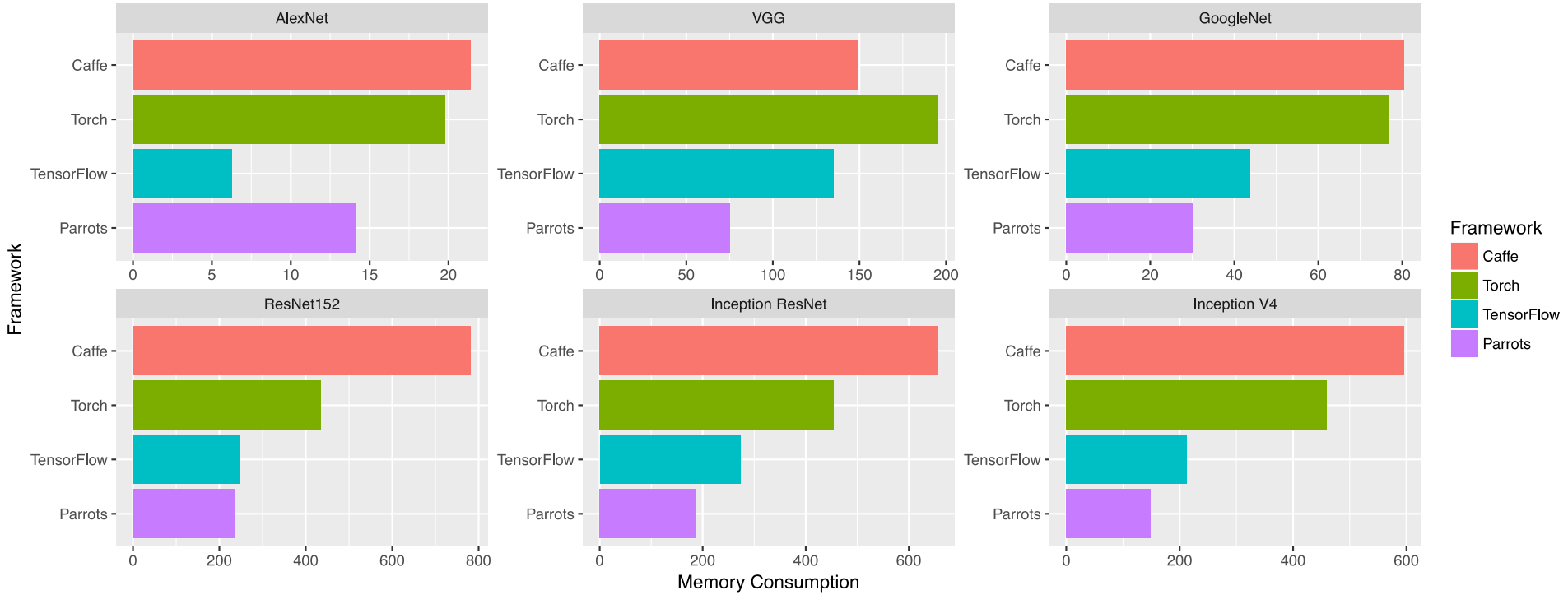
Efficiency

- Small file IO
- GPU memory usage
- Training speed

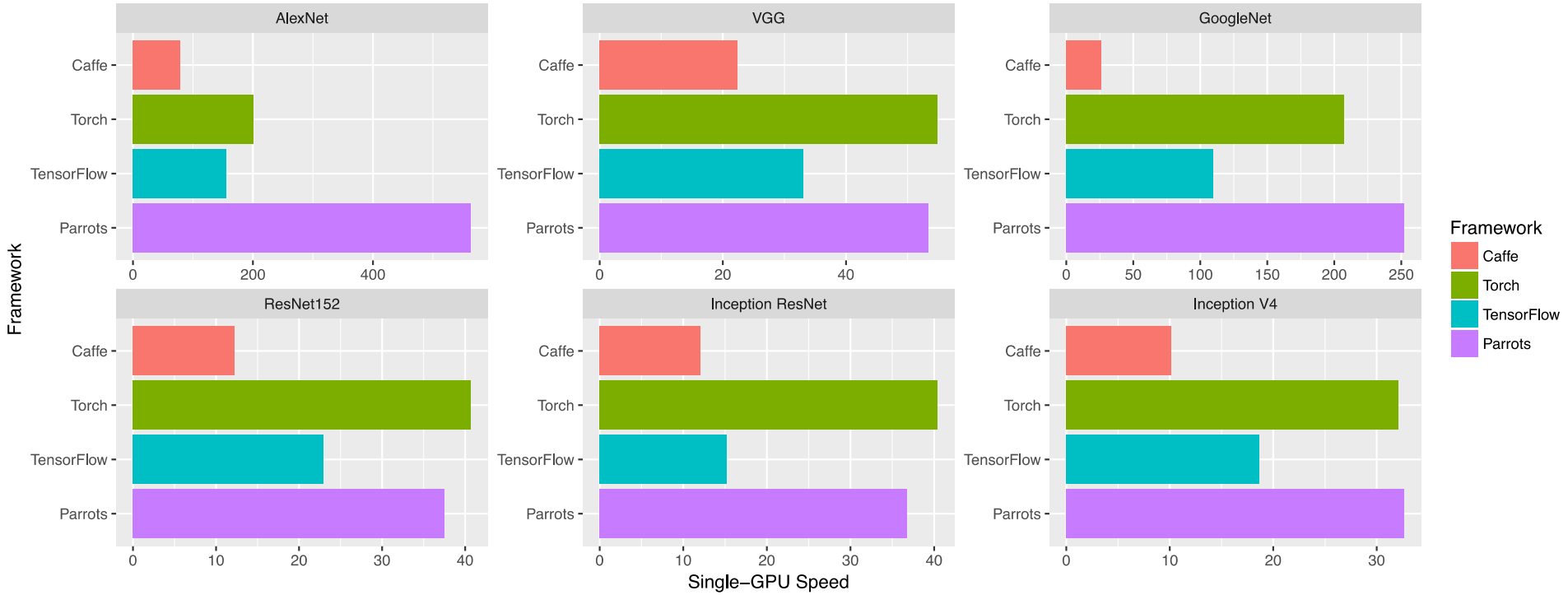
Highly concurrent Small File Access



