Hewlett Packard
Enterprise

KFABRIC LUSTRE NETWORK DRIVER

Chris Horn, Ian Ziemba, Amith Abraham, Ron Gredvig

May, 2023

Cray User Group 2023

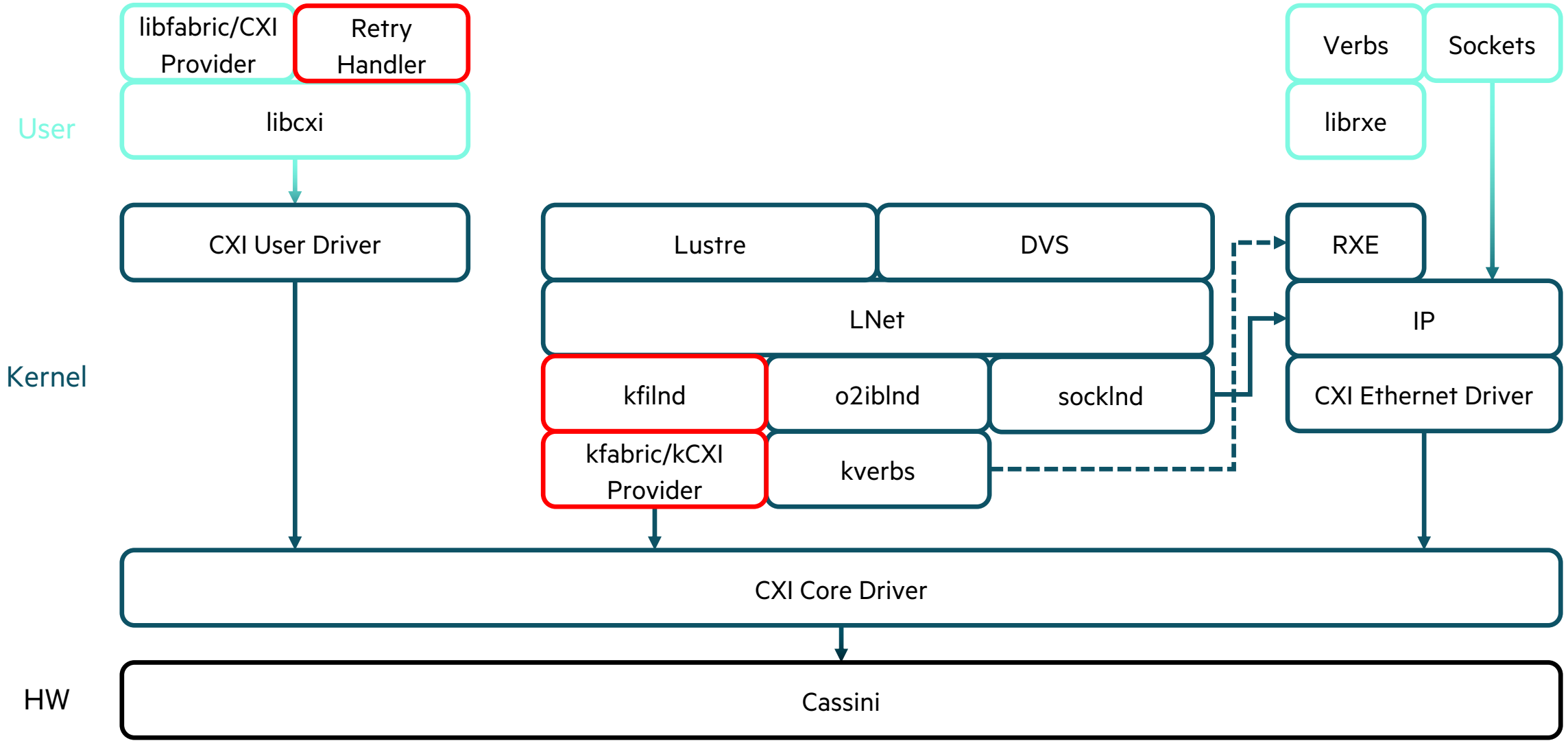
OUTLINE

- Software Overview
  - kfilnd
  - kfabric/kfi_cxi
  - Retry Handler
- kfilnd/kfabric/kfi_cxi features
- Serviceability & resiliency
- Diagnosing network trouble
- Future work

