# Enhanced Adaptive Compression in Lustre

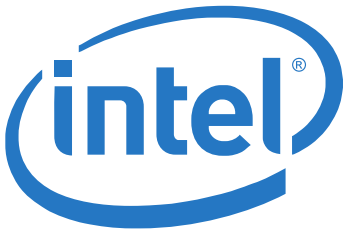
## IPCC for Lustre

Anna Fuchs

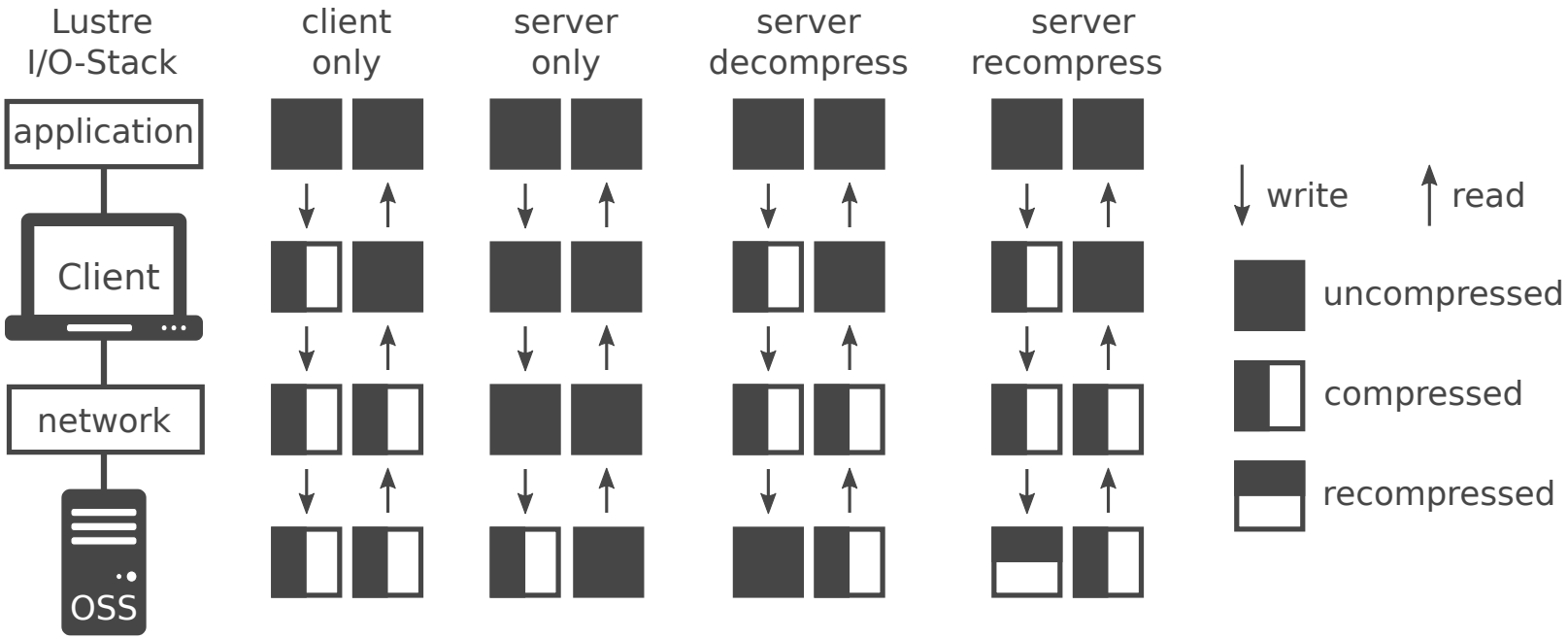
anna.fuchs@informatik.uni-hamburg.de

Research Group Scientific Computing
Department of Informatics, Universität Hamburg

