13th ANNUAL WORKSHOP 2017

OPEN SOURCE NFS/RDMA ROADMAP

Chuck Lever, Linux Kernel Architect

Oracle Corporation

March 30, 2017

SYLLABUS

Today's topics

- Upstream accomplishments in 2016
- Challenges and opportunities
- Standards activity and what is motivating it

What I'm not going to cover

- Linux distribution features sets and delivery schedules
- Quantitative and comparative performance results
- Adoption rates

2016 OPEN SOURCE HIGHLIGHTS

NEW UPPER LAYER FEATURES

- **NFS version 4 minor version 1 and newer, with RDMA**
  - Integrated backchannel
  - NFSv4.1 sessions
  - pNFS (all layout types)
  - NFSv4.2 features such as READ_PLUS, ALLOCATE, SEEK_HOLE
  - Interoperates with Solaris NFSv4.1 prototype

- **NFS/RDMA with Kerberos**
  - krb5, krb5i, krb5p
  - Full interop with Linux/Linux
  - Limited interop for Linux/Solaris, more to come

PERFORMANCE AND SCALABILITY

- **On-demand MR allocation**
- **SG_GAP support with FRWR**
- **In-place RDMA Send**
- **Experimental features**
  - Remote Invalidation
  - Large inline threshold
  - Rudimentary transport property exchange

DEEPER TESTING

- **Testing and development work now includes Linux server**
- **Distributed testing with a variety of NICs and fabrics**
  - NIC vendors
  - Linux distributors
- **Platform diversity**
  - Still x86-64 only
  - Gap: So far, no testing on ARM, PPC, SPARC, or z/390
- **IOMMU enables NFS/RDMA in guests**
  - DMA-API usage debugging
  - Strict IOMMU settings

WIRESHARK IMPROVEMENTS

- **Already in v2.3.0rc**
  - Reliable RPC-over-RDMA frame detection
  - RPC-over-RDMA transport header parsing is working
  - Display filters available for transport header fields
  - RPC call/reply matching improvements

- **Next steps**
  - Passing re-assembled RDMA_NOMSG messages up to RPC dissector
  - Re-assembly of RDMA_MSG messages that include Read or Write chunks

CHALLENGES

MAGICAL PONIES

- Unstable NFS WRITEs ought to be nearly as fast as NFS READs
- One large client ought to reach a million IOPS
- NFS/RDMA is well-positioned to expose performance benefits of persistent memory
- RPC-over-RDMA ought to work efficiently on platforms with large pages

THERE BE DRAGONS

- **NFS I/O operations**
  - Current broadly deployed durable storage technologies still involve I/O on the NFS server
  - Existing client RPC stacks depend on context switches and heavyweight locking
  - RDMA Read requires an additional round trip
  - Still only one QP per mount point

- **NFS small I/O and metadata operations**
  - Receive is typically not zero-copy
  - The cost of providing a Reply chunk is usually wasted
  - Explicit RDMA is used for frequent non-I/O requests
  - Default inline threshold is 1KB

TRANSPORT PROTOCOL REALITIES

- **Server cannot return oversized replies**
- **Canceled RPCs can result in connection loss / denial of service**
- **Can be difficult to match multiple result data items to Write chunks**
- **No in-band support for Remote Invalidation**
- **Incomplete support for reverse-direction RPC transactions**
  - No RPC call direction indicated in the transport header
  - How to use chunks for reverse direction transactions

TRANSPORT PROTOCOL REALITIES

- **Credit accounting implementations assume one RPC (call and reply) per credit**
  - Non-antiphonal messages
  - Retransmits
  - Multiple RPC-over-RDMA messages per RDMA Send
  - Multiple RDMA Sends per RPC message
  - Distinct control plane and data plane

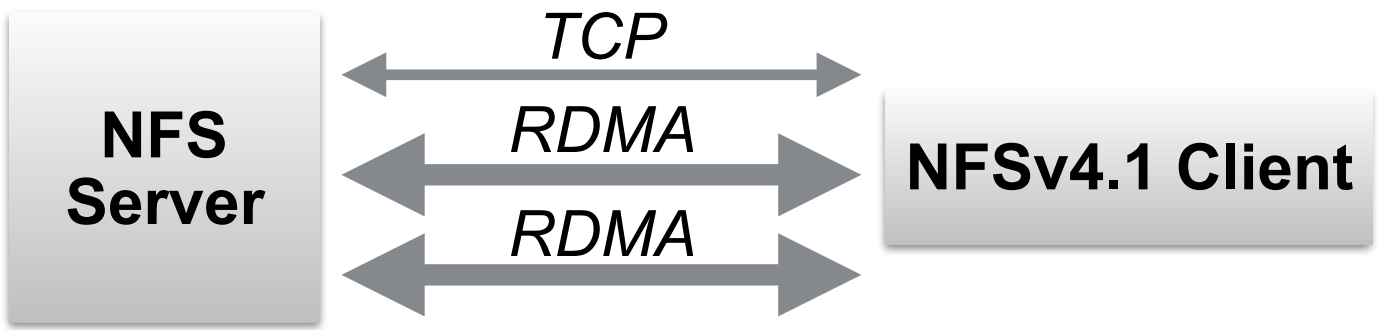
- **Extensibility is limited**
  - No extensibility without a version number bump
  - No connection property exchange

