



Taming LNet



Doug Oucharek
Intel® High Performance Data Division

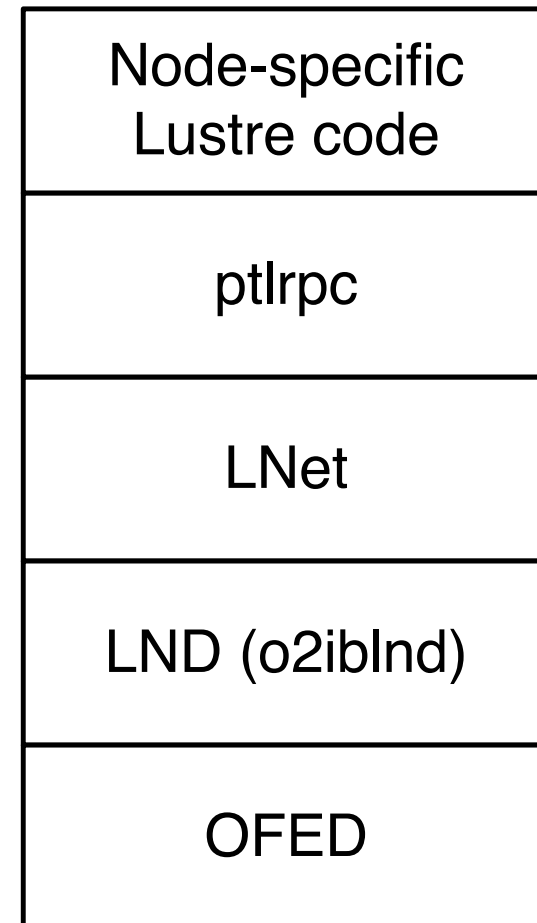
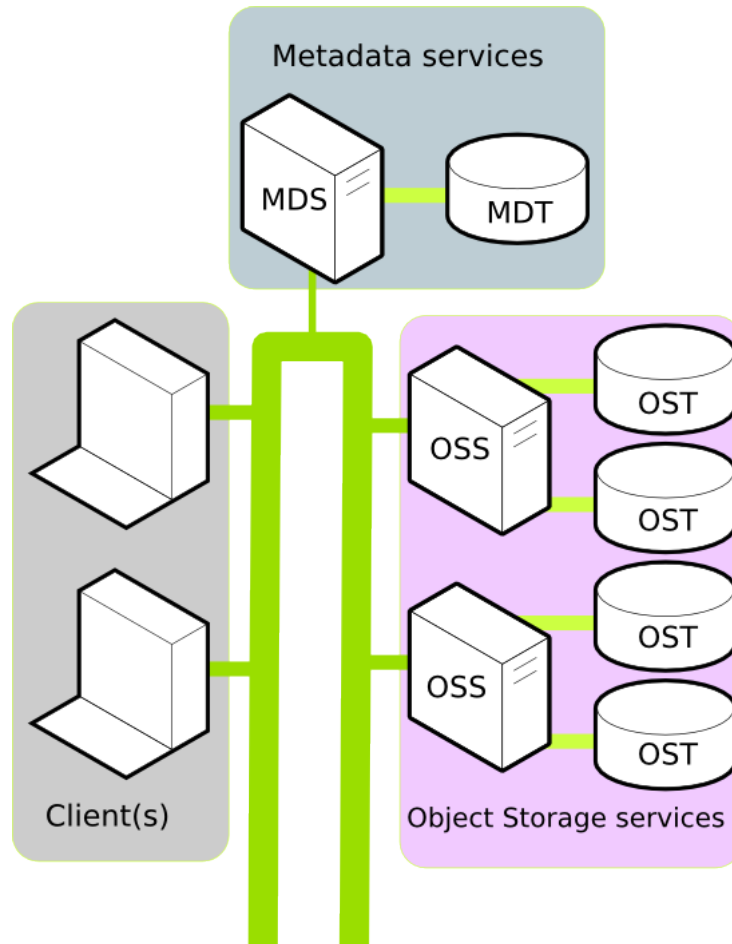