2016-09-22

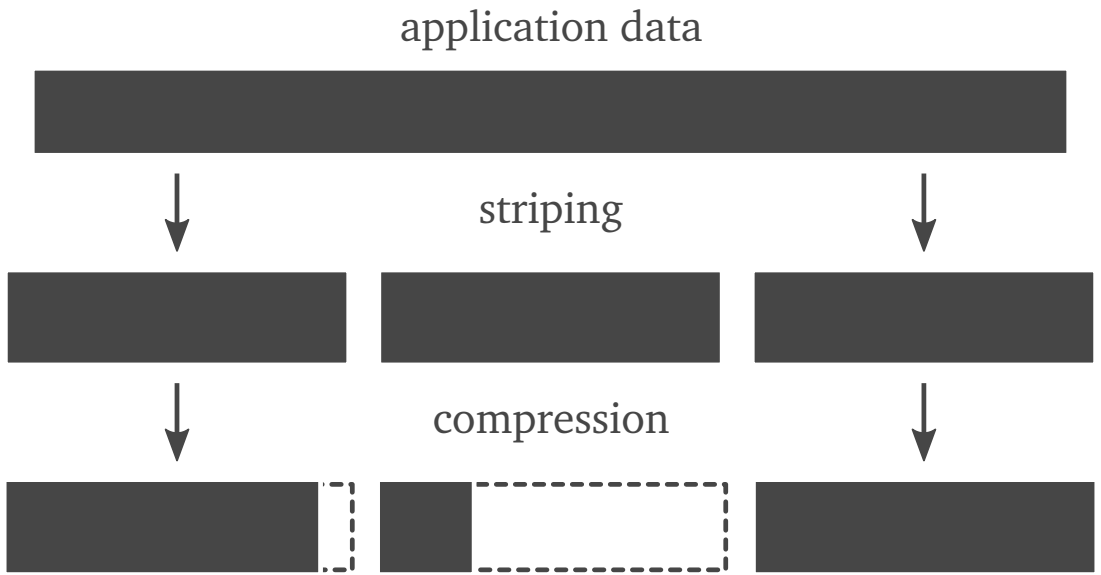
# Lustre stack




- Different combinations depending on the needs
- Server able to decompress for random I/O