Comparison on Memory Consumption

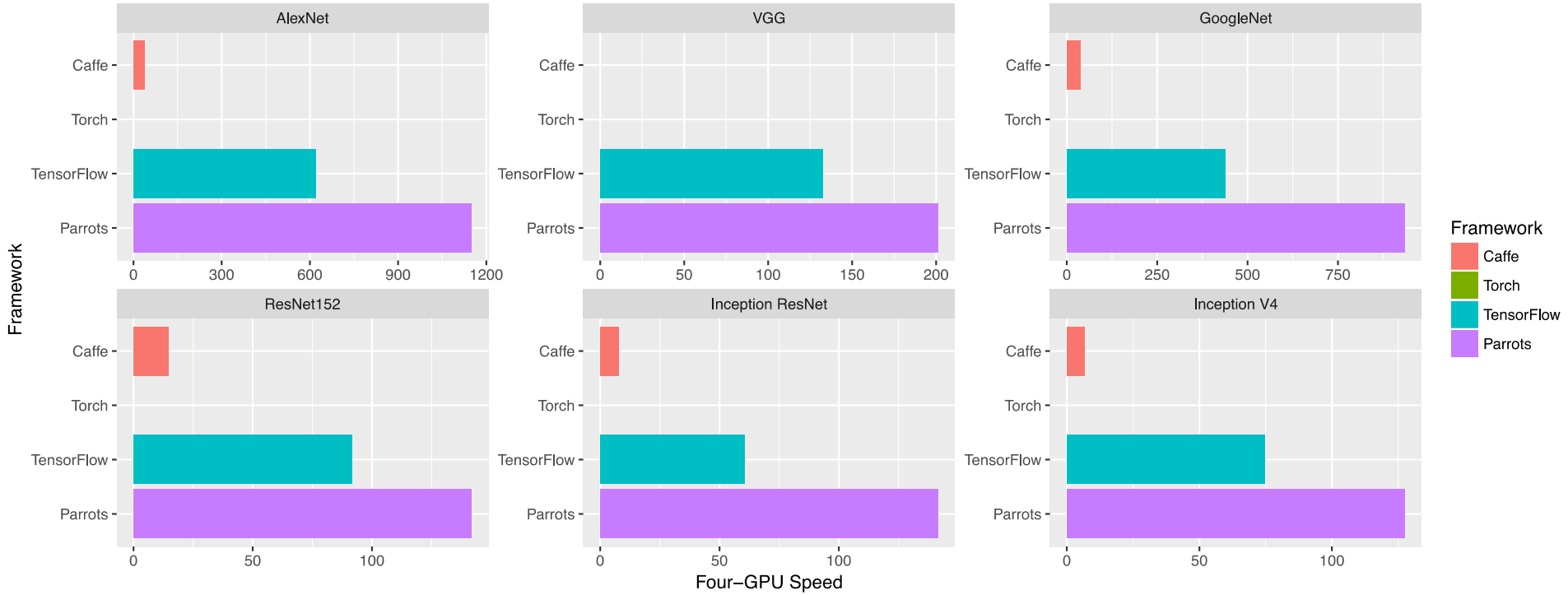


Comparison on Single-GPU Speed

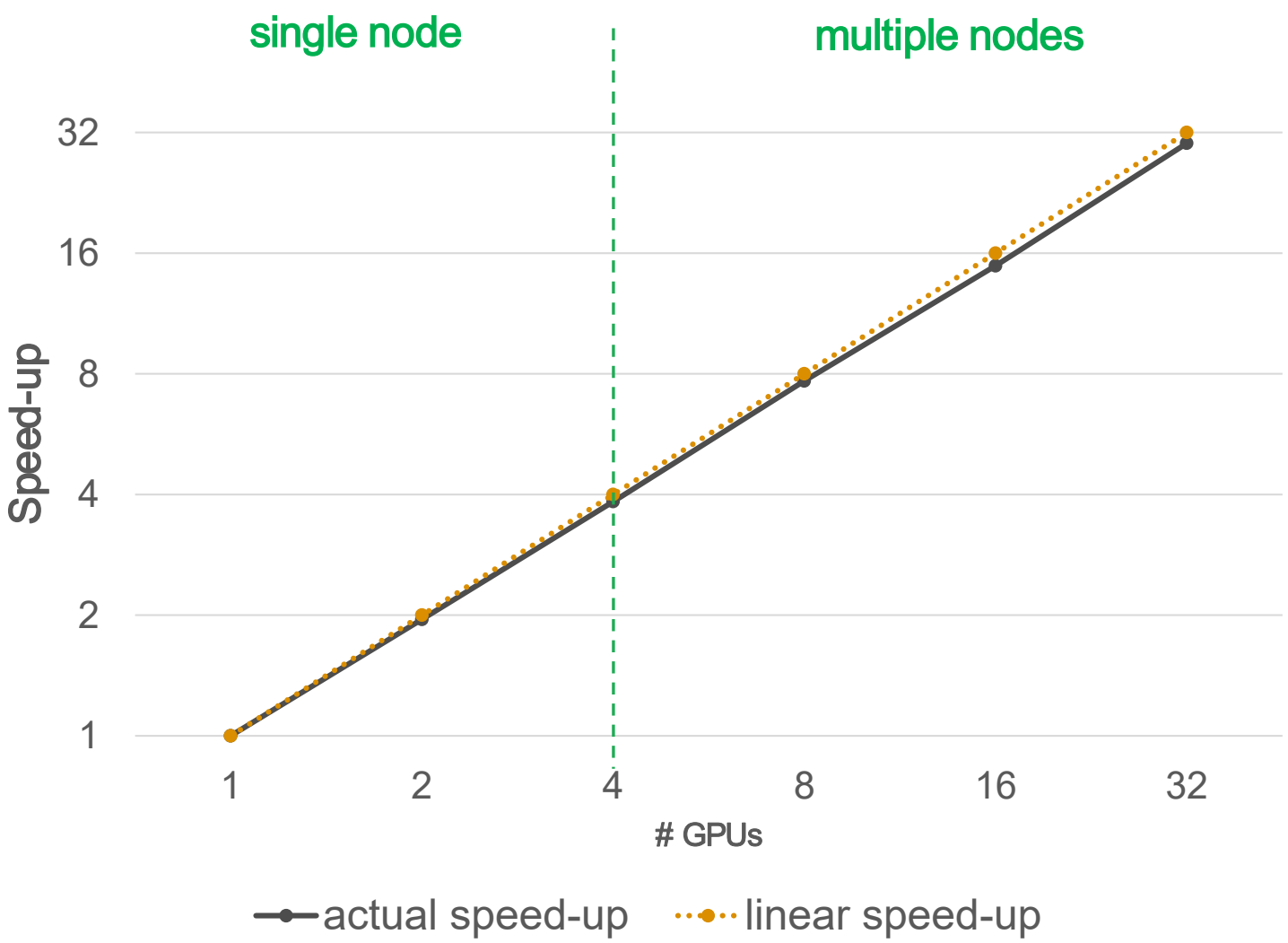


* Higher is better

