UCX Development in Huawei

Alex Margolin

UCF Annual Workshop, December 2020
1. Huawei’s HPC activities – Cause and effect

2. Zooming in on UCX (*Today*)

3. Huawei’s roadmap for UCX (*Tomorrow*)

4. Teasers from other talks we’ll give this week
A Comprehensive HPC Solution – On All Levels

*Supports TaiShan/x86 hybrid scheduling.
A-Z Server Solution by Huawei

Intelligent Management Chip
Ascend AI Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Intelligent Management Chip
Huawei Atlas 300 AI Accelerator
Kunpeng Processor
Intelligent NIC Chip
Huawei Intelligent NIC
Huawei Intelligent NIC Chip
Huawei ES3000 V5 SSD
Intelligent SSD Controller Chip
Huawei ES3000 V5 SSD
Product Evolution

High computing-efficient
Provide Huawei Kunpeng Processor being compatible with ARM, TaiShan server and efficient computing solutions to help reduce TCO and time to market

Solid and reliability
Huawei-developed processor cores and server chips 17 years of computing innovation to guarantee high quality

Openness and innovations
Open platform based on mainstream software and hardware in the industry
Build the Kunpeng ecosystem to lay a solid foundation for intelligent computing

Hi1612
1st 64-bit server class ARM Processor
2016

Kunpeng 916
1st ARM Processor supporting multiple sockets
2019

Kunpeng 920
1st 7nm Data center Processor

K3
1st ARM mobile Processor
2014

1st ASIC chip for optical networks
1st ARM wireless base station
1991

2005

2009
Latest CPU – Huawei Kunpeng 920

- **High Performance**: 930+ SPECint®_rate_base2006 estimated score
- **High Bandwidth**
  - Memory bandwidth: 2.4x ↑
  - I/O bandwidth: 1.7x ↑
  - Network bandwidth: 10x ↑
- **High Integration**: 4 chips in 1 (CPU, south-bridge, NIC, SAS controller)
- **High Efficient Computing**: 35% ↑

*Tested in Huawei lab, comparison between Kunpeng 920-6426 and last generation Kunpeng 916. Results may vary in different environments

7 nm process | 64 cores | 8 memory controllers | PCIe 4.0 & 100GE
Flexible Form-Factor

<table>
<thead>
<tr>
<th>Model</th>
<th>2280 Balanced Model</th>
<th>5280 Storage Model</th>
<th>X6000 High-Density Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>For diversified workloads</td>
<td>For diversified workloads</td>
<td>For diversified workloads</td>
<td>For diversified workloads</td>
</tr>
<tr>
<td>2U Rack Server</td>
<td>4U Rack Server</td>
<td>2U 4-Node Server</td>
<td>HPC compute node main product</td>
</tr>
<tr>
<td>2-socket</td>
<td>2-socket</td>
<td>2-socket per node</td>
<td></td>
</tr>
<tr>
<td>32*DDR4-2933 MHz</td>
<td>32*DDR4-2933 MHz</td>
<td>16*DDR4-2933 MHz</td>
<td></td>
</tr>
<tr>
<td>27<em>2.5&quot; HDDs or 16</em>2.5&quot; NVMe SSDs</td>
<td>40*3.5&quot; HDDs</td>
<td>6*2.5&quot; HDDs or NVMe SSDs</td>
<td></td>
</tr>
<tr>
<td>CCIX, 8*PCIe 4.0</td>
<td>CCIX, 8*PCIe 4.0</td>
<td>CCIX, 2*PCIe 4.0</td>
<td></td>
</tr>
<tr>
<td>GE / 10GE / 25GE</td>
<td>GE / 10GE / 25GE</td>
<td>10GE / 25GE / 100GE RoCE</td>
<td></td>
</tr>
<tr>
<td>Air-cooled</td>
<td>Air-cooled</td>
<td>Air-cooled</td>
<td>Liquid-cooled</td>
</tr>
</tbody>
</table>

2280 Balanced Model:
- 2U Rack Server
- 2-socket
- 32*DDR4-2933 MHz
- 27*2.5" HDDs or 16*2.5" NVMe SSDs
- CCIX, 8*PCIe 4.0
- GE / 10GE / 25GE
- Air-cooled

5280 Storage Model:
- 4U Rack Server
- 2-socket
- 32*DDR4-2933 MHz
- 40*3.5" HDDs
- CCIX, 8*PCIe 4.0
- GE / 10GE / 25GE
- Air-cooled