- **RPCSEC GSS is not an ideal fit**
  - RPC-over-RDMA transport header fields are not protected
  - Integrity and confidentiality require host CPU resources

HERE'S THE GOOD NEWS

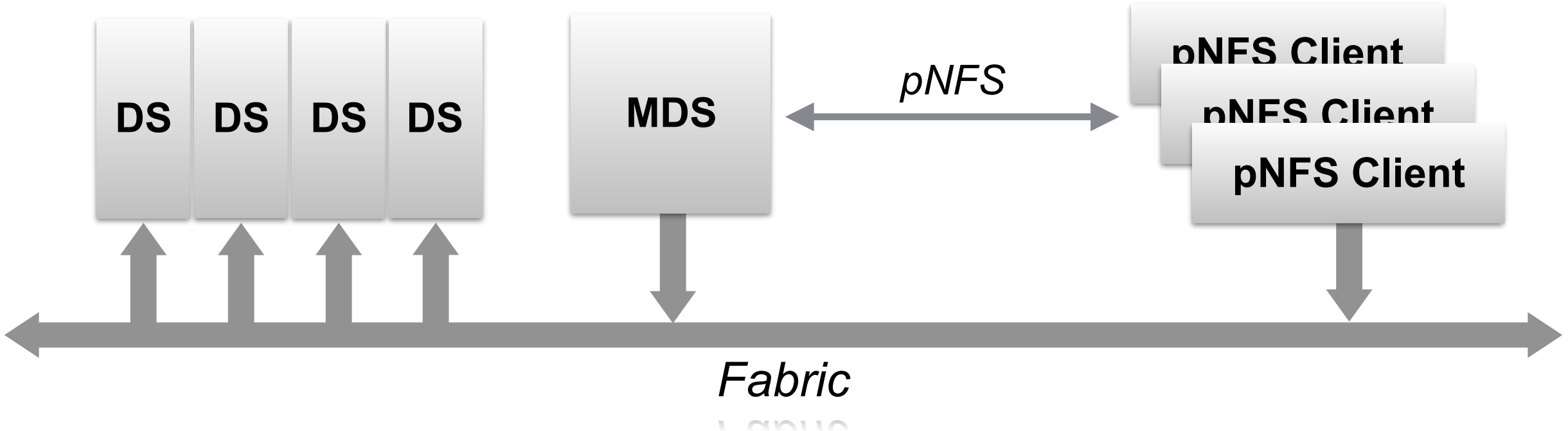
NFSV4.1 MULTI-PATH CAPABILITIES

- **Client ID and session trunking**
  - Multiple network paths from one NFS client to one NFS server



