Outline

- Announcements
  - one-page project write-ups due today!

- Last time: Low-level messaging layers
  - Active Messages, Fast Messages: “RISC” approach to communication
  - focus on delivering hardware performance to application

- This lecture: High-level messaging layers
  - rationale: making messaging layers more usable
    - support for connections, protection (multiprogramming), fault-tolerance
  - case studies
    - VMMC-2
    - U-Net/MM [Niranjan Nilakantan]
    - Virtual Interface Architecture [Amit Nene]
Improving Usability of Messaging Layers

- Wish-list
  - support for multiprogramming (protection)
  - sender specifies source location, receiver specifies destination
    - no restrictions on user virtual memory space
  - guarantee in-order delivery, reliable delivery, sender-receiver decoupling
    - tolerate faults in addition to detecting them

<table>
<thead>
<tr>
<th></th>
<th>Support for Multiprogramming</th>
<th>Restrictions on User Address-Space</th>
<th>Receiver specifies Destination</th>
<th>Fault-tolerance</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMMC-2</td>
<td>✔</td>
<td>?</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>U-Net/MM</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✕</td>
</tr>
<tr>
<td>VIA</td>
<td>✔</td>
<td>✕</td>
<td>?</td>
<td>✔</td>
</tr>
</tbody>
</table>

Review of VMMC

- Shortcomings
  - requires that source and destination buffers be pinned down
  - difficult to implement zero-copy connection-oriented protocols
  - fault-detection, but no fault-tolerance
VMMC-2 on Myrinet

VMMC-2: Transfer Redirection

- Extends VMMC interface with receiver specified destination
  - sender sends to default buffer
  - receiver redirects data to chosen buffer
  - enables zero-copy connection-oriented protocols

- Buffer management is responsibility of higher-level protocol(s)
VMMC-2: User-level TLB

**UTLB is a per-process array holding physical addresses of pages belonging to this process’ virtual memory regions that are pinned in the host physical memory**

- **Management of mappings**
  - all mappings are initiated by the VMMC-2 library (host)
    - send buffers pinned on demand, receive buffers on redirect call
  - buffer identification using `<start index, count of contiguous entries>`
- **Protection**
  - all updates to UTLB go through the kernel
- **Performance**
  - fast lookup data structure to facilitate queries
  - UTLB cache on the Myrinet network interface
    - 8192 entries: `<process id, virtual address, physical address>`
- **Alternative:** Maintain global TLB on network interface (U-Net/MM)

VMMC-2: Reliable Data-link Layer

- **Tolerates CRC errors, corrupted packets, link errors**
  - all mappings (imported buffers) from failed nodes are invalidated
- **Mechanism**
  - implemented at the data-link layer between network interfaces
  - sender-side buffering
    - per-node queues
    - dynamically managed
  - sequence-number based retransmission
    - maintained on a per-node basis
    - acknowledgments by receiver free up buffering at sender
    - packet loss results in all subsequent packets being dropped
- **More efficient than implementing reliability in higher layers**
VMMC-2: Performance

- Point-to-point costs
  - latency = 13.4 µs, overhead = 4 µs
  - bandwidth = 93 MB/s \( (n_{1/2} = 1024 \text{ bytes}) \)

- How much do the high-level features cost
  - transfer redirection
    - 6 µs if mapping is present in the UTLB
    - 2.5 µs for each incoming packet (~108 bytes)
  - address translation
    - HIT: 1.5 µs on the host processor, negligible on network interface
    - HOST MISS: 75 µs (50 µs in VMMC-2 library, 25 µs in device driver)
    - NI CACHE MISS: 3 µs
  - reliable communication
    - latency: 2.3 µs (11.1 versus 13.4)
    - bandwidth: 5 MB/s (98 versus 93)

VMMC-2 Performance (contd.)

- Sockets library [Figures 5, 6 from the paper]
Lecture Summary

- State-of-the-art of cluster messaging technology
  - network interface on I/O bus
  - user-level access to NI
  - support for multiprogramming, fault-tolerance
  - latency: ~10 µs, bandwidth: ~100 MB/s

- Several active research projects
  - Active Messages, Fast Messages, U-Net, SHRIMP, others …

- Technology transition to industry
  - Virtual Interface Architecture (Microsoft/Compaq/Intel)

- Remaining problems
  - support for higher-level programming models
  - improving performance robustness (flow control, buffer management)
  - integration with process scheduling (Lecture 11)

Next Lecture

- Shared Virtual Memory
  - Performance Evaluation of Memory Consistency Models for Shared Memory Multiprocessors, Gharachorloo et al.
  - TreadMarks: Shared Memory Computing on Networks of Workstations, Zwaenepoel et al.
  - Improving Release-Consistent Shared Virtual Memory using Automatic Update, Iftode et al.