X6000 High-Density Model:
- 2U 4-Node Server
- 2-socket per node
- 16*DDR4-2933 MHz
- 6*2.5" HDDs or NVMe SSDs
- CCIX, 2*PCIe 4.0
- 10GE / 25GE / 100GE RoCE
- Air or liquid-cooled
Breaking Down to Components

Powerful computing
- 64-core 2.6 GHz high-performance processor

High memory bandwidth
- 8 memory channels

High I/O throughput
- PCIe 4.0
  - Twice the PCIe 3.0 bandwidth

Software and hardware synergized tuning
- Huawei-developed compilers and math libraries
- Huawei-developed MPI
Classifying by Arithmetic Intensity

- Arithmetic intensity means ratio of (Arithmetic instructions)/(Off-chip memory operands).
- Lower arithmetic means memory-bound.
- Higher arithmetic means compute-bound.

Using Roofline model analysis, a large number of HPC algorithms and applications are memory-bound. TaiShan HPC targets memory-bound applications, such as CAE/CFD, weather, life sciences, and oil & gas.
# Applications Sample – by Arithmetic Intensity*

*obviously, this may depend on the input and settings

<table>
<thead>
<tr>
<th>Application</th>
<th>Scenario</th>
<th>Numerical Method</th>
<th>Arithmetic Intensity</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCM</td>
<td></td>
<td>Navier-Stokes</td>
<td>0.14</td>
</tr>
<tr>
<td>OpenFoam</td>
<td>CFD</td>
<td>Finite Volumes – Finite Element</td>
<td>0.13</td>
</tr>
<tr>
<td>Turbine</td>
<td></td>
<td>DNS</td>
<td>0.56</td>
</tr>
<tr>
<td>MHD – FDM</td>
<td>Magneto Hydro Dynamics</td>
<td>Finite Difference Method</td>
<td>0.33</td>
</tr>
<tr>
<td>MHD - Spectral</td>
<td>Magneto Hydro Dynamics</td>
<td>Pseudo Spectral Method</td>
<td>0.45</td>
</tr>
<tr>
<td>QSFDM</td>
<td>Seismology</td>
<td>Spherical 2.5D FDM</td>
<td>0.46</td>
</tr>
<tr>
<td>SEISM3D</td>
<td></td>
<td>Finite Difference Method</td>
<td>0.47</td>
</tr>
<tr>
<td>Barotropic</td>
<td>Ocean Circulation Model</td>
<td>Shallow Water Model</td>
<td>0.51</td>
</tr>
<tr>
<td>BQCD</td>
<td>High-Energy-Physics</td>
<td>Hybrid Monte-Carlo</td>
<td>0.45 (DP), 0.9 (SP)</td>
</tr>
<tr>
<td>B-CALM</td>
<td>Electro-Magnetic Sim.</td>
<td>Finite Difference time-domain</td>
<td>0.3 (SP)</td>
</tr>
<tr>
<td>WRF</td>
<td>Weather Forecast model</td>
<td>Stencil code</td>
<td>0.5-1.5</td>
</tr>
<tr>
<td>HEPSPEC</td>
<td>SPEC2006 selection for HEP (CERN)</td>
<td>NAMD, DEALII, SOPLEX, POVRAY, OMNETPP, ASTAR, XALANCBMK</td>
<td>&gt;=0.5</td>
</tr>
<tr>
<td>Gromacs</td>
<td>Molecular dynamics package</td>
<td>Bennett Acceptance Ratios</td>
<td>&lt;1</td>
</tr>
<tr>
<td>KKRNano</td>
<td>Nanotechnology</td>
<td>Korringa-Kohn-Rostoker</td>
<td>4 (DP)</td>
</tr>
</tbody>
</table>
Industry's First RoCE-based Online Computing Solution

In-network computing (INC):

- Basic principle: add a component in the RoCE switch to offload reduction operations.
- The RoCE-based software and hardware combination solution improves the performance by 30% to 50%.

