# VIA over the CLAN Network

David Riddoch, Steve Pope

{djr,slp}@uk.research.att.com

Laboratory for Communications Engineering,
University of Cambridge, England
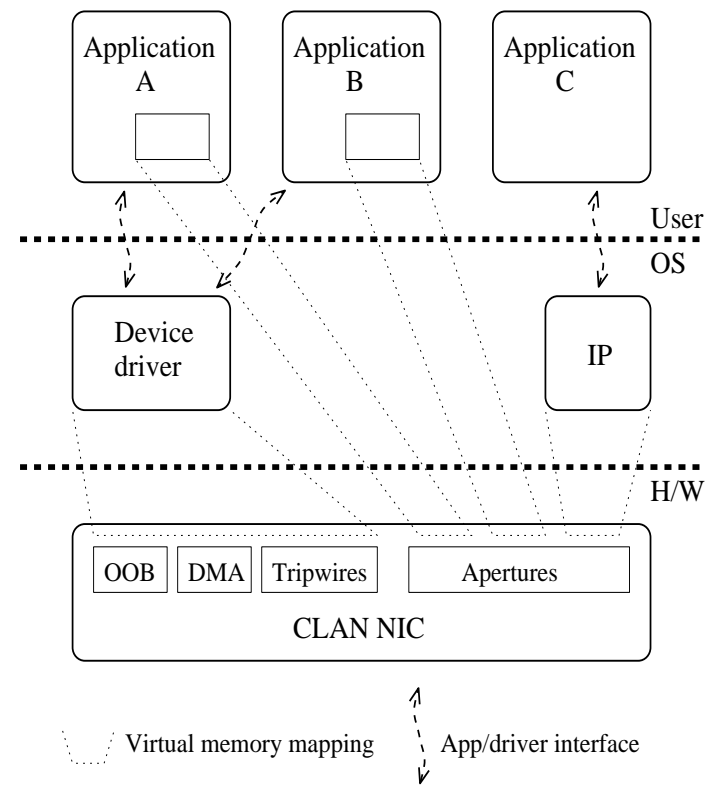
AT&T Laboratories - Cambridge

**AT&T Laboratories
Cambridge**

**UNIVERSITY OF
CAMBRIDGE**

# The CLAN Network

- *Not* related to Giganet cLAN!
- Non-coherent distributed shared memory
  - send-directed communication
- Out-of-band messages
- Programmable DMA
- Tripwires for synchronisation



- Very little per-endpoint resource in the NIC

**AT&T Laboratories Cambridge**
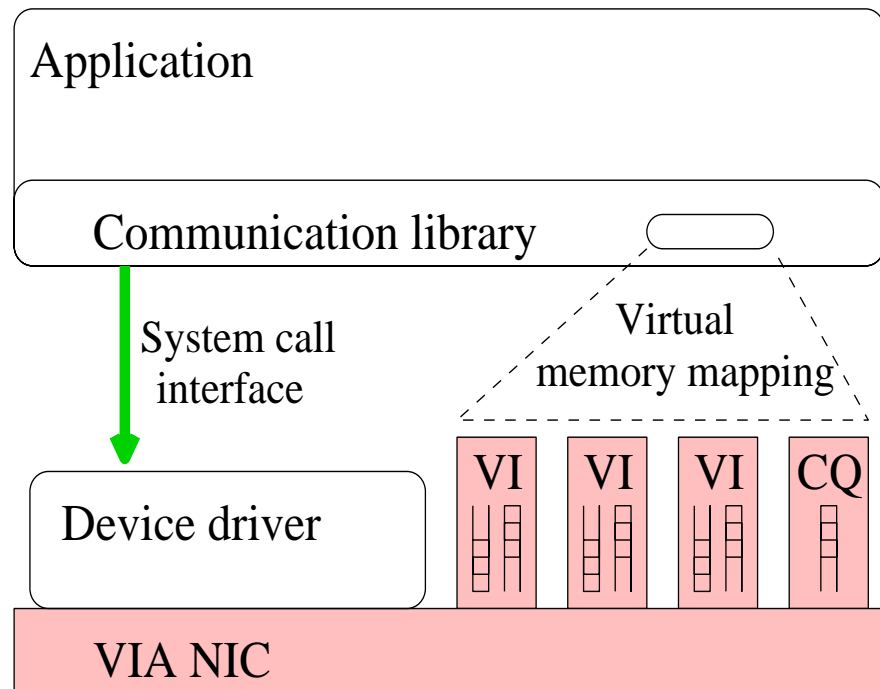
**UNIVERSITY OF CAMBRIDGE**

# CLAN Software Interface

- Out-of-band messages, tripwires and DMA are managed by the device-driver
    - good, reduces per-endpoint resources in NIC

