T10PI End-to-End Data Integrity Protection for Lustre

Shuichi Ihara, Li Xi

DataDirect Networks, Inc.

2018/04/25

Why is data Integrity important?

- **Data corruptions is painful!**
  - Frequency is low, but cost is very high.
  - A lot of unusual operations and step by step procedures to recover.
- **What causes data corruptions?**
  - Facility
  - Hardware include network
  - Software
  - Human errors

Type of data corruption

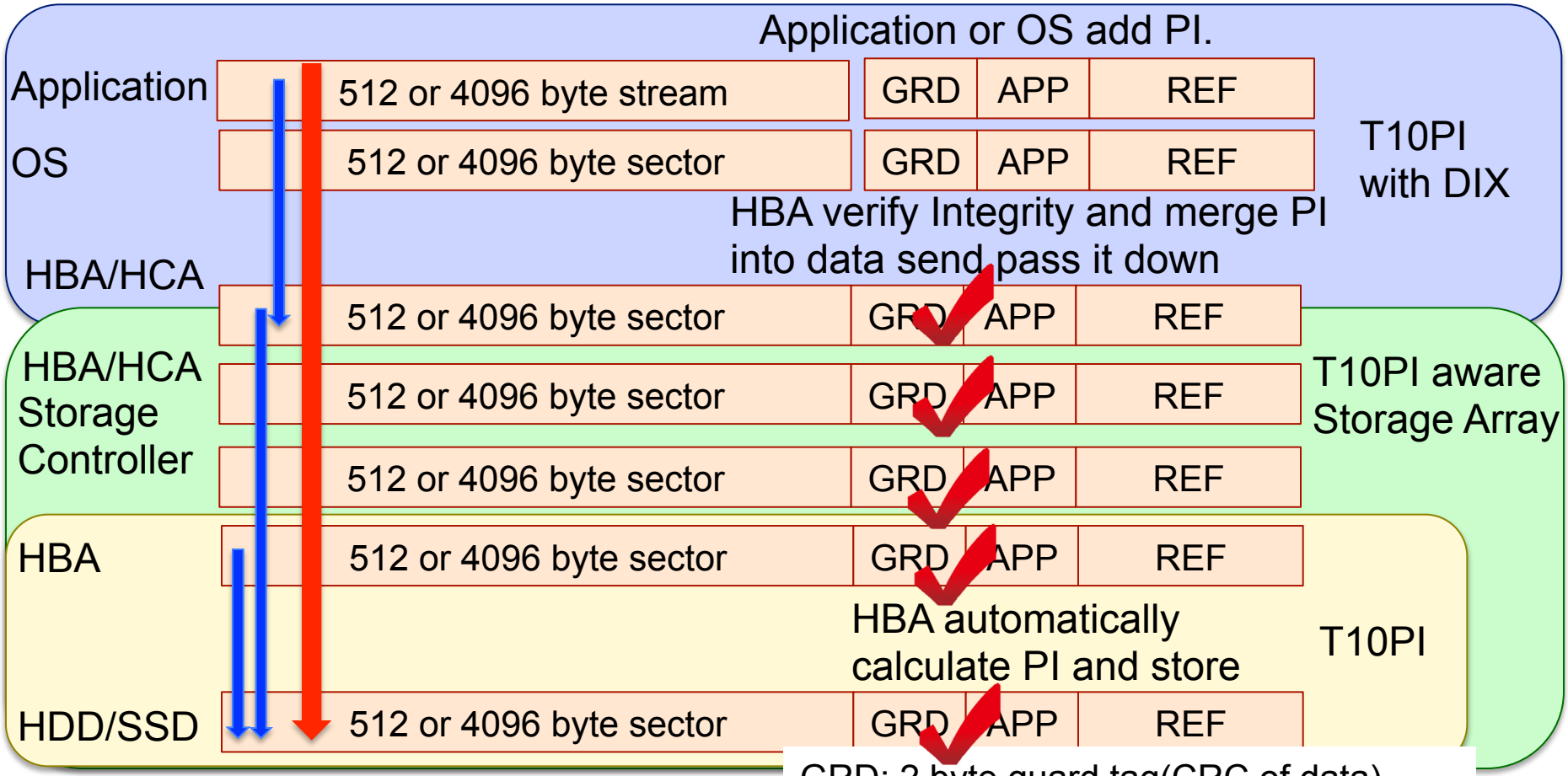
- **Two types of data corruption**
  - Latent sector/block errors
    - Application can't read sector/block and return an error.
  - Silent data corruption
    - Application can read sector/block, but it's NOT expected data and NOT valid data.
- **Silent data corruption causes another corruptions**
  - Application read data as expected and write new data based on it, but it's wrong!
- **Where/Why this happens?**
  - All storage stacks(App, OS, HBA, Storage Fabric/Array, Disk)
  - Lack of integrity check, each storage stack trusts upper/lower compartment.

