ADVANCED NETWORK SEMANTICS FOR CONVERGED HPC, AI AND ANALYTICS WORKLOADS

Sayantan Sur, Intel

ExaComm Workshop held in conjunction with ISC 2018

Legal Disclaimer & Optimization Notice

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.

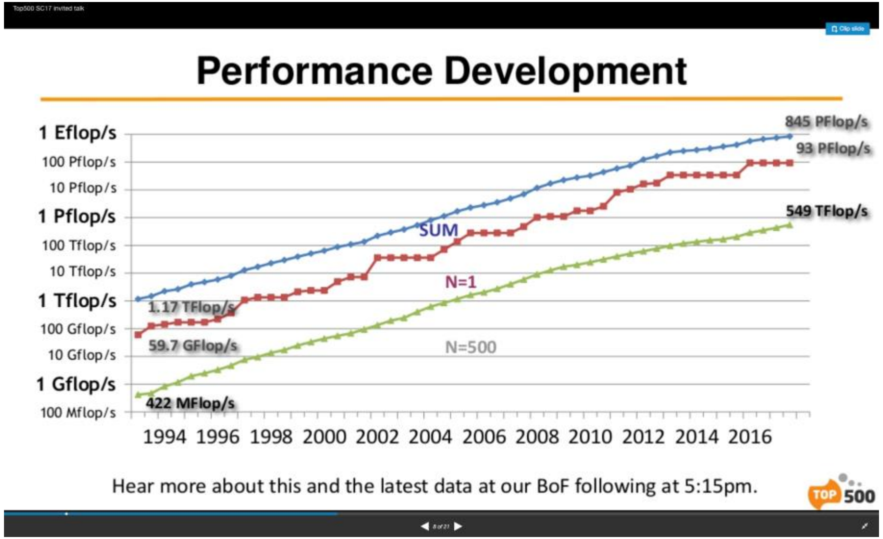
INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS". NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

What is HPC?



Performance Development

Hear more about this and the latest data at our BoF following at 5:15pm.

Courtesy Top500.org

Inputs:

- Better processors
- Better fabric
- Better system design

Output:

- Increasing overall performance

HPC is an activity characterized by the workload's nature, intent and response to scale

Can other problems use high performance?

IoT and Analytics

- Total data in the world doubling every two years[1]
- Vast amounts of data unutilized
  - “only 1 percent of data from an oil rig with 30,000 sensors is examined”[2]
- Simply adding more commodity compute / fabric doesn’t help – need high performance!

Artificial Intelligence

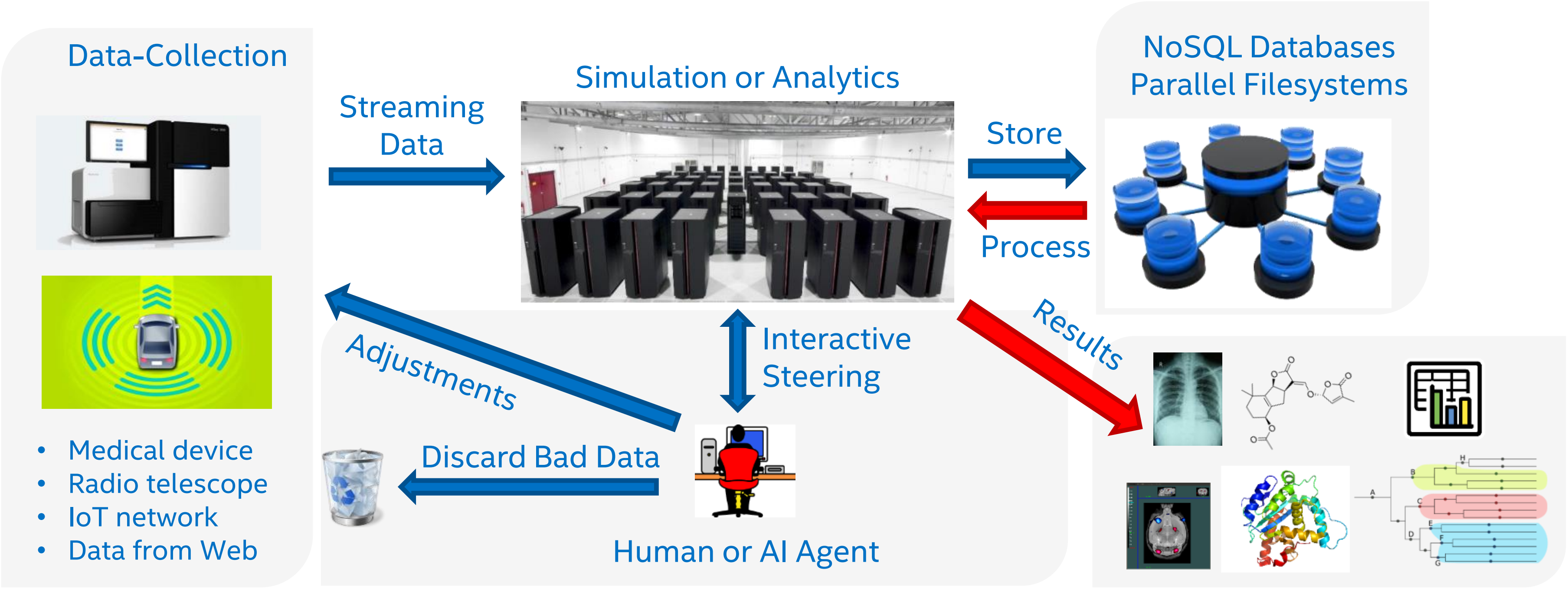
- Data Parallel Deep Learning already utilizing HPC Fabrics / techniques
- “Deep Learning at 15PF” on Intel Xeon Phi[3]
- Model Parallel imposes challenging memory / bandwidth limits