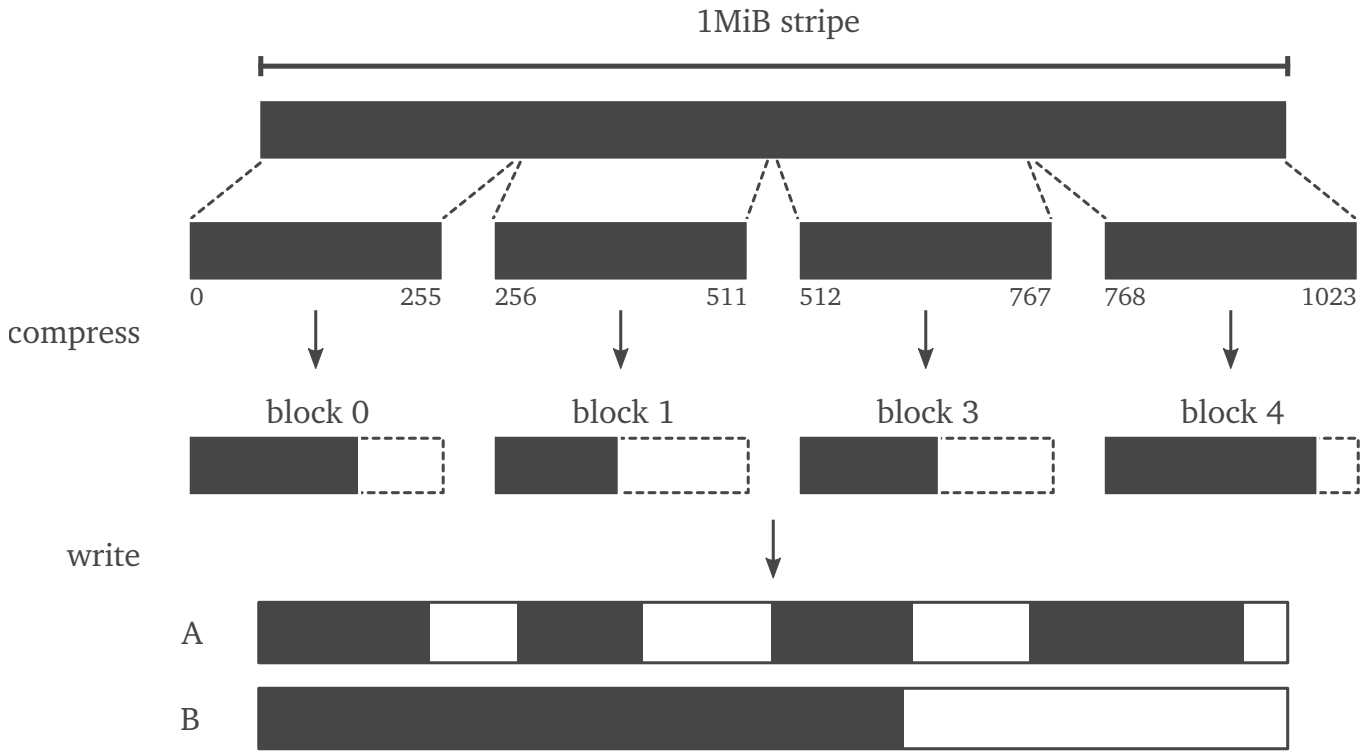
# Basic functionality

- Based on stripes
- Early abort for incompressible data
- Metadata for every stripe (or record)
- Additional service within PTLRPC-layer (before GSS)
  - Reuse or extension of GSS-layer functionality



# Sub-striping

- + Independent blocks compressed in parallel
- + Reduce read-modify-write issues
- – Read-ahead potentially worse