- Want to avoid system calls on the data path

- Solution: Application and device-driver communicate using *shared memory*

- An asynchronous circular buffer allows messages to be passed between the application and device-driver without making system calls

# Virtual Interface Architecture

- Industry standard U/L network interface

- Connection-oriented

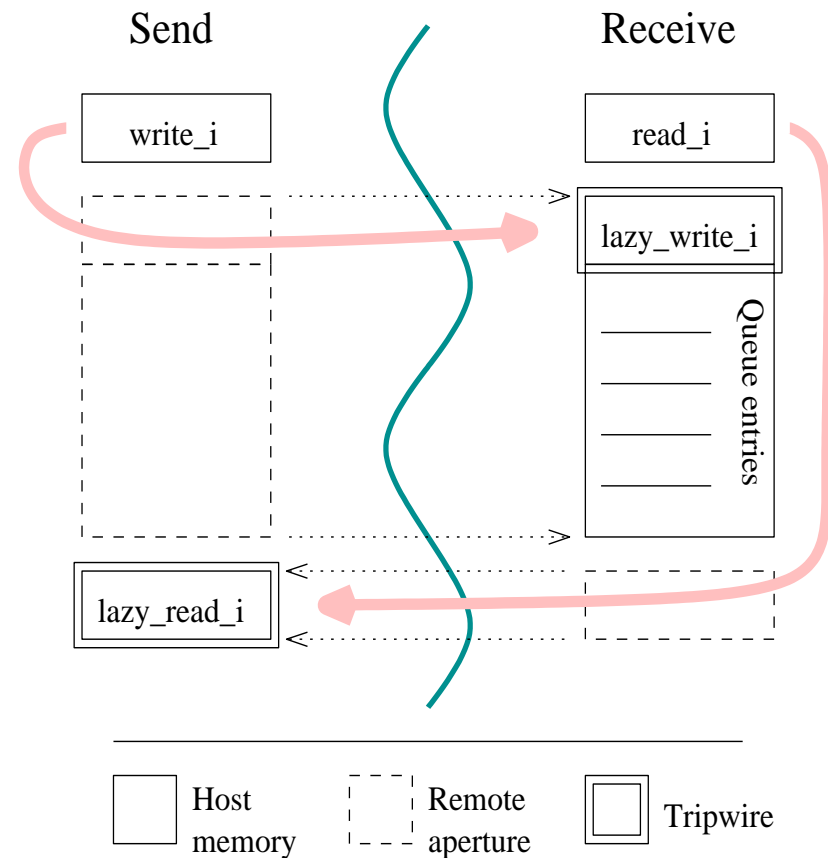- Lots of per-endpoint resource in the NIC
  - scalability, $$$$$$



Application

Communication library

System call interface

Virtual memory mapping

Device driver

VI    VI    VI    CQ

VIA NIC

# CLAN VIA

- **Message queue in shared memory used to pass:**
  - receive buffer descriptors from receiver to sender
  - completion messages from sender to receiver