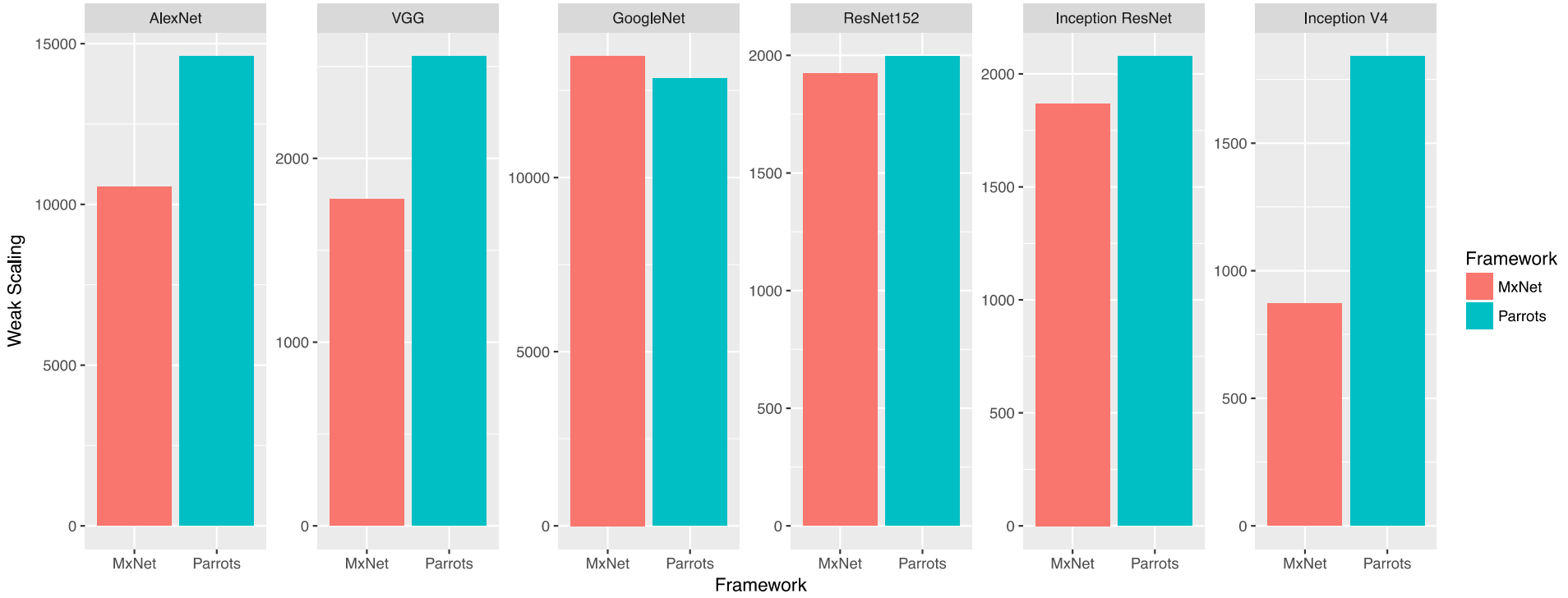
Comparison on Multi-GPU Speed



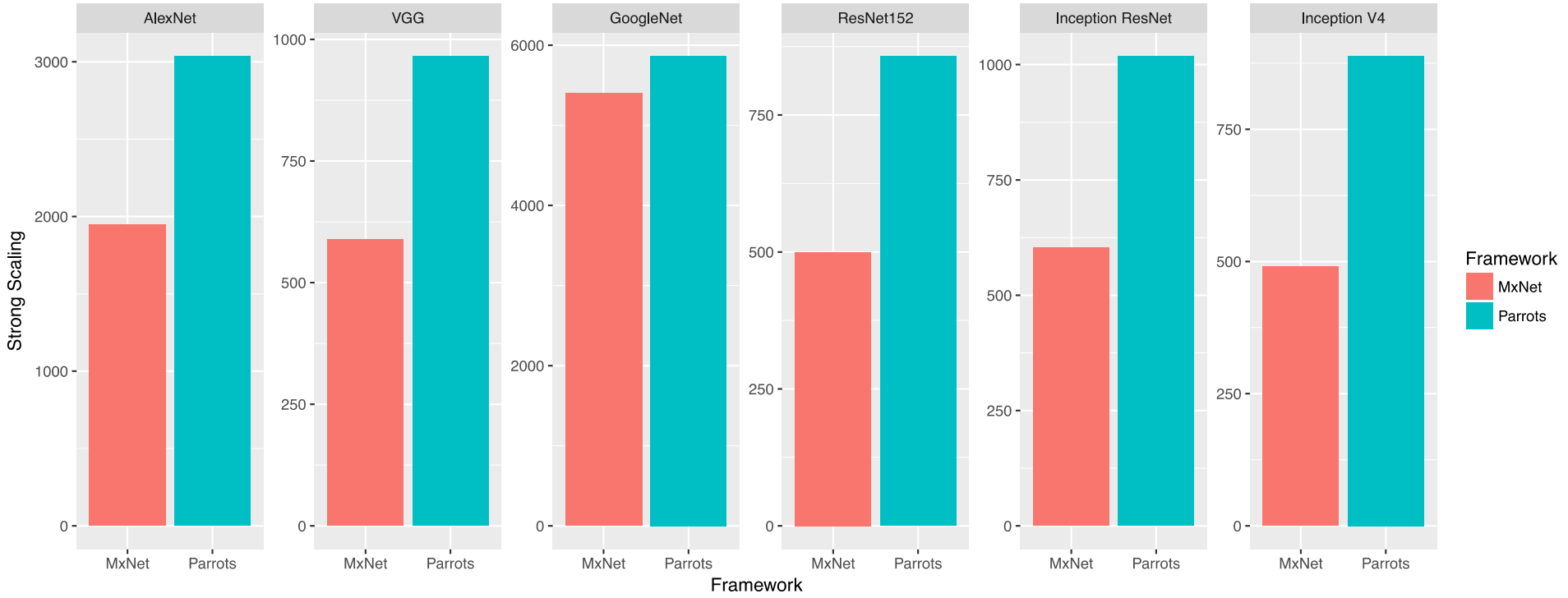
Scalability



Comparison on Weak Scaling



Comparison on Strong Scaling



Behind the Scene