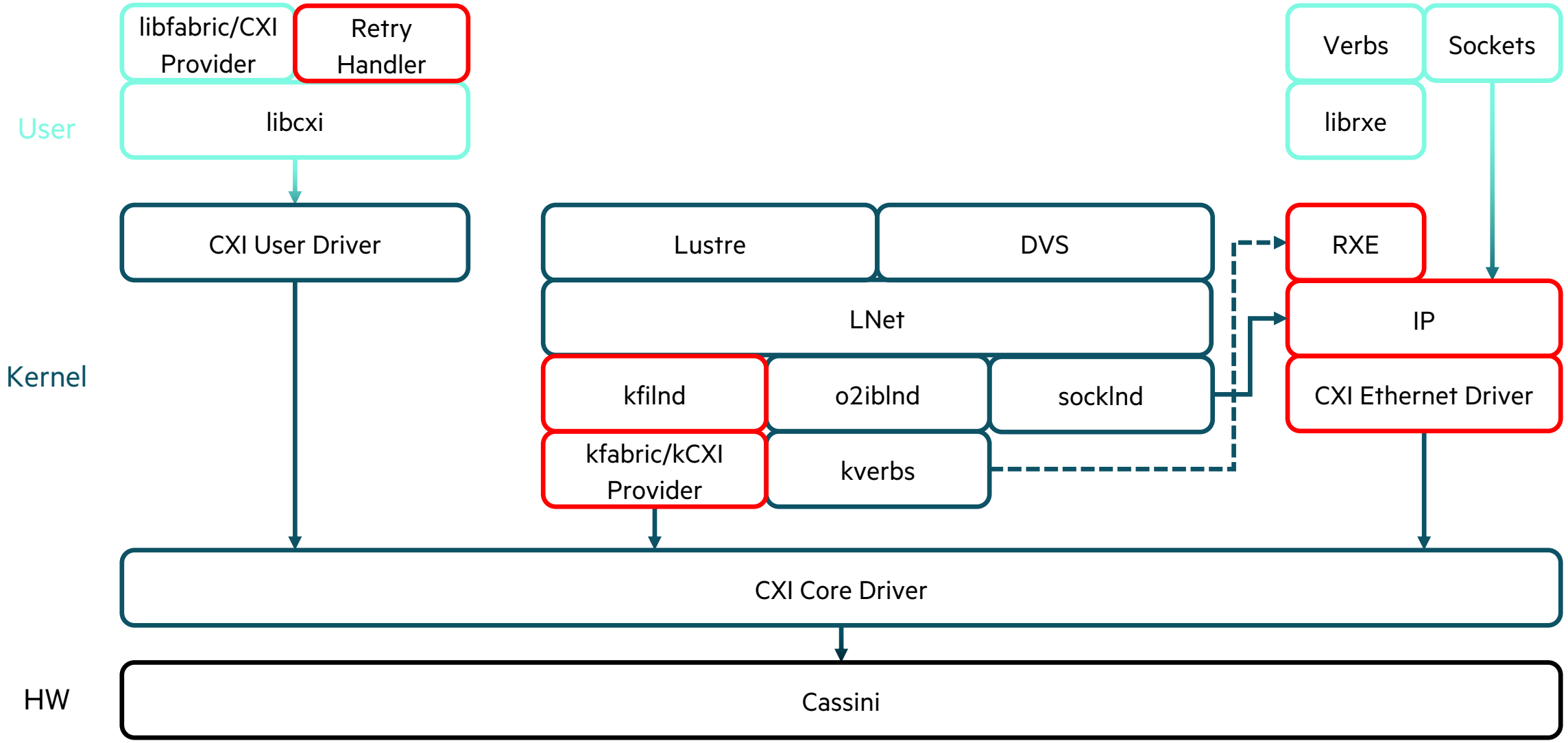
SOFTWARE OVERVIEW

- kfabric Lustre network driver == kfilnd
  - Uses numeric LNet NIDs: 1@kfi, 2@kfi, …
  - NID number == Destination Fabric Address (DFA)
    – Reflects group, switch and port numbers
  - Implements LND api (lnd_send, lnd_recv, etc.) using kfabric
  - Contributed to community Lustre 2.16
- kfabric
  - Connectionless and network-agnostic API used for RDMA in the kernel
  - Envisioned as common mid layer for multiple Upper Layer Protocols
  - Providers map kfabric API to lower-level network software/hardware
- kfi_cxi
  - kfabric provider for Cassini
  - optimized for LNet
- Retry Handler
  - Necessary because ordering is offloaded into the Cassini
  - Handles dropped packet retransmit and resource cleanup

SOFTWARE STACK



SOFTWARE STACK



KFILND/KFABRIC/KFI_CXI FEATURES

- Connectionless API
  - reliable unconnected datagram message endpoints
    - Single endpoint supports ”connections” to many peers
  - Avoid teardown/setup of connections in the LND
- Per-CPT Cassini resources
  - Each CPT is allocated a set of Cassini resources
  - Decreases contention for network resources amongst CPUs
- Multi-Recv Buffers
  - Allows for efficient use of HW resources
  - Single large buffer can byte-pack multiple msgs vs. dedicated buffer for every message
- Target events for GETs
  - Both initiator and target receive completion event when bulk transfer completes
    - avoids additional completion message
- Efficient message protocol with offloaded tag matching
  - Enables kfilnd to post recv buffer for a specific transaction
    - Rather than posting a pool of buffers for all TNs