Data Integrity of Lustre

- **Lustre checksum**
  - Checksum on between OSCs and OSTs.
  - Prevent server/client wrong RPC handling if it's corrupted.
  - No store checksums into Disks.
- **Backend Storage**
  - metadata checksum is available in Ext4, but not supported in Lustre.
  - ZFS has very strong mechanism for data Integrity
    - CoW, Transaction based, End-to-End checksum, Scrub, etc..
    - Data integrity inside ZFS.
- **Is this enough?**
  - Still missing guarantee on some places.
    - After sever received RPCs (e.g. Memory corruptions, OS to HBA to Storage Array, etc)
  - There was Lustre End-to-End Data Integrity discussion(LU-2584)
    - Proposed T10 PI/DIX support and submitted patches by Xyratex
    - Required to replace whole Lustre checksum with new T10PI/DIX checksum

T10PI(DIF) and DIX(Data Integrity Extensions)

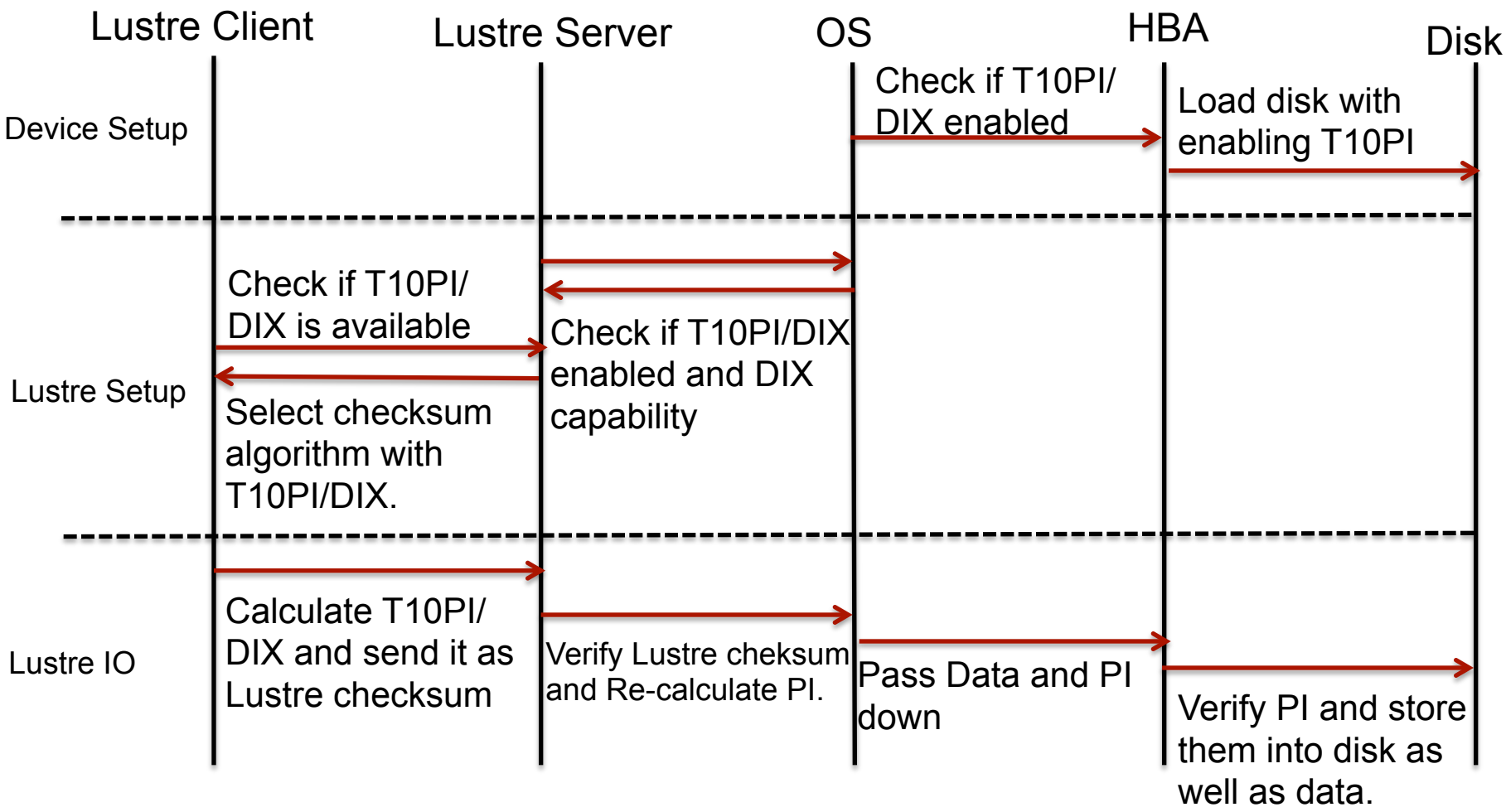
The standard specify an additional 8 byte field designated for data integrity/protection for each data block.