Cross-module Parallelization

Light-weight Computing Flow

Intermediate Byte-code compilation

Structure-aware Memory reuse

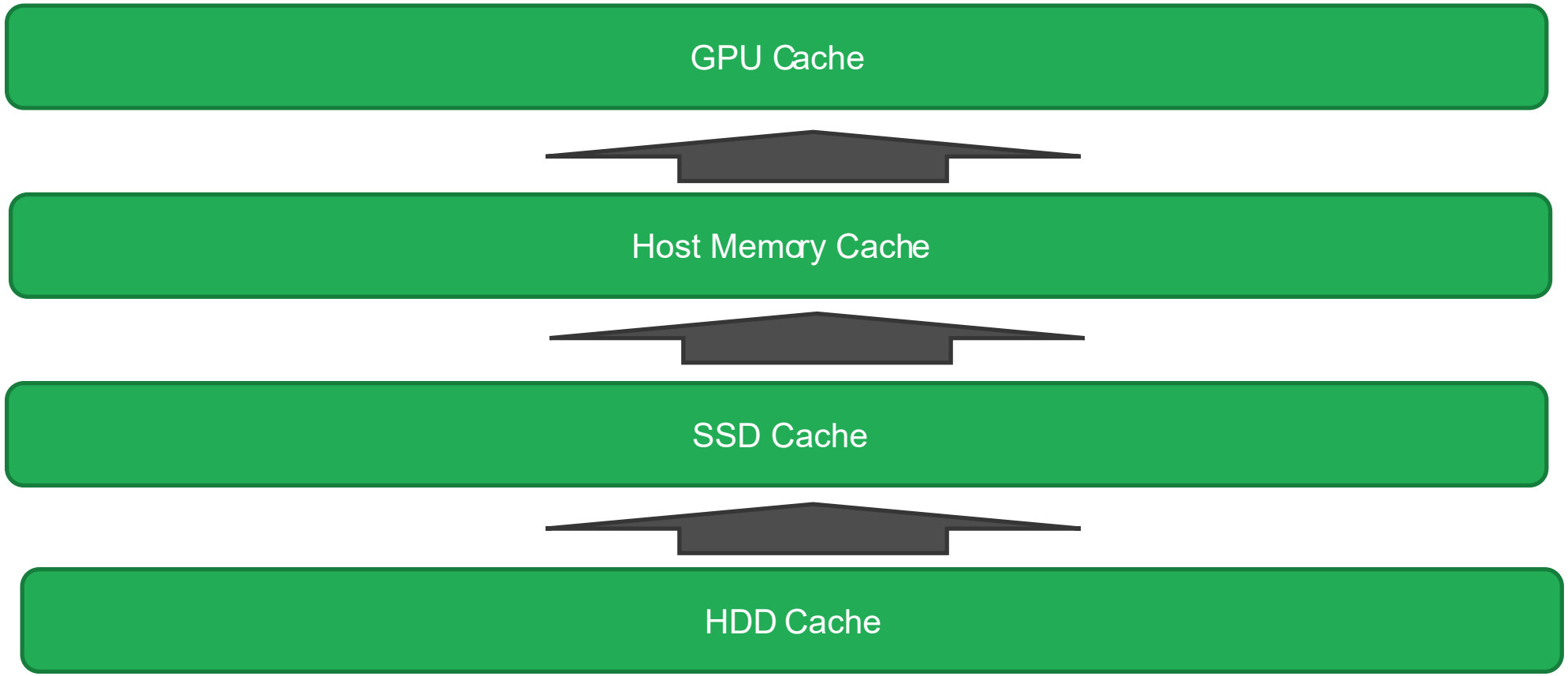
Parrots