[1] https://insidebigdata.com/2017/02/16/the-exponential-growth-of-data/

[2] McKinsey Global Institute report on The Internet of Things: Mapping the value beyond the hype

[3] https://arxiv.org/pdf/1708.05256.pdf

Converged workflows could create new value



Data-Collection

- Medical device
- Radio telescope
- IoT network
- Data from Web

Streaming Data

Simulation or Analytics

Store

Process

NoSQL Databases
Parallel Filesystems

Results

Interactive Steering

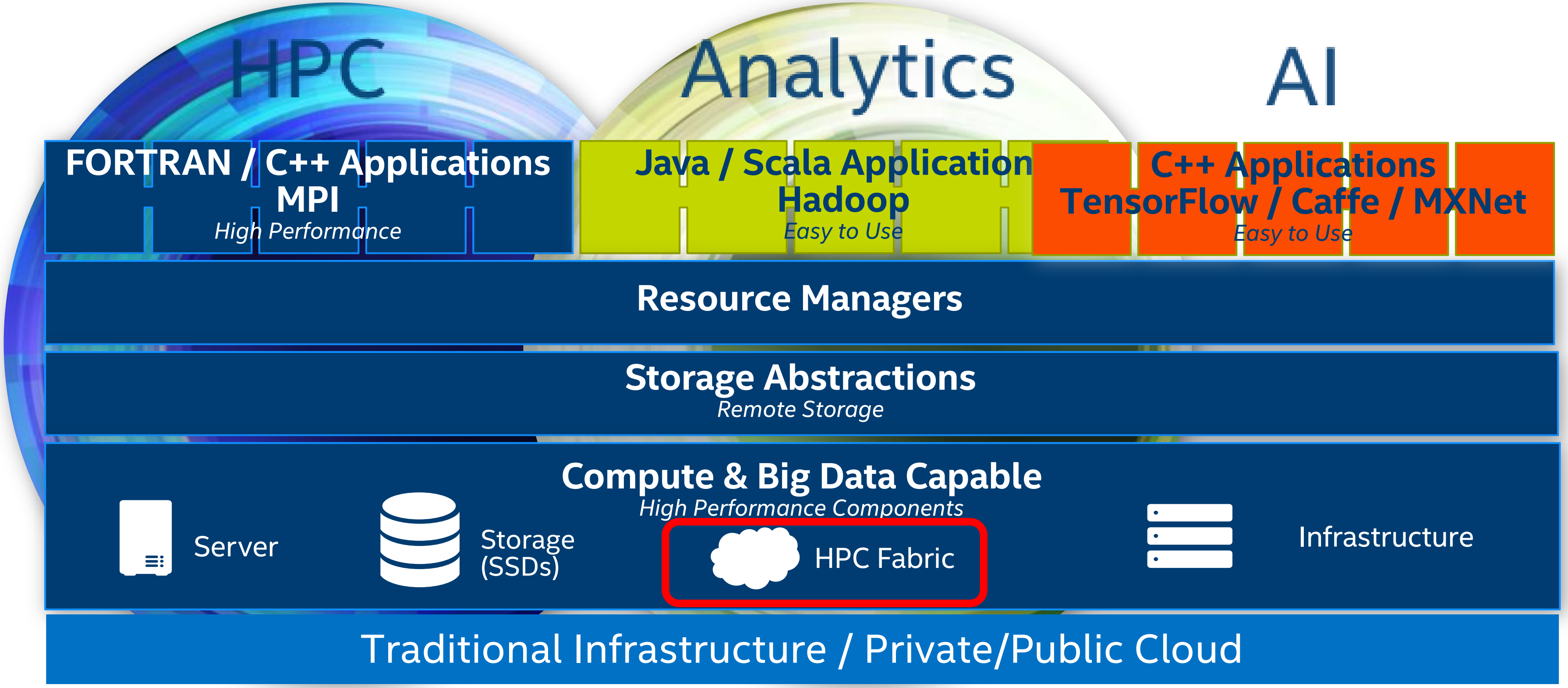
Adjustments

Discard Bad Data

Human or AI Agent

Fabric must comprehend multiple application domains to serve a converged workflow

Example Desired State: Unified Architecture



HPC

Analytics

AI

FORTRAN / C++ Applications MPI
High Performance

Java / Scala Application Hadoop
Easy to Use

C++ Applications TensorFlow / Caffe / MXNet
Easy to Use

Resource Managers

Storage Abstractions
Remote Storage

Compute & Big Data Capable
High Performance Components

Server

Storage (SSDs)

HPC Fabric

Infrastructure

Traditional Infrastructure / Private/Public Cloud

Requirements on Fabric and Fabric Software

Hardware + Software: Support a wider set of fabric users than MPI, PGAS, File systems …

Hardware: Accelerate new communication paradigms emerging in converged workflows

Software: Provide semantic abstractions that enable communication offload while enhancing portability

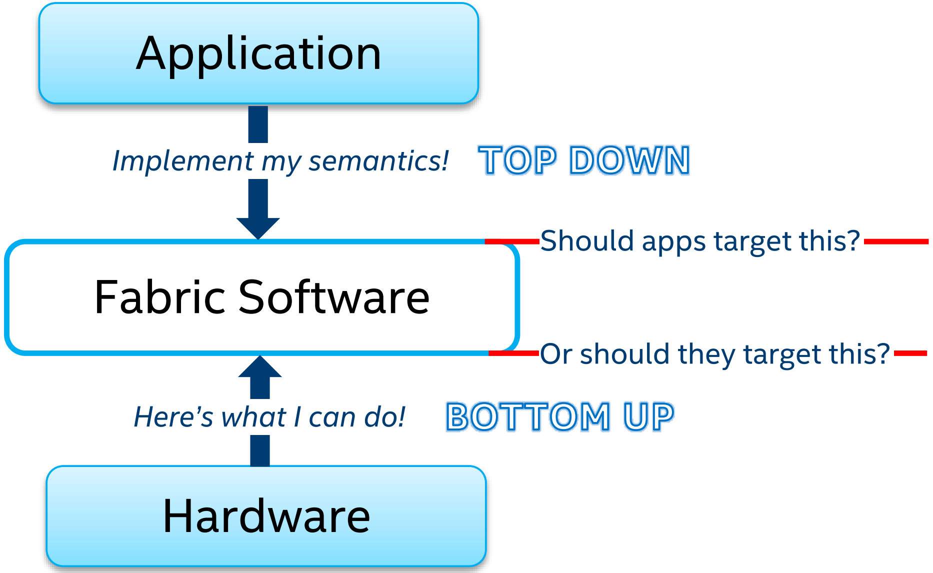
How can Open Fabrics Interface (OFI) help?

Enable “software leads hardware”

- Same abstractions on multiple fabrics, regardless of feature set
- Carefully defined semantics - net overheads remains the same even when hardware assists are unavailable