Overview



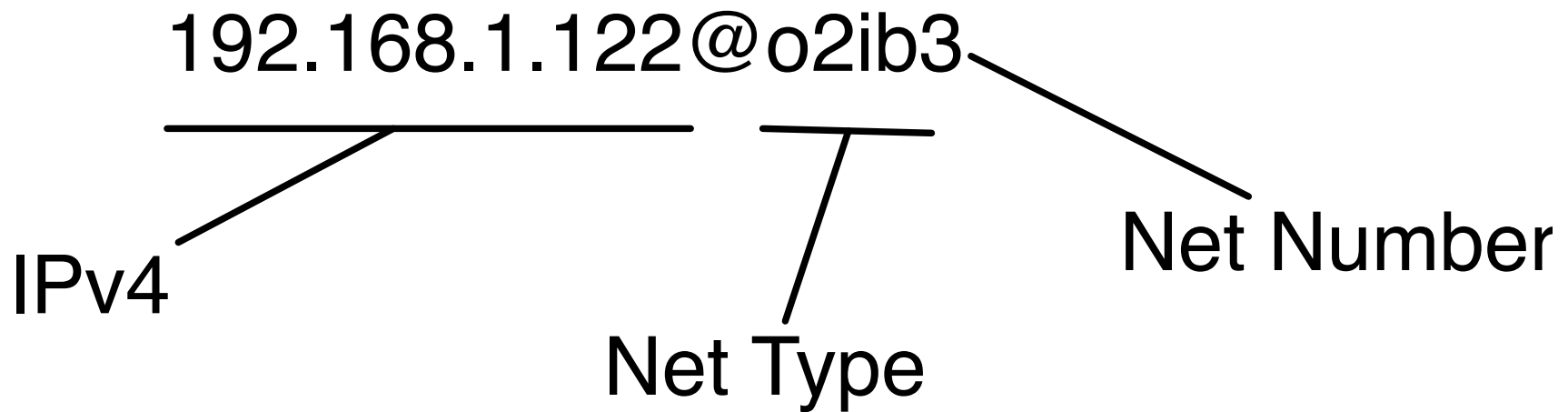
- Architecture of LNet
- Look at LNet config and problems
- Using LNet Selftest
- IB Tuning
- Dynamic LNet Config
- Wireshark