Deep Learning Virtual Machine

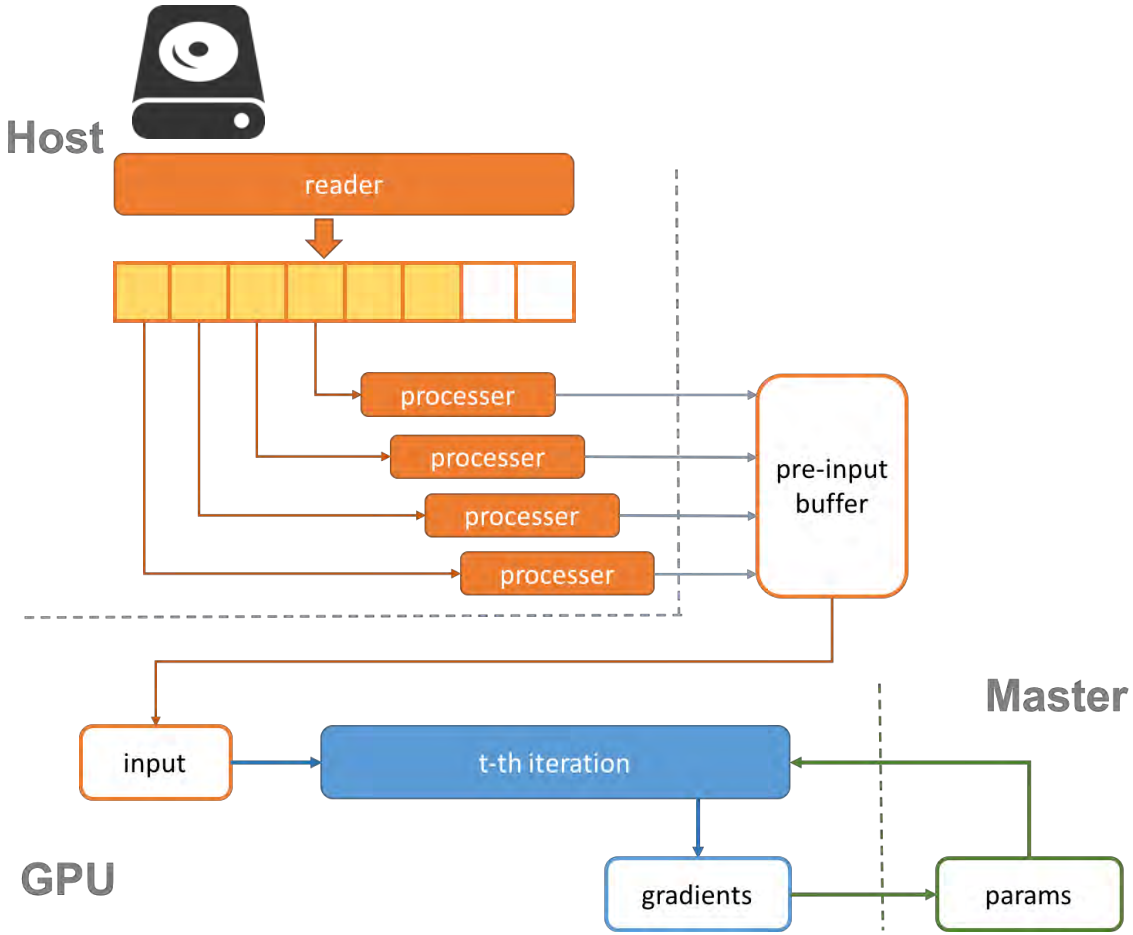
Distributed Dynamic Cache

Adaptive Learning Control

Multi-level Cache



Serial Loading Parallel Processing (SLPP)