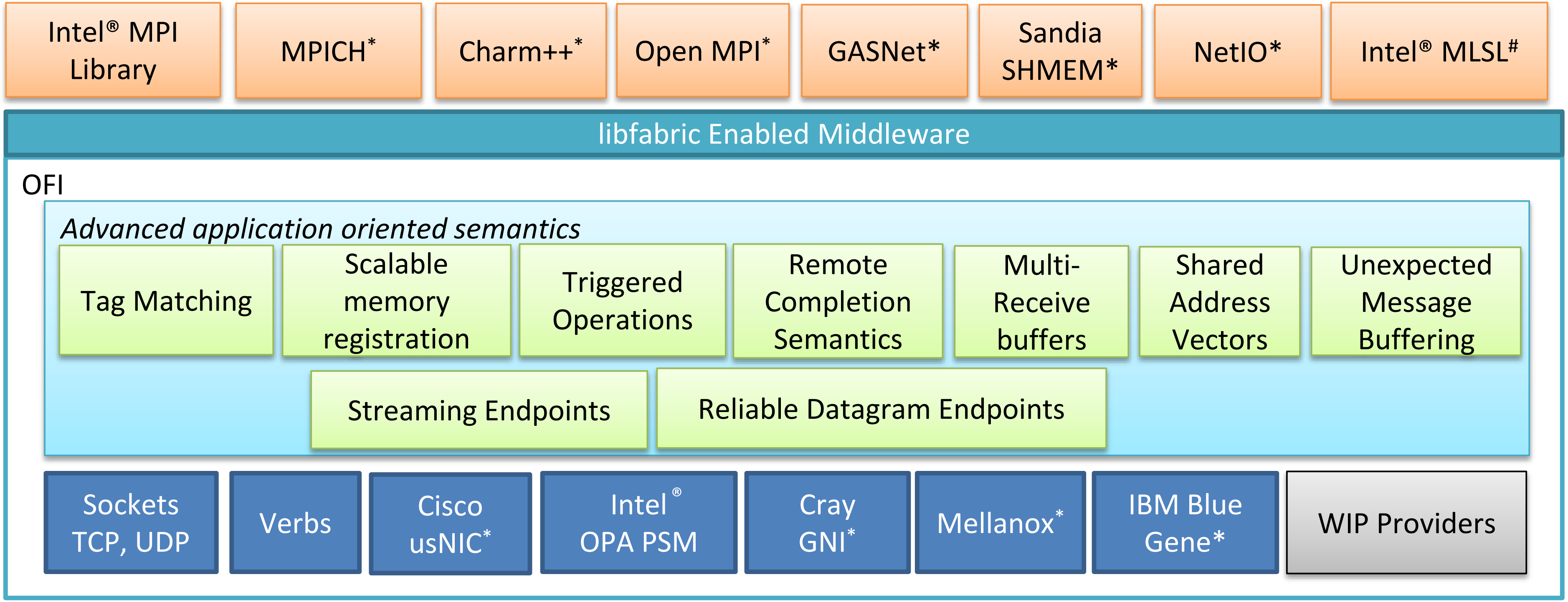
Interfaces aligned to a variety of use models – HPC (tag-matching, RMA), Client-Server (connected, datagram)

Powerful network and service discovery features



OFI – State of the Union

OFI Insulates applications from wide diversity of fabrics underneath



Intel® MPI Library | MPICH* | Charm++* | Open MPI* | GASNet* | Sandia SHMEM* | NetIO* | Intel® MLSL#

libfabric Enabled Middleware

OFI

Advanced application oriented semantics

Tag Matching | Scalable memory registration | Triggered Operations | Remote Completion Semantics | Multi-Receive buffers | Shared Address Vectors | Unexpected Message Buffering

Streaming Endpoints | Reliable Datagram Endpoints

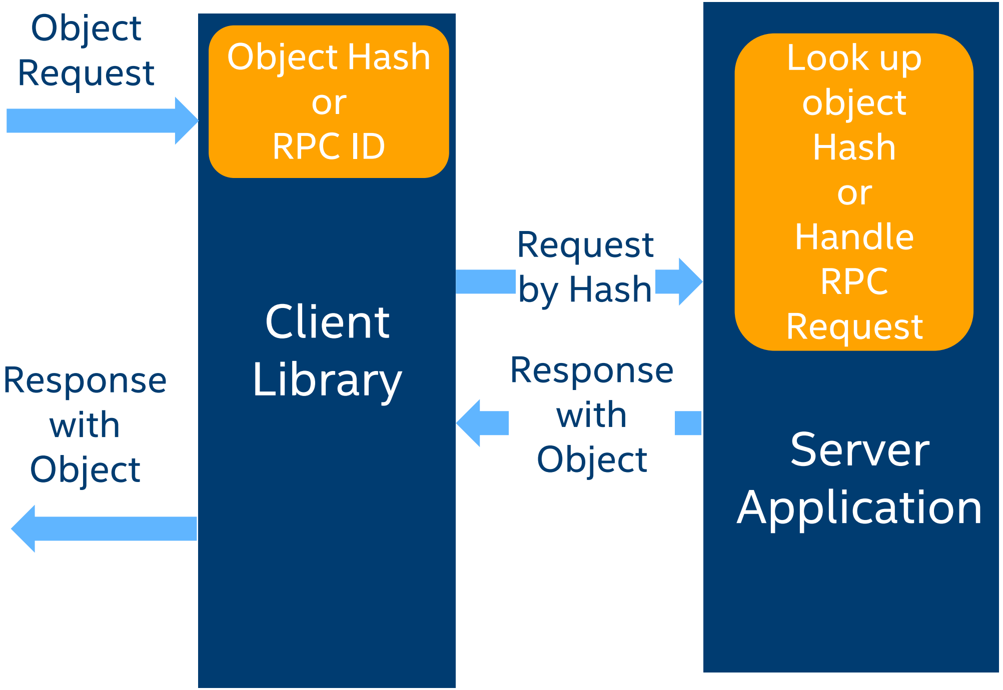
Sockets TCP, UDP | Verbs | Cisco usNIC* | Intel® OPA PSM | Cray GNI* | Mellanox* | IBM Blue Gene* | WIP Providers