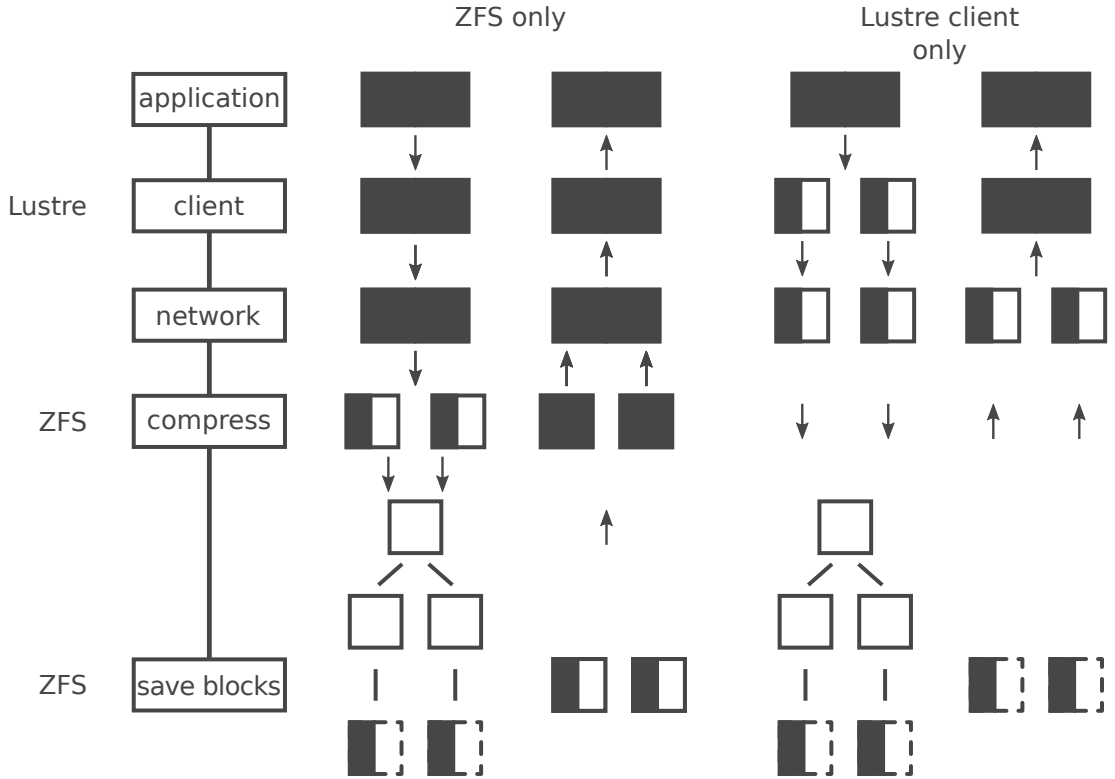


# ZFS

- Primary focus on ZFS backend
  - Already supports record-level compression
  - Each block compressed independently
  - All-zero blocks converted into file holes

# ZFS

- Primary focus on ZFS backend
  - Already supports record-level compression
  - Each block compressed independently
  - All-zero blocks converted into file holes
- Goal: ZFS-Lustre interaction; reuse data structure



# Alignment

- Align ZFS's records and Lustre's stripes and sub-stripes
- Match alignment for best performance
- Skip compression in ZFS – *discussion*
- Extend ZFS-Lustre interface

- Lustre (client/server) de-/compresses, ZFS manages metadata (location, offsets, etc.)

# Algorithms

- Modern algorithms reach 3+ GiB/s compression, 6+ GiB/s decompression throughput