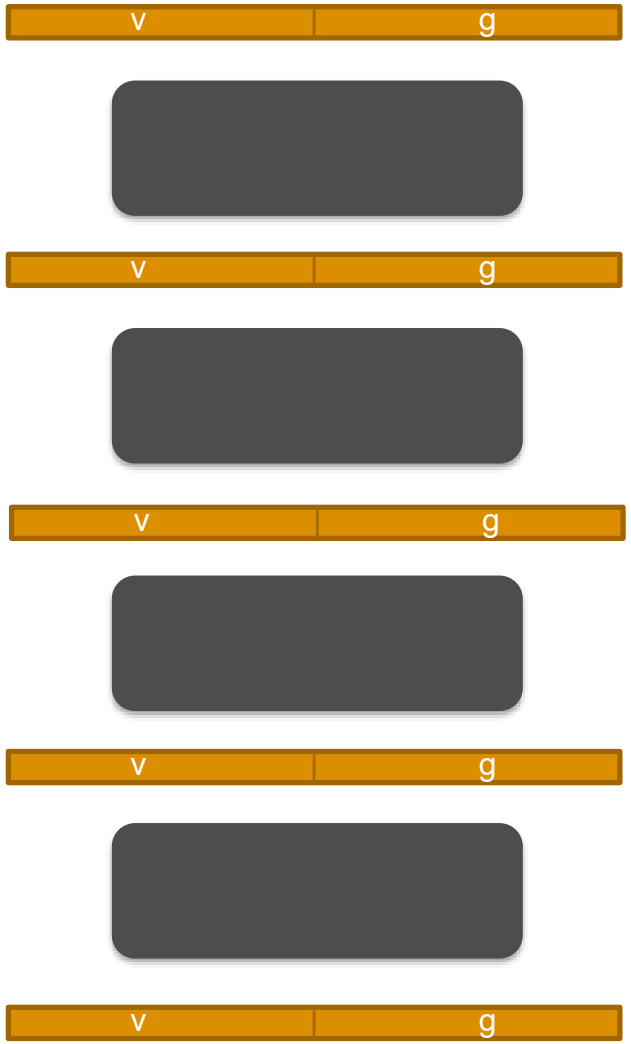


Rationales

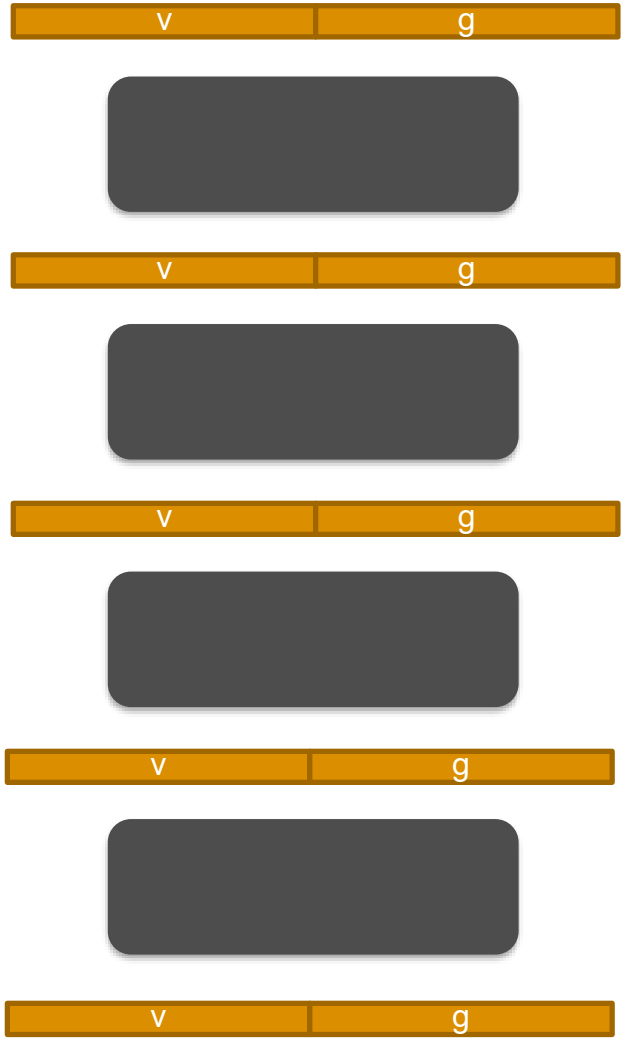
- Overlap data loading, processing, and on-GPU computation via proper scheduling.
- Serial I/O often yields higher throughput, while in-memory processing can be done in parallel.
- Maximum utilization of GPU, CPU, and I/O.