# Exploration

* Other names and brands may be claimed as property of others

What new communication paradigms in Analytics could be accelerated?

RPC / Distributed Object Store Paradigms



Commonly occurring paradigm in Analytics middleware

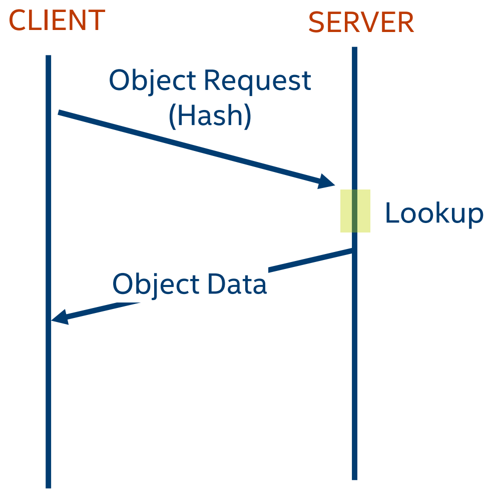
- NoSQL databases
- Spark

Multiple RPC libraries available

- gRPC, Netty, etc.

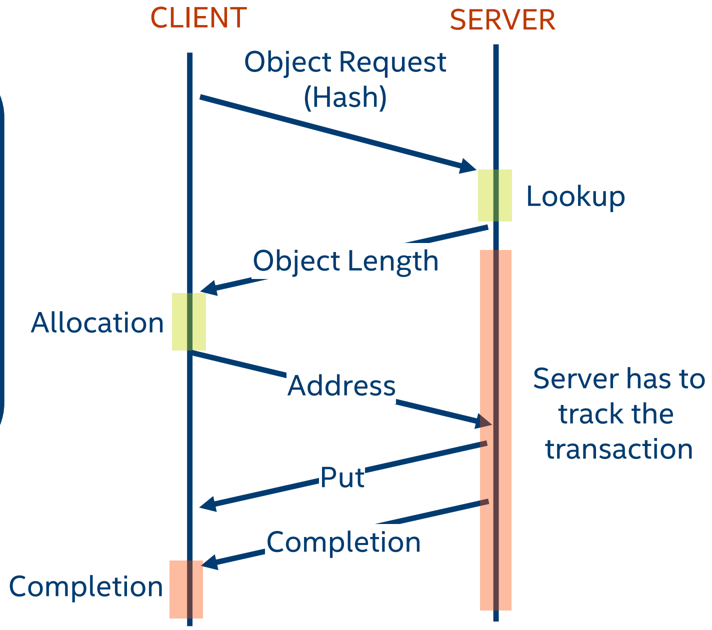
How to express RPC semantics on HPC Fabrics which have rich offloads?

RDMA Doesn’t quite fit the RPC Model



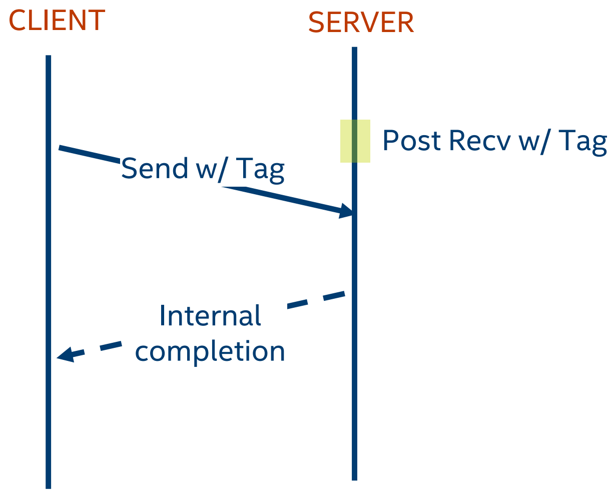
When objects are small

Does this look familiar to anyone?