# Algorithms

- Modern algorithms reach 3+ GiB/s compression, 6+ GiB/s decompression throughput
- Lack of implementation within kernel
  - Older LZ4 available from kernel 3.11
  - CentOS7 Kernel 3.10
  - Solution for Lustre – *discussion*

# Algorithms

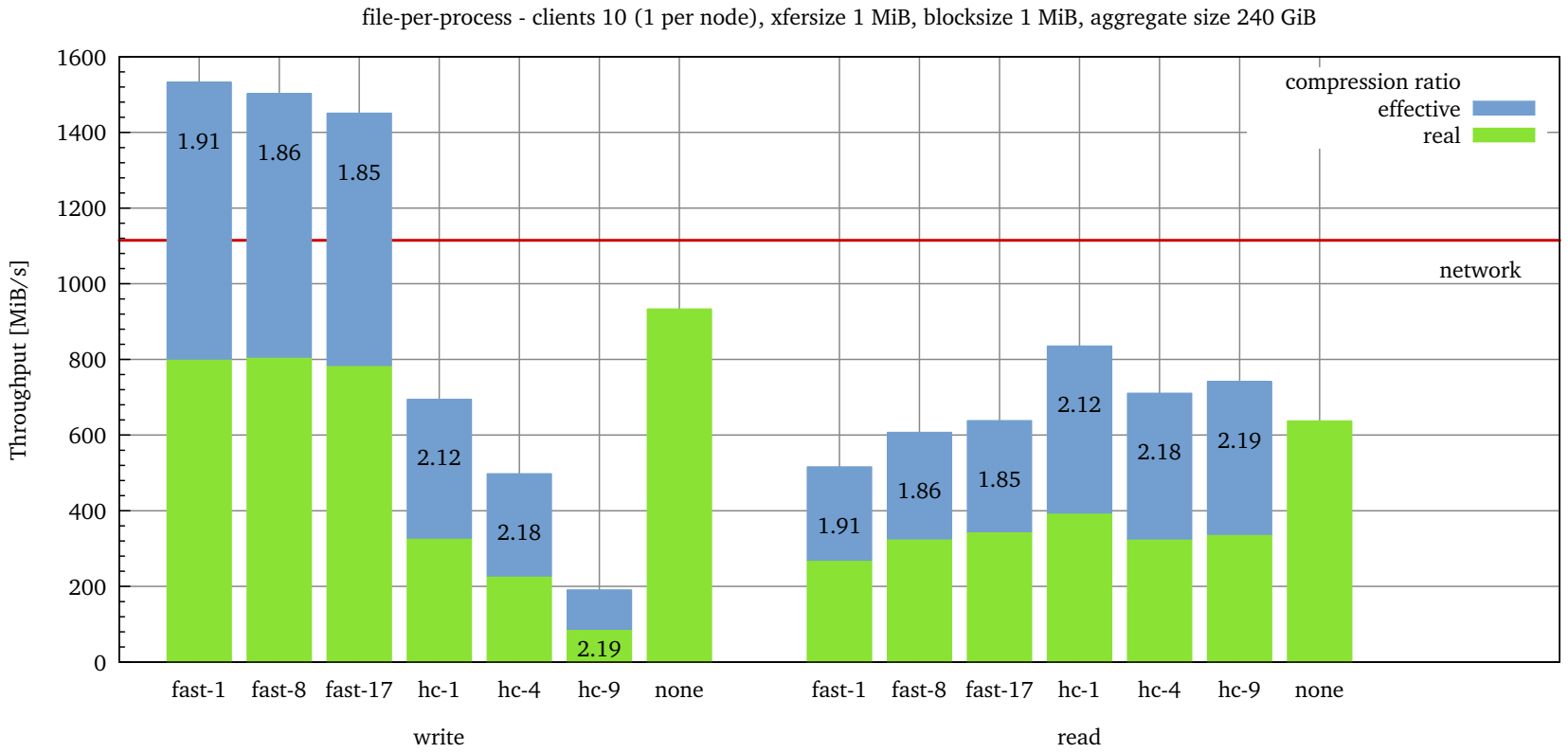
- Modern algorithms reach 3+ GiB/s compression, 6+ GiB/s decompression throughput
- Lack of implementation within kernel
  - Older LZ4 available from kernel 3.11
  - CentOS7 Kernel 3.10
  - Solution for Lustre – *discussion*
- Memory handling
  - Common algorithms work with independent buffers
  - Data in Lustre PTLRPC accessible on page level (*scatterlists?*)
  - Memory consumption – *discussion*
- Intel QuickAssist