Dependency-based Memory Reuse

Conventional

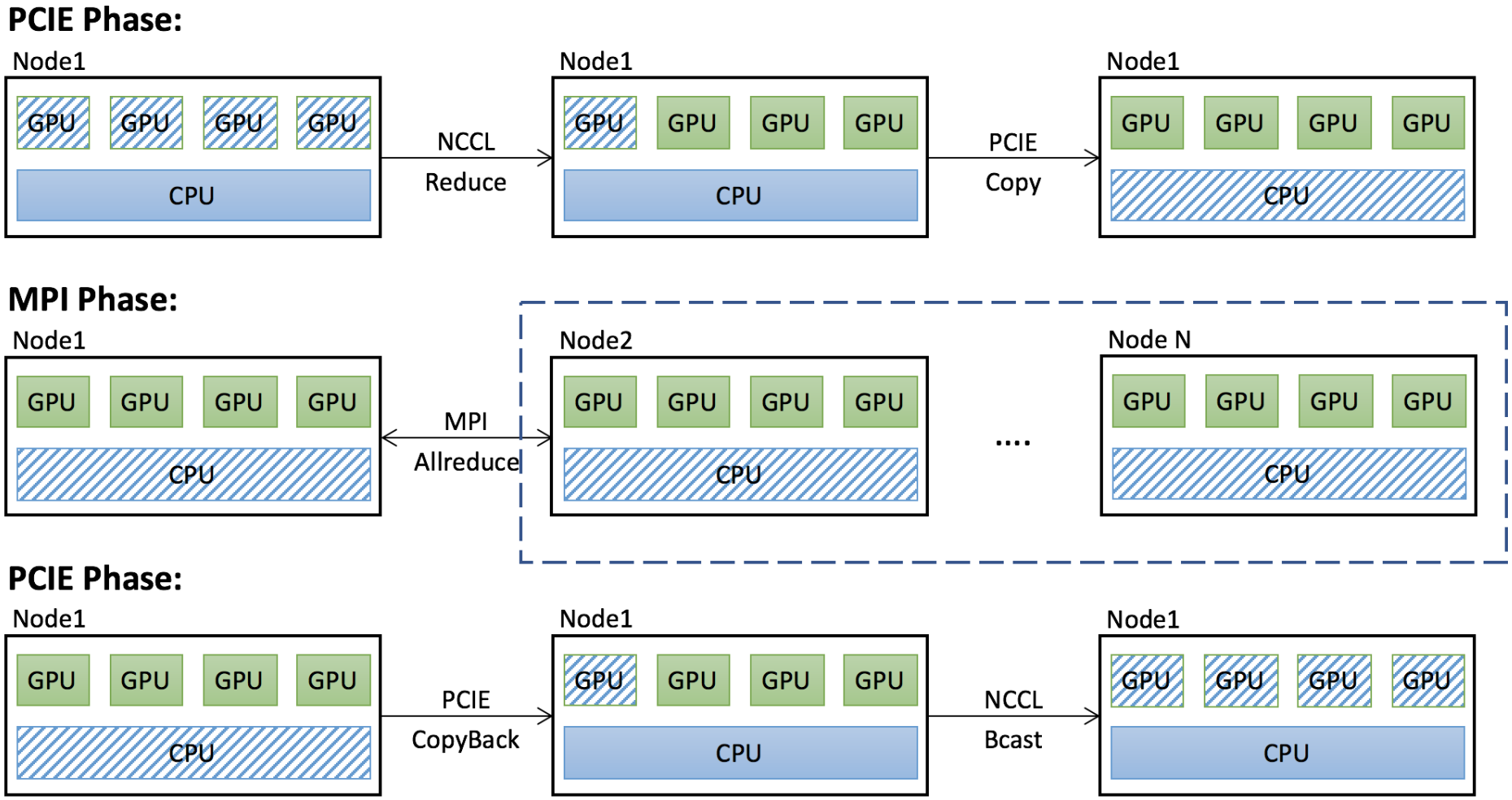


Ours



Two-level Communication

To achieve high scalability



Feature Matrix

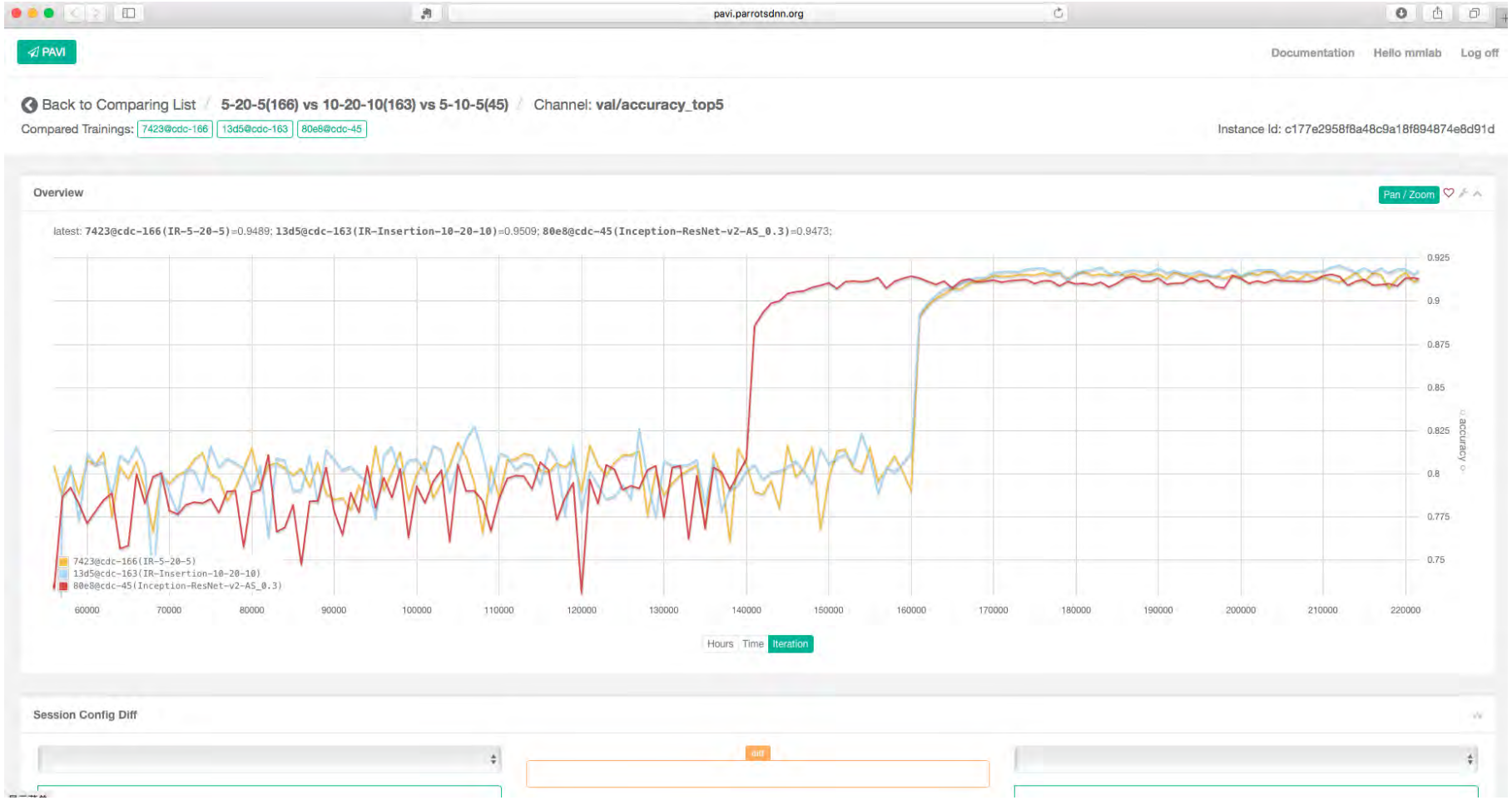
	Parrots	Caffe	TensorFlow	Chainer
Multi-GPU support	Y	Y	L	L
Distributed training (multi-nodes)	Y	N	L	N
Model parallelism	Y	N	L	L
Concurrent feeding	Y	N	Y	N
Parallel data preprocessing	Y	N	Y	N
RNN support	Y	N	Y	Y
Variable input size	Y	Y	Y	Y
Dynamic network structure	Y	N	Y	Y
Partial flow execution	Y	N	Y	N
Block composition	Y	N	Y	Y
Customized layer	Y	Y	Y	Y
Customized updating policy	Y	N	Y	Y
Interoperability with Caffe	Y	Y	N	N