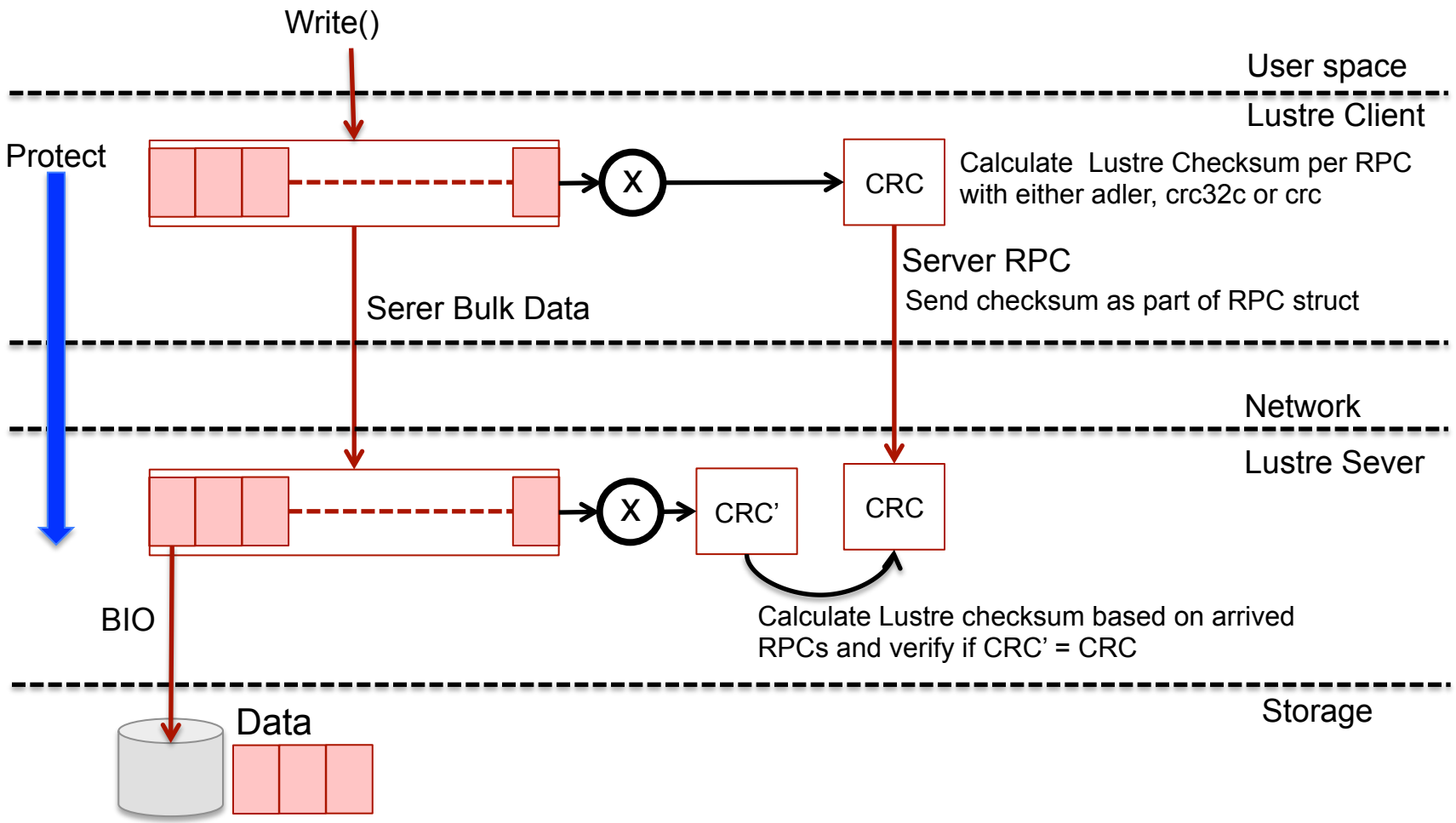
Proposed Design of Lustre End-To-End Data Integrity

- Fully transparent End-to-End Data integrity from Lustre client to disk.
- Relies on open standard format T10PI/DIX and any T10PI/DIX supported hardware work.
- Don’t change Lustre RPC format and extends current Lustre checksum framework.
- Consider minimum performance impacts.
- Keep compatibility for old Lustre version or non-T10PI supported hardware.