<table>
<thead>
<tr>
<th>Reduction</th>
<th>PPN</th>
<th>HMPI (RoCE)</th>
<th>HMPI+ INC</th>
<th>INC Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>8Bytes</td>
<td>24</td>
<td>8.94</td>
<td>5.81</td>
<td>53.87%</td>
</tr>
<tr>
<td></td>
<td>96</td>
<td>11.36</td>
<td>8.87</td>
<td>28.07%</td>
</tr>
</tbody>
</table>

Applying INC for Reduction

- Latency (microseconds)
- Number of Node Processes

<table>
<thead>
<tr>
<th>Latency (microseconds)</th>
<th>Number of Node Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>96</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
</tr>
<tr>
<td>0</td>
<td>96</td>
</tr>
</tbody>
</table>

- Comparison between HWMPI and INC
Packets are like sheep... and it’s 64 herds!
Now #1 in I/O-500

- Capture user-experienced performance
- Reported performance is representative for:
  - applications with well optimized I/O patterns
  - applications with random-like workloads
  - workloads involving metadata small/objects

<table>
<thead>
<tr>
<th>#</th>
<th>institution</th>
<th>system</th>
<th>storage vendor</th>
<th>filesystem type</th>
<th>client nodes</th>
<th>client total proc</th>
<th>score</th>
<th>bw</th>
<th>md</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O-500 #1</td>
<td>Pengcheng Laboratory</td>
<td>Pengcheng Cloudbrain-II on Atlas 900</td>
<td>Pengcheng Laboratory</td>
<td>MadFS</td>
<td>255</td>
<td>18360</td>
<td>7043.99</td>
<td>1475.75</td>
<td>33622.19</td>
</tr>
<tr>
<td>I/O-500 #2</td>
<td>Intel</td>
<td>Wolf</td>
<td>Intel</td>
<td>DAOS</td>
<td>52</td>
<td>1664</td>
<td>1792.98</td>
<td>371.67</td>
<td>8649.57</td>
</tr>
<tr>
<td>I/O-500 #3</td>
<td>WekaIO</td>
<td>WekaIO on AWS</td>
<td>WekaIO</td>
<td>WekaIO Matrix</td>
<td>345</td>
<td>8625</td>
<td>938.95</td>
<td>174.74</td>
<td>5045.33</td>
</tr>
<tr>
<td>10-node #1</td>
<td>Pengcheng Laboratory</td>
<td>Pengcheng Cloudbrain-II on Atlas 900</td>
<td>Pengcheng Laboratory</td>
<td>MadFS</td>
<td>10</td>
<td>1440</td>
<td>1129.75</td>
<td>168.43</td>
<td>7578.06</td>
</tr>
<tr>
<td>10-node #2</td>
<td>Intel</td>
<td>Wolf</td>
<td>Intel</td>
<td>DAOS</td>
<td>10</td>
<td>420</td>
<td>758.71</td>
<td>164.77</td>
<td>3493.56</td>
</tr>
<tr>
<td>10-node #3</td>
<td>TACC</td>
<td>Frontera</td>
<td>Intel</td>
<td>DAOS</td>
<td>10</td>
<td>420</td>
<td>508.88</td>
<td>79.16</td>
<td>3271.49</td>
</tr>
</tbody>
</table>
HPC Solution Overview (when I joined Huawei...)

- Platform layer
  - Computing system (x86, TaiShan)
  - Next-generation NAS storage system
  - Network interconnect (low-latency technology)
  - Kunpeng processor
    - Cache coherent interconnect technology
  - High-Performance NAS Storage
    - 2020
  - RoCE
    - 2020

- Middleware
  - Resource management
  - Huawei MPI
  - Toolchain
  - Job scheduling

- Upper-layer applications
  - Weather & Ocean
  - Manufacturing
  - EDA
  - Life sciences
  - Oil & Gas
  - AI
  - Huawei MPI
  - Early 2020
  - Huawei compilers, math libraries, etc.
    - 2020
  - Huawei job scheduler
    - 2020
  - Hybrid cloud solution

- Huawei cluster management software
  - Early 2020

- HPC public cloud, private cloud and hybrid cloud systems
  - 2020
HPC Solution Overview Today

Platform layer
- Computing system (x86, TaiShan)
- Next-generation NAS storage system
- Network interconnect (low-latency technology)

Middleware
- Huawei MPI
- HPC application characterization, monitoring and tuning
- Resource management
- Job scheduling

Upper-layer applications
- Weather & Ocean
- Manufacturing
- EDA
- Life sciences
- Oil & Gas
- AI