Lustre* and LNet

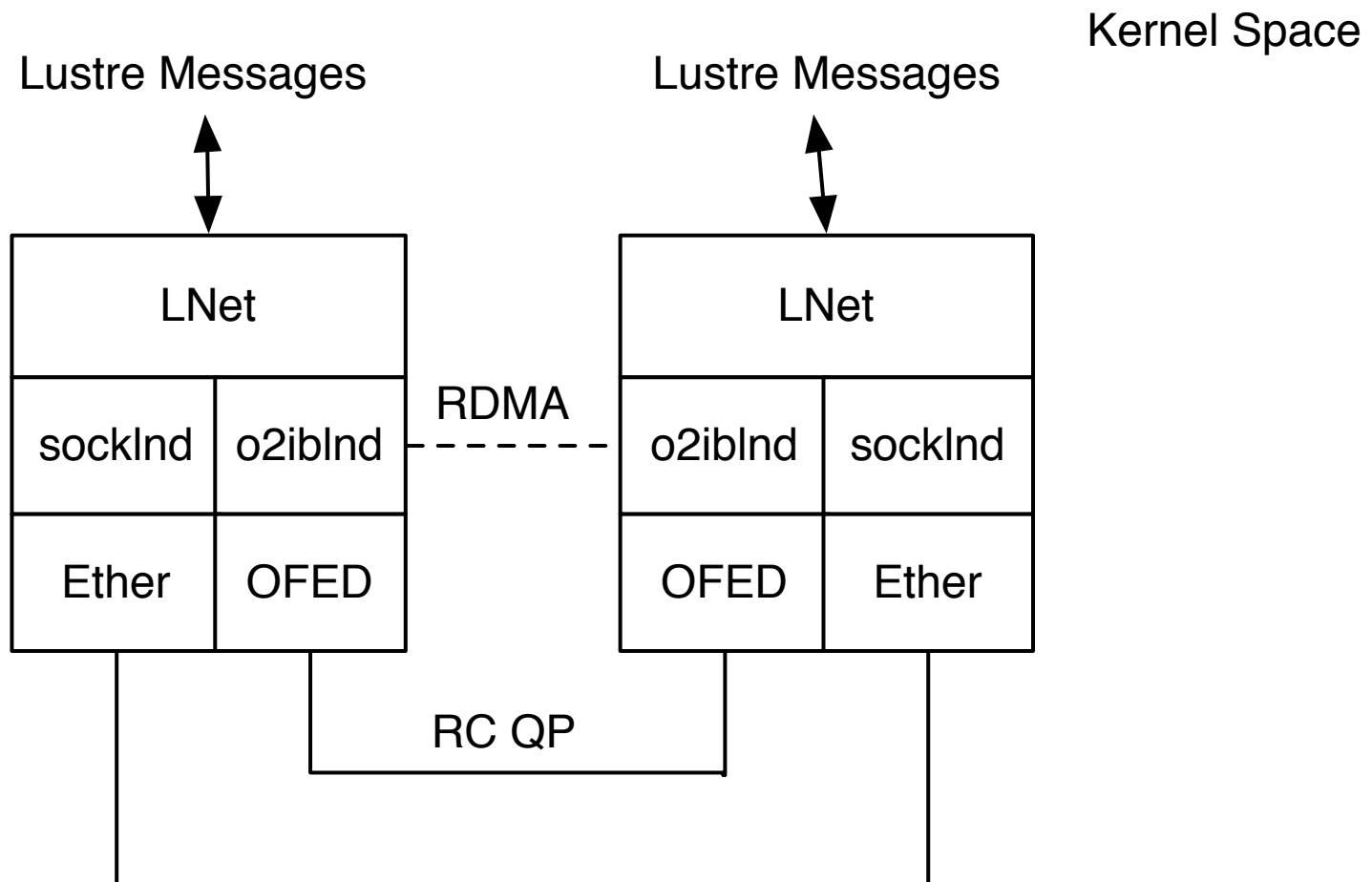


* Some names and brands may be claimed as the property of others.

Format of a NID



LNet and IB



LNet Config



Config:

```
options lnet networks="o2ib0(ib0)"  
routes="tcp0 192.168.1.2@o2ib0"
```

Tuning:

```
options ko2iblnd peer_credits=128  
fmr_pool_size=2048 credits=1024
```

- Network number must be used consistently across cluster

LNet Selftest



- Kernel module for testing LNet and LND's

```
#!/bin/bash
export LST_SESSION=$$
lst new_session read/write
lst add_group ion 10.211.55.9@tcp1
lst add_group server 10.211.55.7@tcp1
lst add_batch bulk_rw
lst add_test --batch bulk_rw --concurrency 16 --from ion --to
server brw write size=1M
lst run bulk_rw
lst stat server & sleep 30; kill $!
lst end_session
```

IB Tuning



- Defaults: Tuned to Mellanox IB
- For TrueScale: set `map_on_demand` to 32

```
options ko2iblnl peer_credits=128
peer_credits_hiw=64 credits=1024
concurrent_sends=256 ntx=2048
map_on_demand=32 fmr_pool_size=2048
fmr_flush_trigger=512 fmr_cache=1
```