- **DMA for data transfer**
  - but can also use PIO for small messages!
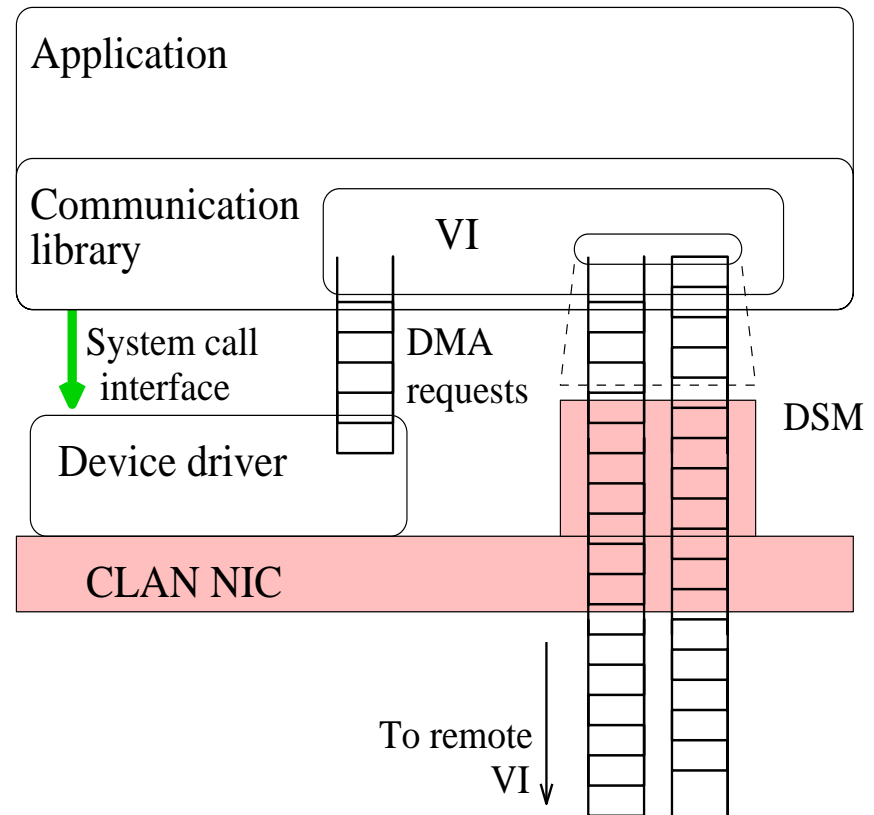
# Asynchronous Message Queue

- Based on circular buffer
- No 'full' flag, instead considered full when there is one free slot
- Avoid high latency reads with 'lazy' copies of queue pointers
- Tripwires for synchronisation

Send                                Receive

write_i                             read_i

lazy_write_i

Queue entries

lazy_read_i

Host memory        Remote aperture        Tripwire

**AT&T Laboratories Cambridge**

UNIVERSITY OF CAMBRIDGE

# CLAN VIA Data Transfer

1. receiver posts buffer descriptor

2. converted to CLAN RDMA cookie & passed to sender

3. sender initiates DMA requests to transfer data

4. write completion message

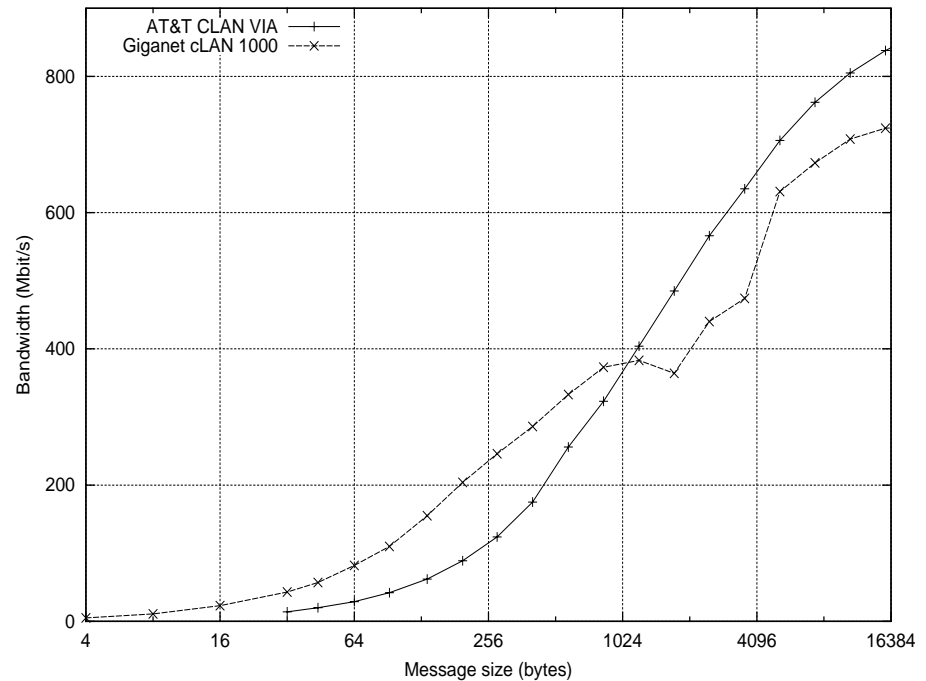5. tripwire at receiver provides synchronisation