When objects are big

Traditional Tag matching doesn't fit either!



MPI-style Tag Matching Model

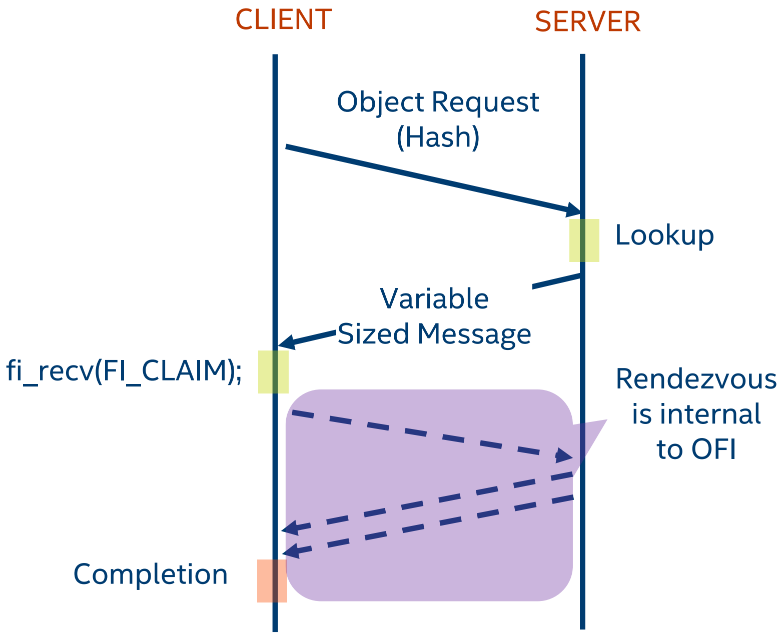
MPI use model

- Apps encouraged to pre-post recv
- Sender knows size of data
- Receiver knows max size of data
- Match order strictly defined

Distributed Object store use model

- Object Request may arrive at any time
- Object size not known until lookup is performed on the hash
- Ordering might be relaxed