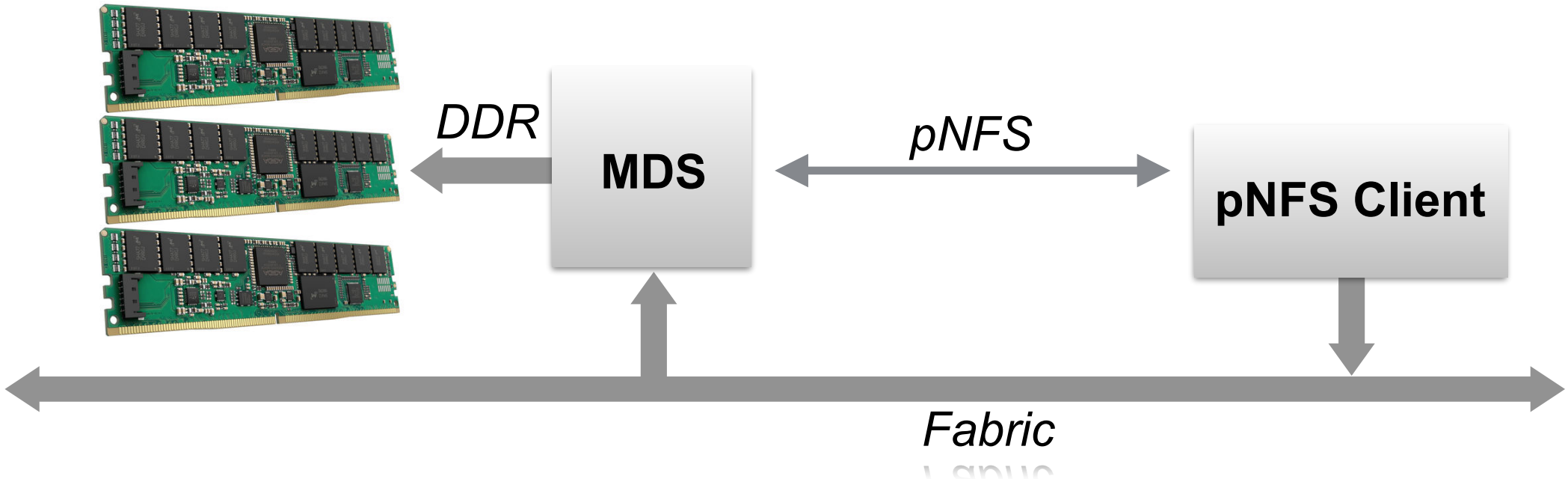
NFSV4.1 MULTI-PATH CAPABILITIES

- **Existing pNFS block layout type using RDMA-enabled block transports**
  - iSER
  - SRP
  - NVMe on Fabrics



NFSV4.1 MULTI-PATH CAPABILITIES

- **New pNFS layout type**
  - Similar to SMB "push mode" described by Talpey, *et al*
  - NFS server registers persistent memory, returns R_keys in pNFS layouts
  - NFS client accesses target persistent memory via explicit RDMA operations
  - Avoids RPC, server host interrupts in I/O path
  - User space clients might avoid local OS kernel interaction in I/O path



2017 FOCUS

- **Recovery**
  - Always disconnect on RPC time-out
  - Handle DEVICE_REMOVAL events
    - Device hotplug
    - Device failover
    - Suspend/resume with active NFS/RDMA mounts
- **Kerberos interop**
  - Server support for multi-chunk RPCs
- **Full stack performance**
  - Multi-pathing support
  - Improve server Receive efficiency
  - Relieve lock contention on client
- **Transition to RoCE**

IETF AND NFS/RDMA STANDARDS

UPDATES OF EXISTING RFCS

- **RPC-over-RDMA Version One**
  - RFC 5666 (2010): defines the RPC transport layer behavior
  - Document update: document existing implementation behavior and clarify interop issues
  - Status: update headed to RFC Editor

- **RPC-over-RDMA bidirectional operation**
  - New RFC: enables reverse-direction RPC calls on RPC-over-RDMA
  - Purpose: enable NFSv4.1 on RPC-over-RDMA
  - Status: new doc headed to RFC Editor

- **NFS binding to RPC-over-RDMA Version One**
  - RFC 5667 (2010): specifies how NFS protocols use RPC-over-RDMA
  - Document update: document existing implementation behavior and finish support for NFSv4.1
  - Status: document being completed in nfsv4 Working Group

NEW DIRECTIONS

- **Client multi-path discovery**
  - New I-D that defines in-band mechanism for discovering NFS server network interface capabilities
  - Enables client ID and session trunking using any transport

- **RPC-over-RDMA CM private data**
  - New I-D that defines mechanism for exchanging transport properties during connection set-up
  - Enables RPC-over-RDMA Version One peers to discover support for large inline thresholds, etc.

- **RPC-over-RDMA Version Two**
  - New I-D that specifies new version of RPC-over-RDMA
  - Adds transport protocol support for
    - Remote Invalidation
    - Larger default inline thresholds
    - Rich error reporting
    - Protocol extensibility

RPC-OVER-RDMA VERSION TWO EXTENSIONS

- **In-band connection property exchanges and updates**
  - Enables discovery and modification of connection properties such as inline threshold
- **Message Continuation**
  - Enables sending RPC messages that span multiple RDMA Sends
- **“Send-based Direct Data Placement”**
  - Transfers payloads eligible for Direct Data Placement using only RDMA Send
  - Enables some forms of zero-copy Receive
- **Responder-provided Read chunks**
  - Enables servers to return arbitrarily large replies without a Reply chunk
  - Eliminates need for client to provide Reply chunk for reply that is likely going to be small

REMAINING WORK

- **Generic zero-copy Receive**
- **Credit accounting improvements**
- **Handling canceled RPCs without risking connection loss**
- **Multiple Read and Write chunks per RPC**
  - Matching results to Write chunks
  - Few NFSv4 clients generate COMPOUNDs with multiple payload-bearing operations
  - Few NFS/RDMA clients generate COMPOUNDs with multiple chunks
- **Security with offload**
  - Protection for transport header fields
  - Cooperation with offloaded security implementations

OPENFABRICS
ALLIANCE

13th ANNUAL WORKSHOP 2017

THANK YOU

Chuck Lever, Linux Kernel Architect

Oracle Corporation