Dynamic LNet Config: Purpose



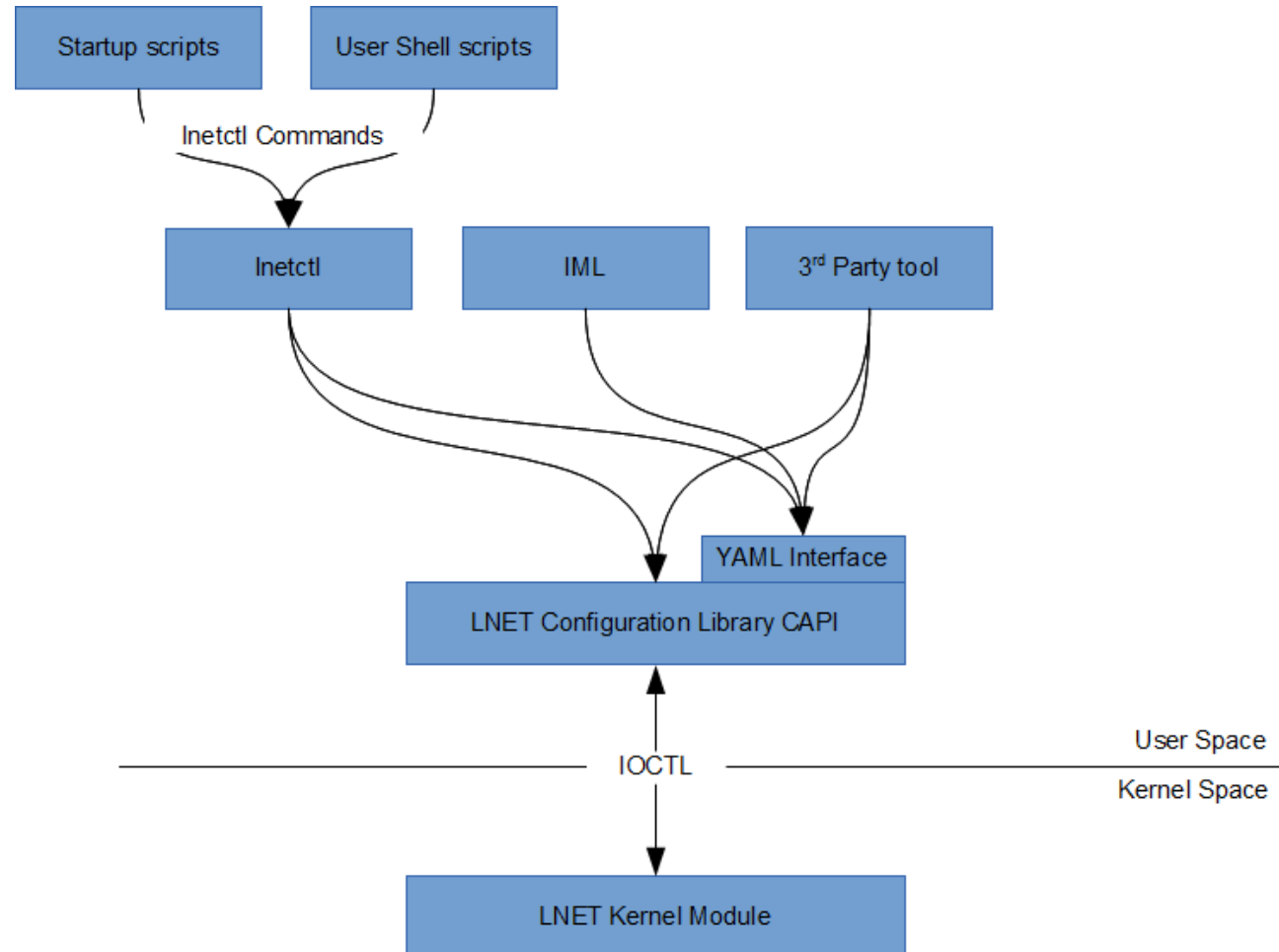
- Dynamically modify LNet configuration
 - will be landed in 2.7
- Ease the process of fine tuning LNet without having to restart the LNet kernel module
- Be more flexible for scripts and management applications

DLC: What can it do



- Adding/Deleting networks
- Adding/Deleting routes
- Configuring router buffer pools
- Enabling/Disabling routing.
- Showing routing information
- Importing/exporting configuration in YAML format