# Features

- Adaptive compression – *how configurable?*
  - Internal adaptivity within system
    - Network data rate
    - Available computational resources
    - File sizes – worthwhile for big files
    - ...
  - High-level user hints with `ladvice`
    - Bad/well compressible
    - Frequently read
    - Written once
    - Archive
    - ...

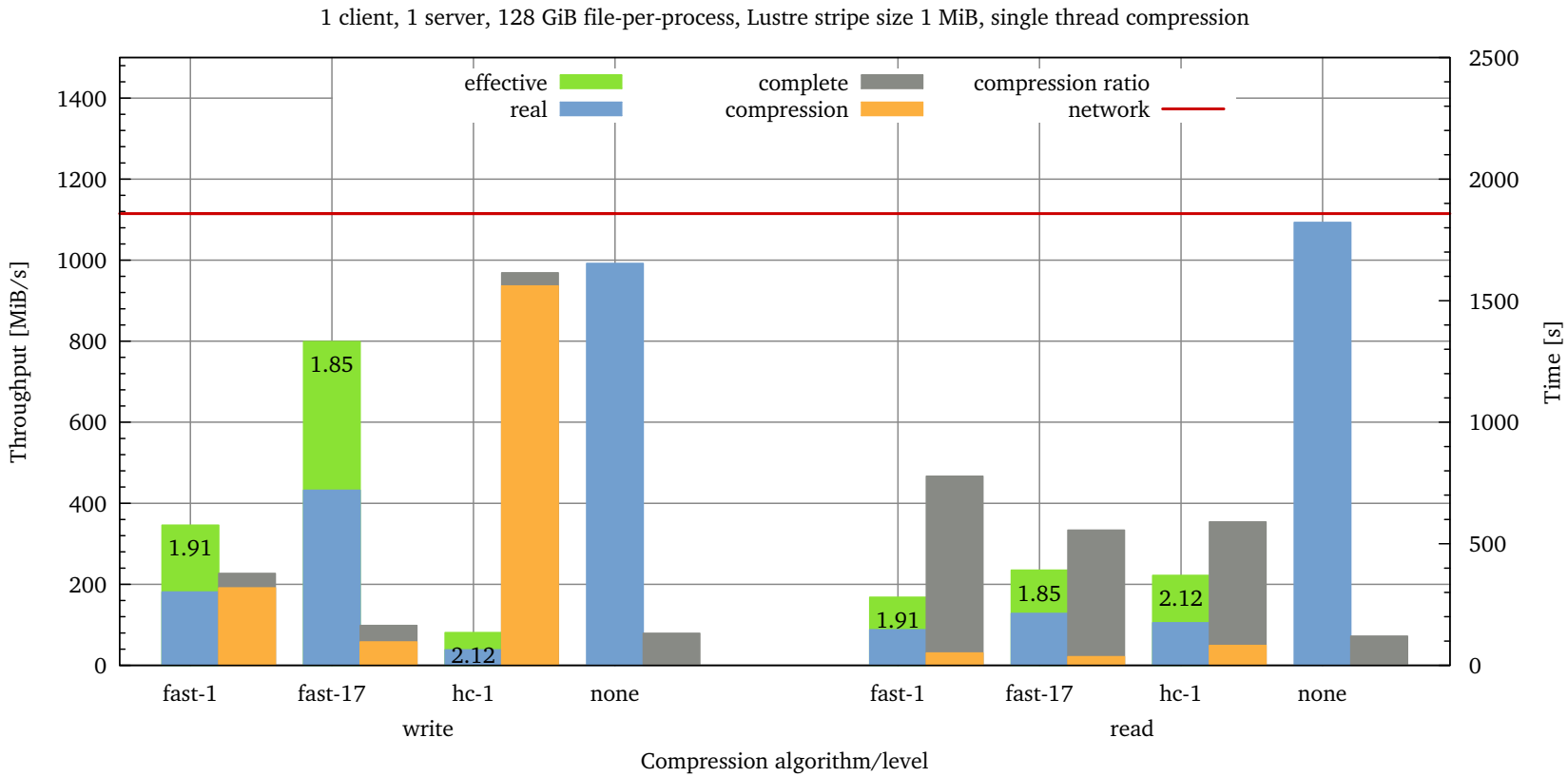
# Userspace – IOR Ethernet

- 10 Lustre clients, 10 servers, 240 files a 1 GiB, 1 MiB stripes, ZFS, Lustre 2.8
- Single thread compression with LZ4