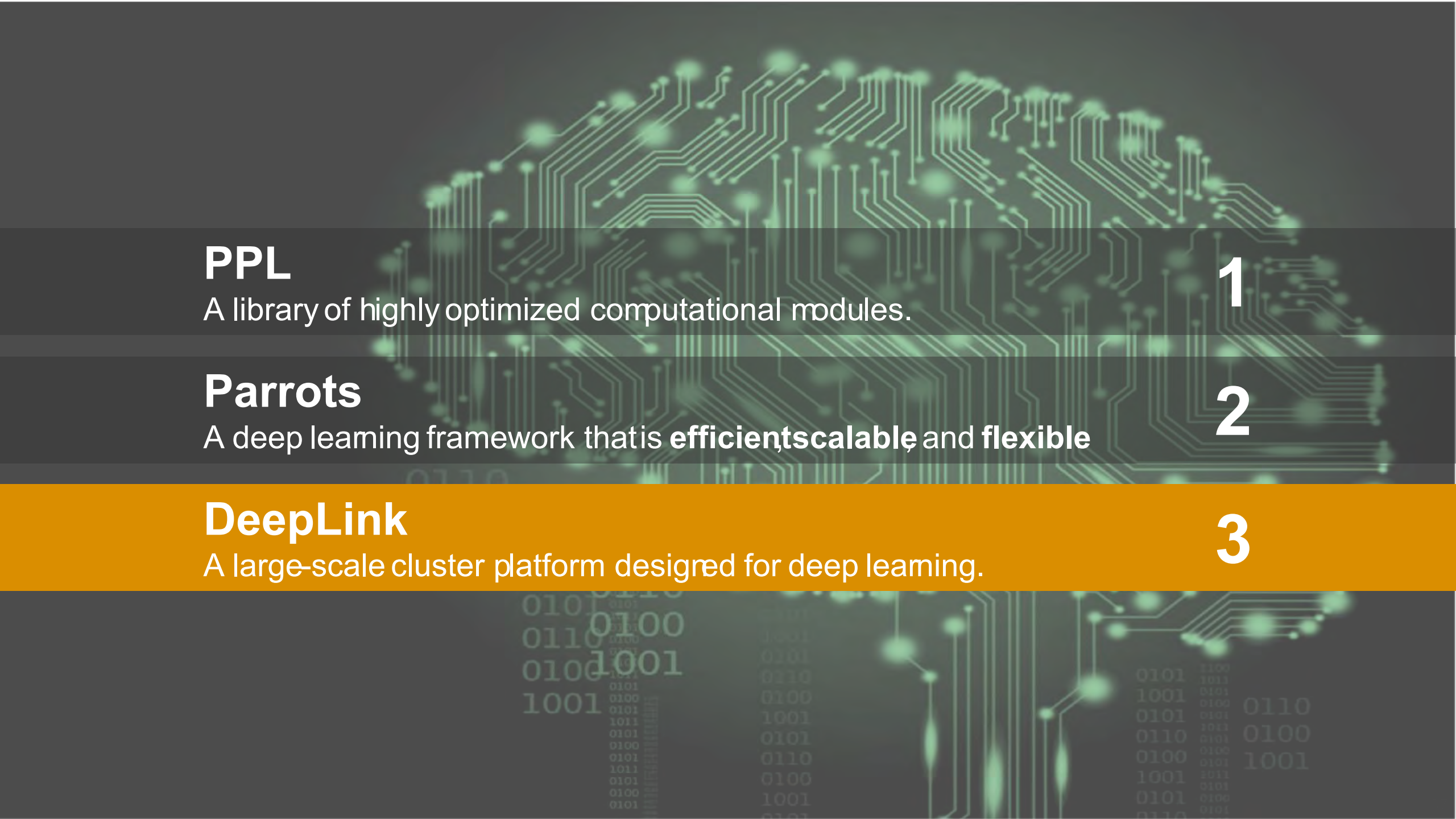
L: low-level support. Non-trivial coding is required to make it happen.

Overall Comparison

	Parrots	Caffe	TensorFlow	Chainer
Efficiency	High	Fair	Low	Fair
Scalability	High	Low	High	Low
Extensibility	High	Low	High	Fair
Productivity	High	Low	Fair	High

Web UI for Training Monitor




PPL
A library of highly optimized computational modules.
1
Parrots
A deep learning framework that is efficient, scalable, and flexible
2
DeepLink
A large-scale cluster platform designed for deep learning.
3

DeepLink Clusters

Designed for Deep Learning

Software Hardware Co-design	Maximize respectivestrengths while ensuring optimal cooperation.
High-performance Hardware	• High speed interconnects • High performance GPU computing • Efficient distributed storage
Customized Middlewares	• Distributed storage & cache system (optimized for small files) • Distributed deep learning framework • Task scheduling & monitoring

Challenges

Interconnects at multiple levels

- GPUs, Nodes, Sub-networks

Distributed data

- Random access becomes particularly difficult

Scale vs. Stability

- Failures of individual nodes/links

Human resources

- Engineers who understand both Deep Learning & HPC are difficult to come by

Overall Architectures