A Better Use Model with OFI 1.7



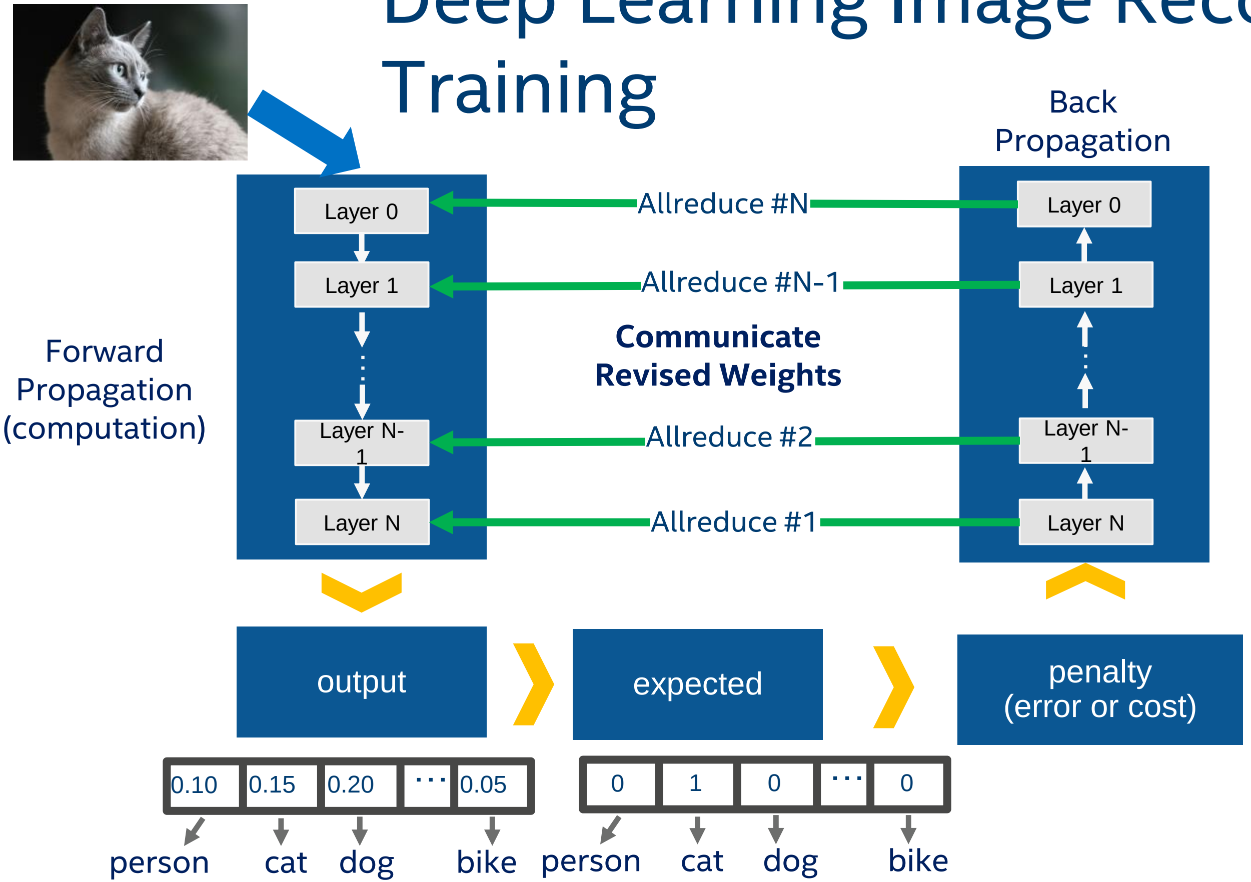
Variable Sized Messages

- When receiver doesn't know size of object prior to communication
- Easy to use when message sizes vary greatly
- Alleviates buffer management
- Removes application level rendezvous

➢ Can leverage tag matching if available

What are the new communication paradigms in AI (Deep Learning)?

Deep Learning Image Recognition Training