Hyper-MPI
- Huawei compilers, math libraries, etc.
- Huawei job scheduler

Kunpeng processor
- Cache coherent interconnect technology
- Supports Burst Buffer NAS systems
- RoCE Technology and dedicated low-latency interconnect

Hybrid cloud and cloud bursting architecture
- HPC public cloud, private cloud and hybrid cloud systems

Huawei cluster management software
- HPC application characterization, monitoring and tuning
- Unified portal for HPC workflows
Outline

1. Huawei’s HPC activities – Cause and effect

2. Zooming in on UCX (*Today*)

3. Huawei’s roadmap for UCX (*Tomorrow*)

4. Teasers from other talks we’ll give this week
UCS (Services)

Pointer-Array

• Added *locked-pointer-array*, for thread-safety where applicable
• ~30-35% increase in iteration speed of *foreach()*: 1.46 ns $\rightarrow$ 1.02 ns
• 10% faster iteration on x86 (test-specific...), thanks to prefetching trickery

Usage? In UCG*: each column represents a context, and has a pointer-array to hold messages (equiv. to Tag-matching).

Statistics

• Apply filters to reported stats (*over UDP)
• To be continued...

Misc.: timer queue, aligned realloc., GCC fixes, etc.
One-to-many communication
- Expand UCX’s P2P focus to cover other types of communication
- Lots of changes are required for this: AM ID range, “exposing” internals, etc.

Kunpeng’s RoCE support
- Basically calls rdma-core, but there are some specifics (not public yet)

RoCE Reachability issues
Suppose you have 2 RoCE ports: do you...
a. set both IP addresses on the same subnet? (how to choose TX port? + Socket-Direct is unhappy)
b. set IP addresses on separate subnets? (some ports can’t talk to other ports!)

Workaround: ask for RoCEv1 (no direct way, use UCX_IB_GID_INDEX=0)
Not a workaround: Link aggregation (LAG)

Solution: have UCX choose RoCEv1 (PR #5581)
UCP (Protocol)

One-to-many communication

• Lots of changes are required for this: transport selection, extra API parameters, etc.

• Not much else, since we mostly use UCT directly – see next slide...

Q: Why use UCT directly?!

A: For several reasons, incl. mostly overhead considerations and too little control when using UCP. More on this during the UCG talk, later this week.
UCG (Groups)

- Huawei’s choice for collective operations.
- Will (eventually) support any collective type.
  *yes, including `MPIX_Neighbor_alltoallw_init()`
- Shamelessly (ab)uses UCX internals to work faster (incl. internal UCT stuff).
- Lives at https://github.com/openucx/xucg (X for eXperimental, a stable version exists)

- To be continued...
Open-MPI support

• Is there anything missing? Yes, we think so.

Code consolidation among UCX components (for starters)
• ./ompi/mca/coll/ucx
• ./ompi/mca/osc/ucx
• ./ompi/mca/pml/ucx
• ./ompi/mca/common/ucx
• ./opal/mca/common/ucx
• ./oshmem/...
• At least 2-3 additional components are in the planning!

• Better integration with the rest: MPI_T, hints when used by other components...
Open-MPI Component Consolidation

**Why?** Because on x6000 – we have 256 cores (x2 workers, x2 QPs - no good)

**How?**

- `ompi/mca/*/ucx` (pml, osc, coll)
  - (a) Submits requests, (b) delegates init, cleanup and progress*

- `ompi/mca/common/ucx`
  - (a) Fills ucp_params + ucg_params, (b) holds datatype context and free-lists

- `opal/mca/common/ucx`
  - The only layer with calls to UCX init, cleanup and progress*

*except during blocking calls on OMPI/mca level

**Another feature:** the same non-contiguous datatype does not get duplicated across PML and COLL MCAs.
Open-source policy

• Most of what we do is contributed... but not everything gets upstream.
  * Not just UCX / MPI – also KNEM, Spack, Open-PMIx / prrte, etc.

• There are a few exceptions:

1. Some hardware-specific code, especially for HW not on the market yet.

2. Integration code with some of Huawei’s proprietary software (e.g. Storage)

3. Code that didn’t pass Huawei’s code quality checklist YET.... takes time!
Outline

1. Huawei’s HPC activities – Cause and effect

2. Zooming in on UCX (*Today*)

3. Huawei’s roadmap for UCX (*Tomorrow*)

4. Teasers from other talks we’ll give this week
Integration with other parts of Huawei’s HPC solution

- When you have a hammer...