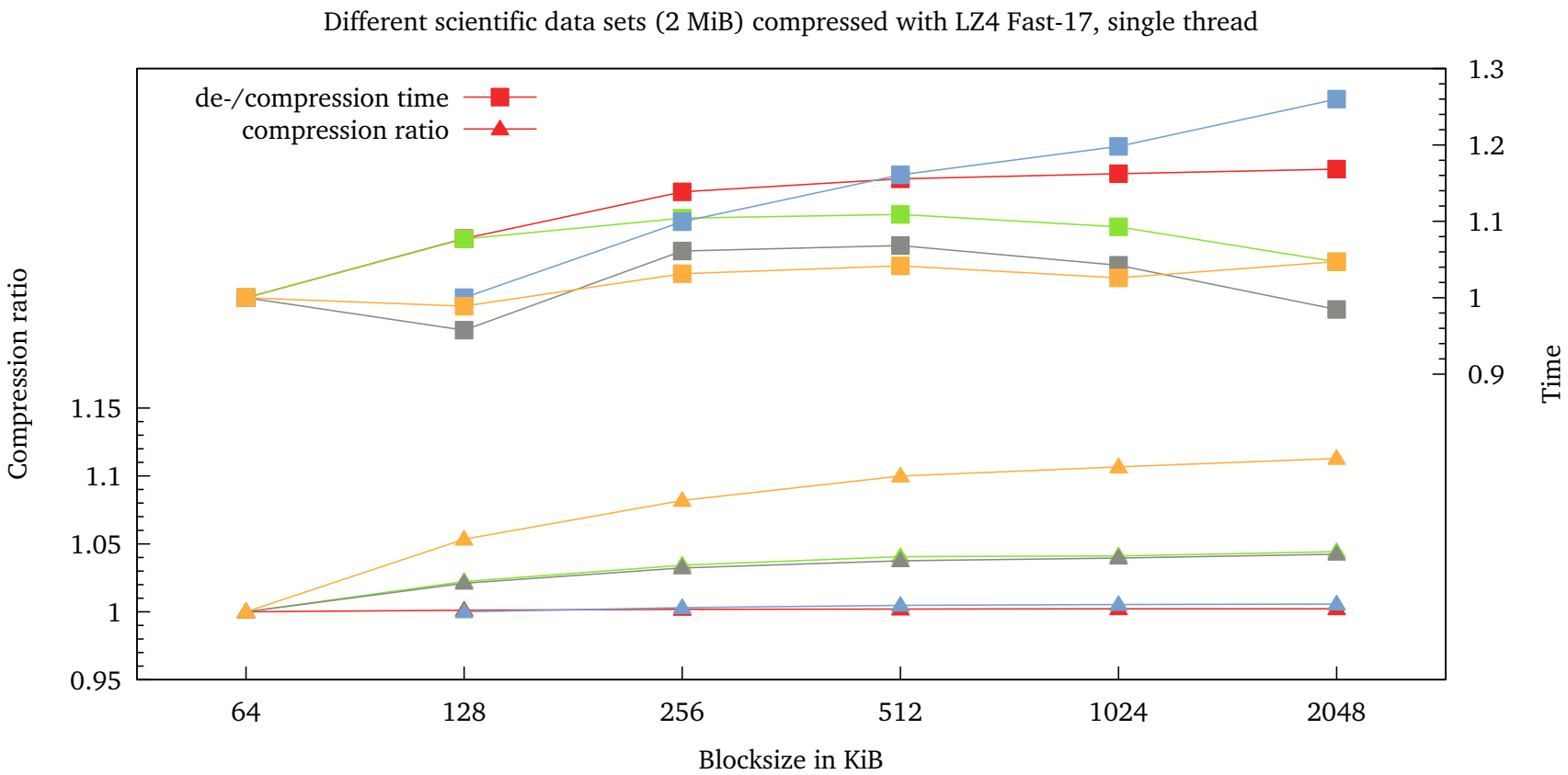
# Userspace – IOR IB

- Single stream 128 GiB, 1 MiB stripe, ldiskfs, Lustre 2.5
- Single thread compression insufficient for IB; very bad read



# Block sizes

- Times and ratios **normalized** to 64 KiB results
- Acceptable ratios with small blocks
- Similar results with LZ4-HC



- Duration June 2016 - February 2019 (expected)
- Current work focused on client-side compression
  - Already requires infrastructure changes on server's side
- Project at `http://wiki.lustre.org/Projects`

| Feature | Feature Summary | Point of Contact | Tracker | Target Date (YYYY-MM) |
|---|---|---|---|---|
| Enhanced Adaptive Compression in Lustre | Introduce compression for the Lustre client and server | Michael Kuhn (Universität Hamburg) | | 2019-02 |

- Duration June 2016 - February 2019 (expected)
- Current work focused on client-side compression
  - Already requires infrastructure changes on server's side
- Project at `http://wiki.lustre.org/Projects`

| Feature | Feature Summary | Point of Contact | Tracker | Target Date (YYYY-MM) |
|---|---|---|---|---|
| Enhanced Adaptive Compression in Lustre | Introduce compression for the Lustre client and server | Michael Kuhn (Universität Hamburg) | | 2019-02 |

- Additional student works (Master Theses)
  - Adaptive Compression for ZFS (finished, to be submitted)
  - Improving ZFS compression – more algorithms
  - Policy module for Lustre – make decisions dynamically
  - ZFS interface for transformed data