Back Propagation

Forward Propagation (computation)

Layer 0

Layer 1

Layer N-1

Layer N

Allreduce #N

Allreduce #N-1

Communicate Revised Weights

Allreduce #2

Allreduce #1

output

expected

penalty (error or cost)

0.10	0.15	0.20	···	0.05
person	cat	dog		bike

0	1	0	···	0
person	cat	dog		bike

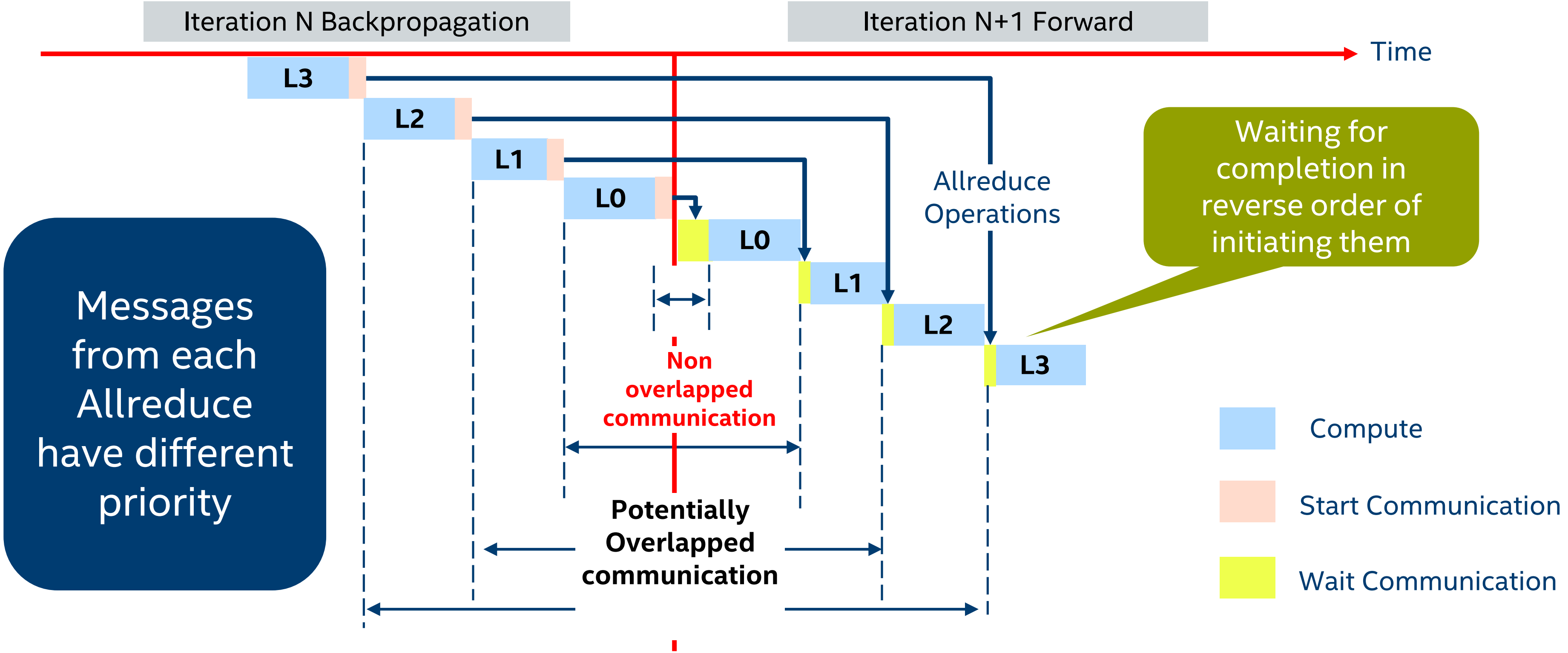
Data parallel mode, the model is replicated on each node