DLC: Block Diagram



DLC: YAML Example

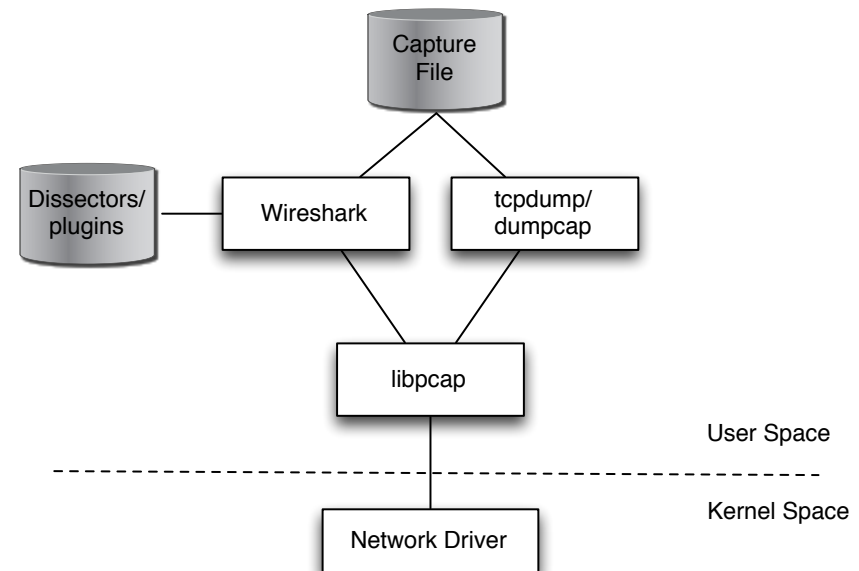


```
---
net:
  - net: tcp3
    status: up
    interfaces:
      0: eth4
    tunables:
      peer_timeout: 180
      peer_credits: 8
      peer_buffer_credits: 0
      credits: 256
  route:
    - net: tcp6
      gateway: 192.168.29.1@tcp
      hop: 4
      detail: 1
      seq_no: 3
    - net: tcp7
      gateway: 192.168.28.1@tcp
      hop: 9
      detail: 1
      seq_no: 4
  buffer:
    - tiny: 1024
      small: 2048
      large: 4096
...

```

Wireshark: Intro

- Protocol Analyzer
- Website:
www.wireshark.org
- Powerful filtering
- Powerful analytics/
stats
- Does support IB



Wireshark: Build + Install



Wireshark:

- Latest Stable: 1.10.2
- Requires: gtk2-devel and libpcap-devel (CentOS 6.x)
- Usual: ./configure, make, make install
- Application: /usr/local/bin/wireshark

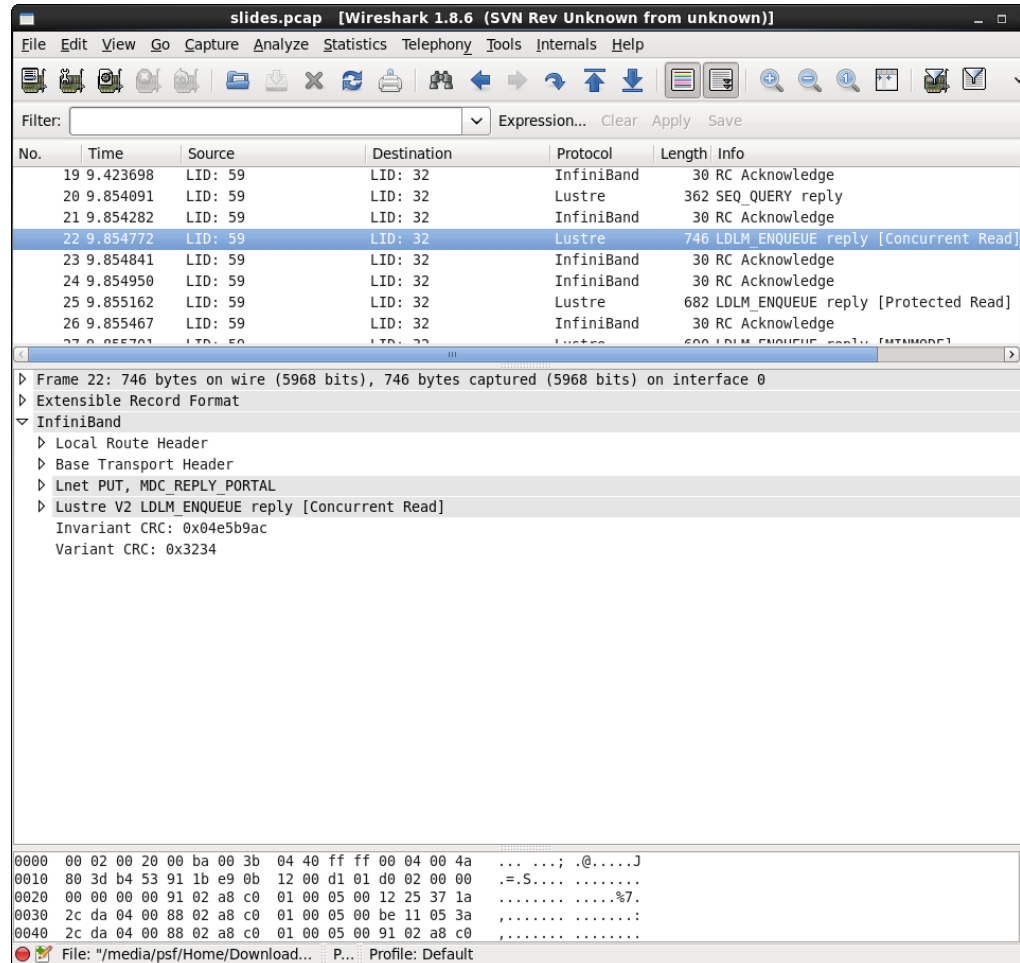
LNet/Lustre Dissectors:

- Need Wireshark source tree
- Go to: lustre/contrib/wireshark (2.4 and up)
- Update Makefile if not using package manager
- Just run “make install” (“make clean” first if previously built)
- Plugins: lnet.so and lustre.so

Wireshark: Capture IB

- Use ibdump (works like tcpdump)
- See:
http://www.mellanox.com/page/products_dyn?product_family=110&mtag=monitoring_debug
- Note: restricted to MTU of 2K or less
- Open captured file with Wireshark

Wireshark: Main Interface



Wireshark: Looking at LNet



```
▶ Frame 1: 354 bytes on wire (2832 bits), 354 bytes captured (2832 bits)
▶ Ethernet II, Src: Parallel_2a:de:5c (00:1c:42:2a:de:5c), Dst: Parallel_8e:1a:f5 (00:1c:42:8e:1a:f5)
▶ Internet Protocol Version 4, Src: OSS (10.211.55.9), Dst: MGS (10.211.55.7)
▶ Transmission Control Protocol, Src Port: 988 (988), Dst Port: 1023 (1023), Seq: 1, Ack: 1, Len: 288
▼ Lnet PUT, MGS_REQUEST_PORTAL
  Type of sockLnd message: KSOCK_MSG_LNET (0x000000c1)
  checksum disabled
  ack not required
  not ack
  ▼ dest_nid = 10.211.55.7@tcp0
    Destination nid: MGS (10.211.55.7)
    lnd network interface: 0
    lnd network type: SOCKLND (2)
  ▼ src_nid = 10.211.55.9@tcp0
    Src nid: OSS (10.211.55.9)
    lnd network interface: 0
    lnd network type: SOCKLND (2)
  Src pid: 12345 (0x00003039)
  Dest pid: 12345 (0x00003039)
  Message type: PUT (1)
  Payload length: 192
  DST MD index interface: 0xffffffffffffffff (18446744073709551615)
  DST MD index object: 0xffffffffffffffff (18446744073709551615)
  Match bits: 0x0004fadac7204803 (1401717457438723)
  hdr data: 0x0000000000000000 (0)
  ptl index: MGS_REQUEST_PORTAL (26)
  offset: 0
  msg filler (padding)
  Payload
▶ Lustre V2 OBD_PING request
```

Wireshark: Other features



- Packet length distribution stats
- Protocol hierarchy distribution stats
- Flow graph
- I/O Graph



Thank You



#OFADevWorkshop