Send

Receive

VI

Transfer Queue

Cookie Queue

VI
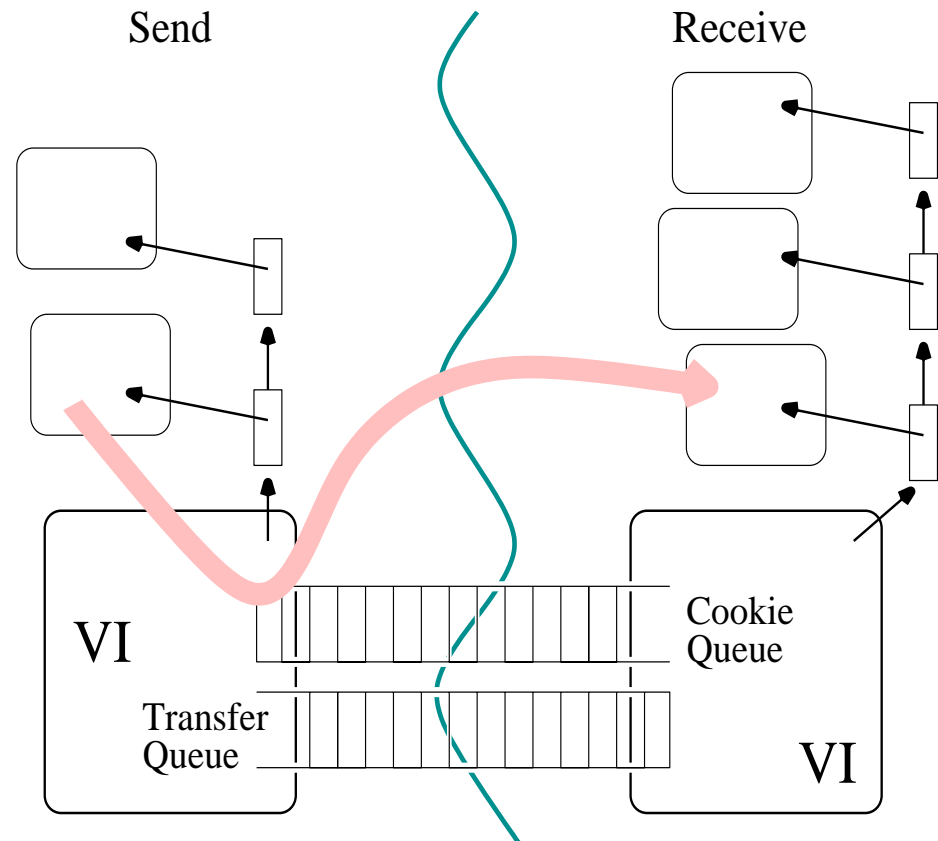
# Initial results

- ## 40 byte round-trip

| cLAN1000 | 19μs |
|----------|------|
| CLAN / DMA | 19μs |
| CLAN / PIO | 11μs |

- ## streaming bandwidth

# Disadvantages ...

- ## Protection
  - sender has direct access to receive buffers
  - proper protection will be possible with next revision of the NIC (under development), but will incur a cost

- ## Poor DMA performance for small messages
  - high overhead for DMA set-up
  - will be addressed with support for DMA chaining in the next revision of the NIC

# What's good about it?

- Low per-endpoint resource requirements in the NIC

- Flexible – software is easier to fix!

- Performance comparable with dedicated solutions

- VIA is not a good solution for all problems
  - can get better performance from raw CLAN network

- Extensions to VIA
  - flow control (never drop data)
  - more efficient buffer management

**AT&T Laboratories
Cambridge**

**UNIVERSITY OF
CAMBRIDGE**