KFILND SERVICEABILITY CHALLENGE

- kfilnd NID number == Destination Fabric Address (DFA)
- DFAs change with re-cabling (including cable swap)
  - Can't swap while LNet is running
- New DFA == new LNet NID
  - On a Lustre server, new NIDs require writeconf (or lctl replace_nids)
  - MGS NID changes -> All clients must update /etc/fstab
  - LNet router gets a new NID it may invalidate routing table on other peers
  - Leverage multi-lnd (socklnd + kfilnd) + UDSP
    - Format filesystem, define routes, define fstab using only tcp NIDs
    - No DFAs in config log, route config, or /etc/fstab
    - LNet discovery uses socklnd, filesystem traffic uses kfilnd
- Lustre User Group (LUG) 2023 talk goes into detail
  - Overview of Kfabric Network Driver, resiliency, serviceability and User Defined Selection Policy
  - https://www.opensfs.org/events/lug-2023/

KFILND RESILIENCY ISSUES

- Handshake protocol/peer cache
  - kfilnd maintains peer cache to translate LNet NID to KFI address handle
  - Handshake performed on initial send:
    - Exchange rx_base and remote session key (scalable endpoints)
    - Negotiate version
  - Issue: Peers purged from cache on any transaction failure
    - Silent message loss when message received from purged peer
    - Sender needs to wait for timeout to detect loss
    - Solution is to save “stale” peer info - set a flag to force handshake
  - Enhancement: Proactive handshake
    - Perform handshake if haven’t spoken with peer in certain time frame
    - Protects against case where Server reboots and peer cache is wiped out
  - Issue: Multiple handshakes in flight
    - Prior cache management could result in multiple handshakes in flight to single peer
  - All fixed in COS 2.3.109 and later
- Improve handling of multiple peer failures
  - kfilnd throttling (COS 2.4.96+)
  - proactive cancel of TN on handshake failure (COS 2.4.94+)
  - Traffic class support (COS 2.4.96+)

DIAGNOSING NETWORK TROUBLE - LNET/KFILND

- Diagnostics
  - initiator/target stats track kfilnd transaction times
  - /sys/kernel/debug/kfilnd/*/{initiator,target}_stats
- LNet Health and Recovery
  - lnetctl net show -v 2 | grep -e 'nid|health value'
  - lnetctl peer show -v 2 | grep -e 'nid|health value'
  - Max value is 1000, lowest is 0
- Console messages:
  - LNet recovery ping errors in Lustre 2.12

```
[353991.533158] LNetError: 996:0:(lib-move.c:4001:lnet_handle_recovery_reply()) peer NI (2120@kfi) recovery failed with -110
```

  - lnet_handle_recovery_reply() is very noisy
  - Replaced by LNet recovery informational messages in latest HPE 2.15 and Lustre 2.16

```
[1034843.558106] LNet: 1 peer NIs in recovery (showing 1): 16@kfi
[1035143.589781] LNet: 5 local NIs in recovery (showing 5): 1@kfi, 2@kfi, 3@kfi, 4@kfi, 5@kfi
```

- Show all NIDs in recovery
  - lnetctl debug recovery -l
  - lnetctl debug recovery -p

DIAGNOSING NETWORK TROUBLE - RETRY HANDLER

- Retry Handler (RH) can point you towards problem areas
- RH instance for every cassini interface
  - journalctl -u cxi_rh@cxi0; journalctl -u cxi_rh@cxi1 … ;
- Lines that mention “nid=X”

```
Apr 27 09:11:19 s-lmo-gaz38a cxi_rh[760463]: RH: PCT timeout event (... nid=16, ...
```

  - X is the DFA/numeric portion of LNet NID
  - Good idea to have a way to map DFA to hostname for triage
- Expected vs. unexpected RH activity
  - If a Lustre server crashes, expect a lot of RH activity
  - A filesystem under high load may result in some RH activity
    - Resource busy NACK -> resend -> success

```
Apr 27 07:44:41 cassini-hosta cxi_rh[11672]: RH: PCT NACK event (... sct=2060, rc=RESOURCE_BUSY, nid=17, ...
...
Apr 27 07:44:41 cassini-hosta cxi_rh[11672]: RH: sct=2060 all retries issued
Apr 27 07:44:41 cassini-hosta cxi_rh[11672]: RH: retry completed for sct=2060
```

FUTURE WORK

- kfilnd NID -> CPT optimization
  - LNet hashes NID to a CPT
  - Current algorithm optimized for IPv4 -> uneven distribution for kfilnd
  - New algorithm spreads kfilnd NIDs more evenly
  - Reduces contention, leverages more CPU, increases performance
- kfilnd workqueue (WQ) configuration
  - Existing WQ config causes contention
  - Reduce WQ priority from high to normal
  - Optimizing inflight work items per CPT CPUs
  - ~25% performance improvement in multi-client metadata
- IPv4 Support

THANK YOU

chris.horn@hpe.com

REAL WORLD EXAMPLES

- Customer reported
- Try to find a real world example of:
- <LNet/Lustre error messages>
- <RH log showing related activity>
- <The above pointed us to root cause X>

KFABRIC SCALABLE ENDPOINTS

- Scalable endpoints allow multiple tx/rx contexts to be opened against a single address.

MULTI-LND SUPPORT