Justitia

Software Multi-Tenancy in Hardware Kernel-Bypass Networks

Yiwen Zhang¹, Yue Tan², Brent Stephens³, Mosharaf Chowdhury¹

University of Michigan¹, Princeton University², University of Illinois at Chicago³
Why use Kernel-Bypass Networks?

KBN provides low latency & high throughput by removing kernel from the data path.

Hardware-based KBN (e.g., RDMA): offloads data transfer to specialized NIC
  - Lower latency & better CPU efficiency
  - Widely used in cloud environments

Traditional Networking

Kernel-Bypass Networking
What if there are multiple applications sharing the same NIC?

- **Bandwidth-sensitive:** large messages (high bandwidth)
- **Throughput-sensitive:** small messages in batches (high message rate)
- **Latency-sensitive:** sparse small messages (low latency)

In hardware KBN, the OS has no control but to **rely on the NIC** to isolate among multiple applications.
Multiple on-NIC Resources

2 on-NIC resources:
• Link Bandwidth
• Execution Unit Throughput

➢ Resource contention

Lat does not saturate either resources, but can be blocked by large messages due to head-of-line (HOL) blocking
How Well Does Hardware KBN Provides Performance Isolation?
### Does RDMA Provide Isolation?

<table>
<thead>
<tr>
<th>Provides Isolation?[^1]</th>
<th>Latency</th>
<th>Throughput</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td><strong>NO</strong></td>
<td><strong>NO</strong></td>
<td><strong>NO</strong></td>
</tr>
<tr>
<td>Throughput</td>
<td>YES</td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>Latency</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

> Performance isolation is not guaranteed when bandwidth-sensitive apps are present.

[^1]: For each experiment we choose 2 types of applications to compete under InfiniBand/RoCE/iWarp. Applications include Mellanox Perftest and other real RDMA applications.
Anomalies in **Latency-Sensitive Apps**

- Latency-sensitive apps need isolation from bandwidth-sensitive apps

![Graph showing latency and bandwidth sensitivity](image)

- Latency-sensitivity: Lat App Alone vs. With B/W App

<table>
<thead>
<tr>
<th>Bandwidth (Gbps)</th>
<th>99th Percentile (us)</th>
<th>Lat App Alone</th>
<th>With B/W App</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 B/W</td>
<td>1.4</td>
<td></td>
<td>2.9</td>
</tr>
<tr>
<td>4 B/W</td>
<td></td>
<td></td>
<td>14.1</td>
</tr>
<tr>
<td>8 B/W</td>
<td></td>
<td></td>
<td>95.0</td>
</tr>
<tr>
<td>16 B/W</td>
<td></td>
<td></td>
<td>194</td>
</tr>
</tbody>
</table>
Anomalies in Throughput-Sensitive Apps

Throughput-sensitive apps also need isolation from bandwidth-sensitive apps.
What About Real Applications?

DARE (Latency-sensitive)

FaSST (Throughput-sensitive)
Justitia

Provides multi-tenancy support for hardware-based KBN

Provides multi-tenancy support in an
• efficient,
• scalable, and
• transparent manner

without modifying
• applications,
• operating systems, or
• hardware
Justitia Key Idea

Tenant directly talks to NIC
- arbitrary large messages
- many connections

Justitia:
Add a software layer between tenants and NIC
- creates a point of control
- prevent tenants from hogging NIC resources
1. **Justitia Daemon**
   - Performs latency monitoring and proactive rate management

2. **Justitia Shaper**
   - Enforces resource utilization provided by the Daemon

*Provides multi-tenancy support for hardware-based KBN*
Justitia in One Slide

Tenant Apps

- App A
  - Lat
  - Memory
  - QP

- App B
  - B/w
  - Memory
  - QP

- App B
  - Tput
  - Memory
  - QP

Justitia Shaper

- Split QP
  - SQ
  - RQ
  - Splitter
  - Message Pacer

Justitia Daemon

- Latency Monitor
- Safe Utilization Calculator
- Tokens (t)

NIC
Passive Latency Monitoring

- Probing latency-sensitive apps is expensive
- Maintains a reference flow

<table>
<thead>
<tr>
<th>Provides isolation?</th>
<th>Latency</th>
<th>Throughput</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Throughput</td>
<td>YES</td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>Latency</td>
<td><strong>YES</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Multi-Resource Token

- Used to rate-limit both bandwidth- and throughput-sensitive apps
- Represent either resource based on the application type

One Token

- Bandwidth-sensitive app: One spilt chunk
- Throughput-sensitive app: Equivalent number of messages
Transpareently split messages with \textit{Split QP}
More in Our Paper

• How does the shaper works
  o Dynamic receiver-side updates
  o Batch pacing for throughput-sensitive apps

• Calculating Safe Utilization

• Determining token sizes
1. Provides isolation while maximizing utilization.
2. Works well for real RDMA-based applications.
3. Complements congestion control in incast scenarios.
5. …
Better Isolation and Utilization

Latency Target = 10us

- With B/W App
- With B/W App + Justitia

<table>
<thead>
<tr>
<th>Bandwidth (Gbps)</th>
<th>99th Percentile (us)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>0</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
</tr>
</tbody>
</table>

- InfiniBand

- RoCE

<table>
<thead>
<tr>
<th>Bandwidth (Gbps)</th>
<th>99th Percentile (us)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>0</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
</tr>
</tbody>
</table>
Isolate Real Applications: DARE vs Crail

- DARE Alone
- With CRAIL
- With CRAIL + Justitia

99th Percentile (us) vs Bandwidth (Gbps)

- InfiniBand

Bandwidth (Gbps):
- 19.3
- 26.1

99th Percentile (us):
- 19.3
- 89.1
- 26.1
Complements with Congestion Control

More details on how we handle incast can be found in the paper.
Justitia

Provides multi-tenancy support in a
• efficient,
• scalable, and
• transparent manner

without modifying
• applications,
• operating systems, or
• hardware

https://github.com/SymbioticLab/Justitia

Provides multi-tenancy support for hardware-based KBN

Thank you!