Model is further split into individual layer of neurons

Computation / layer is fixed

Communication / layer is fixed

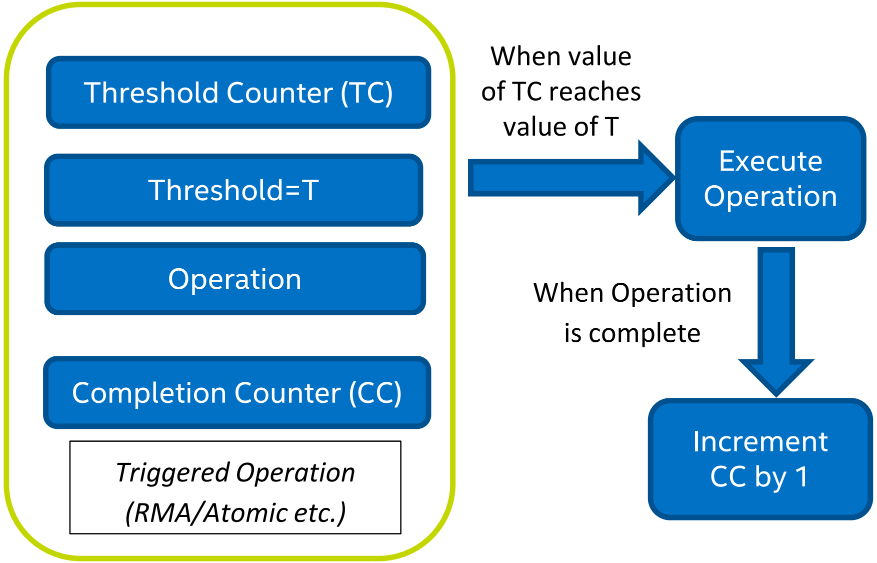
Performance depends on Overlap



Waiting for completion in reverse order of initiating them

Messages from each Allreduce have different priority

Triggered Operations


Threshold Counter (TC)
Threshold=T
Operation
Completion Counter (CC)
Triggered Operation (RMA/Atomic etc.)
When value of TC reaches value of T
Execute Operation
When Operation is complete
Increment CC by 1

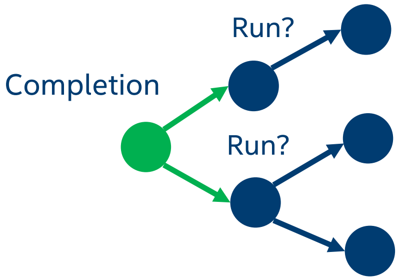
Offloaded Communication

- Trigger op when a condition is met
- Useful to advance a pre-computed schedule

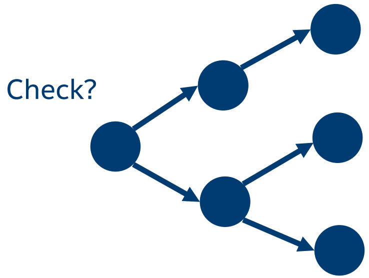
Designed for MPI Collectives

- Latency reduction for blocking collectives
- Increase overlap for non-blocking collectives

How to handle multiple schedules?



Schedule #1



Schedule #2

Completion of one operation could trigger multiple

Should any runnable operation be run?

Should we check if some other op completed so we run those dependencies instead?

Choice of semantic can have performance / implementation implications

Active area of research

Summary

Requirements on HPC Fabric – hardware and software are increasing

Many new software frameworks must access the HPC fabric

Analytics frameworks evolved outside of HPC

Artificial intelligence frameworks can tax HPC fabrics with new semantics

Software components will evolve at their own pace, on a variety of hardware

OFI can help in aligning the software and hardware requirements, enabling software to lead hardware

Completely open and free participation in OFI development

http://libfabric.org/

intel®

experience
what's inside™