1. Working to get UCX adopted in other products (P2P and collectives).

2. Working with the compiler team and the performance teams to optimize the build on our processing elements (not just CPUs).

3. Working to accelerate UCX features we need

- This goes both ways – **UCX could use more input/hints**!
Example: Kunpeng-specific optimizations

The challenge: up to 64 cores per CPU, up to **256 per host**!
This effects:

- algorithm selection,
- resource constraints, (*remember OMPI code consolidation?)
- transport selection,
- ...
- **Application performance.**

Top features for high PPNs:
1. Shared-memory comm. enhancement (P2P and collectives),
2. Finer-grained topology awareness (even within a CPU),
3. Memory footprint reduction,
4. Good old testability at scale.
Concrete Example: Kunpeng-specific optimizations

Proposal: (external) user-space locking library

- Requires some research, to choose the best library to use, for example (publications):
  - “Compact NUMA-aware Locks” by Dice and Kogan (EuroSys ‘19)
  - “Scalable and practical locking with shuffling” by Kashyap et al. (SOSP ’19)
  - ... depends a lot on the architecture:

![Figure 6: Uncontested lock acquisition latency based on the location of the previous owner of the lock.](image)

<table>
<thead>
<tr>
<th>System</th>
<th>Opteron (2.1 GHz)</th>
<th>Xeon (2.13 GHz)</th>
<th>Niagara (1.2 GHz)</th>
<th>Tilera (1.2 GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>same die</td>
<td>one hop</td>
<td>two hops</td>
<td>same core</td>
</tr>
<tr>
<td>Loads</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modified</td>
<td>81</td>
<td>109</td>
<td>289</td>
<td>400</td>
</tr>
<tr>
<td>Owned</td>
<td>83</td>
<td>163</td>
<td>178</td>
<td>254</td>
</tr>
<tr>
<td>Exclusive</td>
<td>83</td>
<td>253</td>
<td>92</td>
<td>273</td>
</tr>
<tr>
<td>Shared</td>
<td>83</td>
<td>164</td>
<td>176</td>
<td>254</td>
</tr>
<tr>
<td>Invalid</td>
<td>136</td>
<td>327</td>
<td>355</td>
<td>492</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System</th>
<th>Opteron (2.1 GHz)</th>
<th>Xeon (2.13 GHz)</th>
<th>Niagara (1.2 GHz)</th>
<th>Tilera (1.2 GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>same M:M</td>
<td>one hop</td>
<td>two hops</td>
<td></td>
</tr>
<tr>
<td>Loads</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modified</td>
<td>83</td>
<td>273</td>
<td>315</td>
<td>431</td>
</tr>
<tr>
<td>Owned</td>
<td>244</td>
<td>255</td>
<td>286</td>
<td>291</td>
</tr>
<tr>
<td>Exclusive</td>
<td>83</td>
<td>191</td>
<td>271</td>
<td>315</td>
</tr>
<tr>
<td>Shared</td>
<td>246</td>
<td>255</td>
<td>286</td>
<td>296</td>
</tr>
</tbody>
</table>
Optimizations Galore!

• New ways to transfer data (even with existing interconnect HW),

• New ways to overlap computation and communication,

• New ways to detect and optimize common patterns,

• New ways to help applications.
A Bit about my team in Israel...

Appears to be a software engineering team, focused on networking...

Our secret weapon?
A Bit about my team in Israel...

Appears to be a software engineering team, focused on networking...

Our secret weapon?
- Half of my team are scientists (not computer scientists...), incl. professors, experts on computational-*: CFD, atmospheric models, molecular dynamics, etc.

P.S. is preaching to the choir... look for us at and
Outline

1. Huawei’s HPC activities – Cause and effect
2. Zooming in on UCX (*Today*)
3. Huawei’s roadmap for UCX (*Tomorrow*)
4. Teasers from other talks we’ll give this week
UCX counters in Score-P and Vampir (Day 3, 10:40 CT)
One-to-many UCT transports (Day 4, 9:00 CT)

4 queues needed:
1+2. The existing P2P queues, for control messages (e.g. Rendezvous).
3. Fanin, for collectives like reduce or gather.
4. Fanout, for collectives like bcast and scatter.
Until UCC is available - UCG status update (Day 4, 10:00 CT)