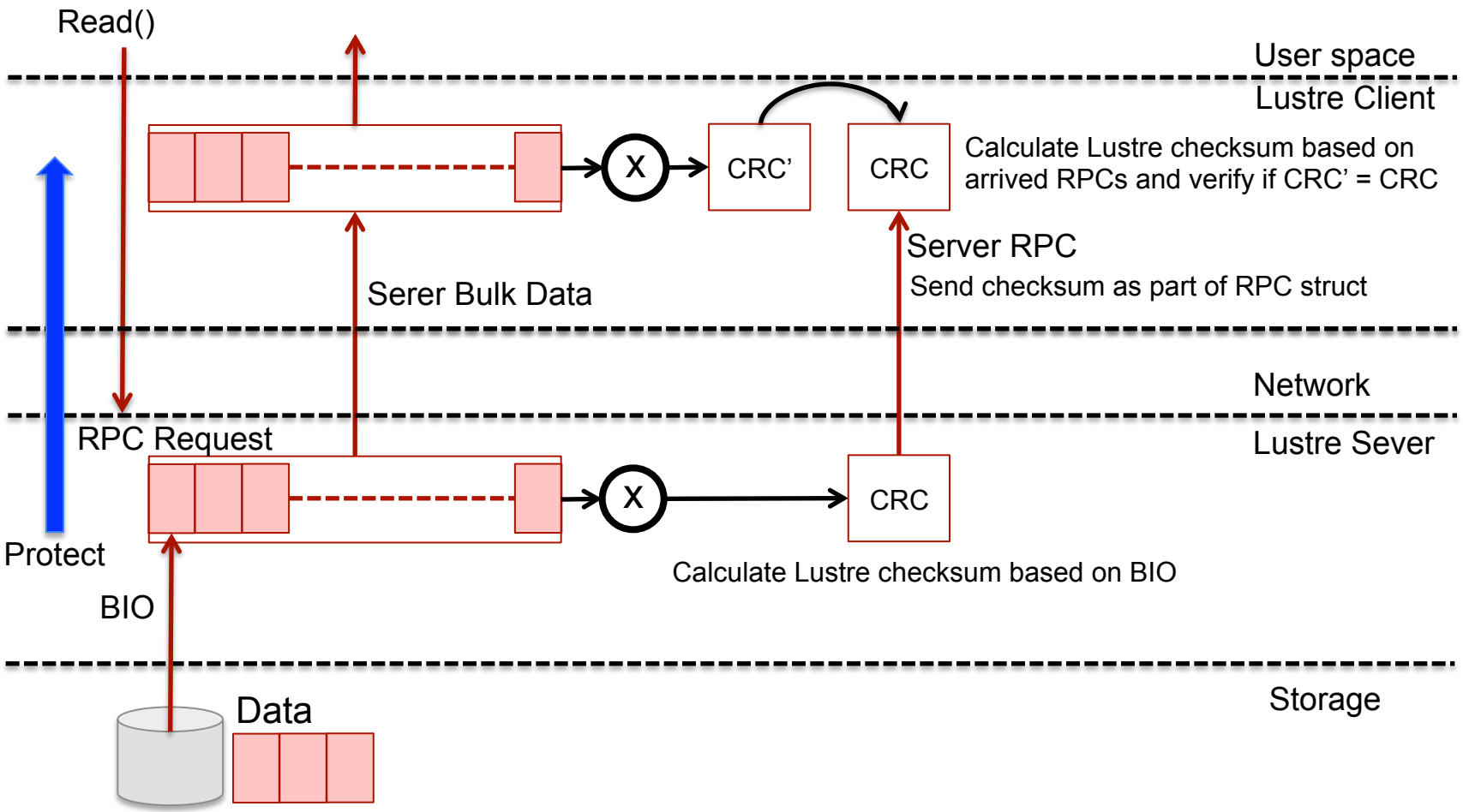
Basic flow of Lustre End-to-End Data Integrity



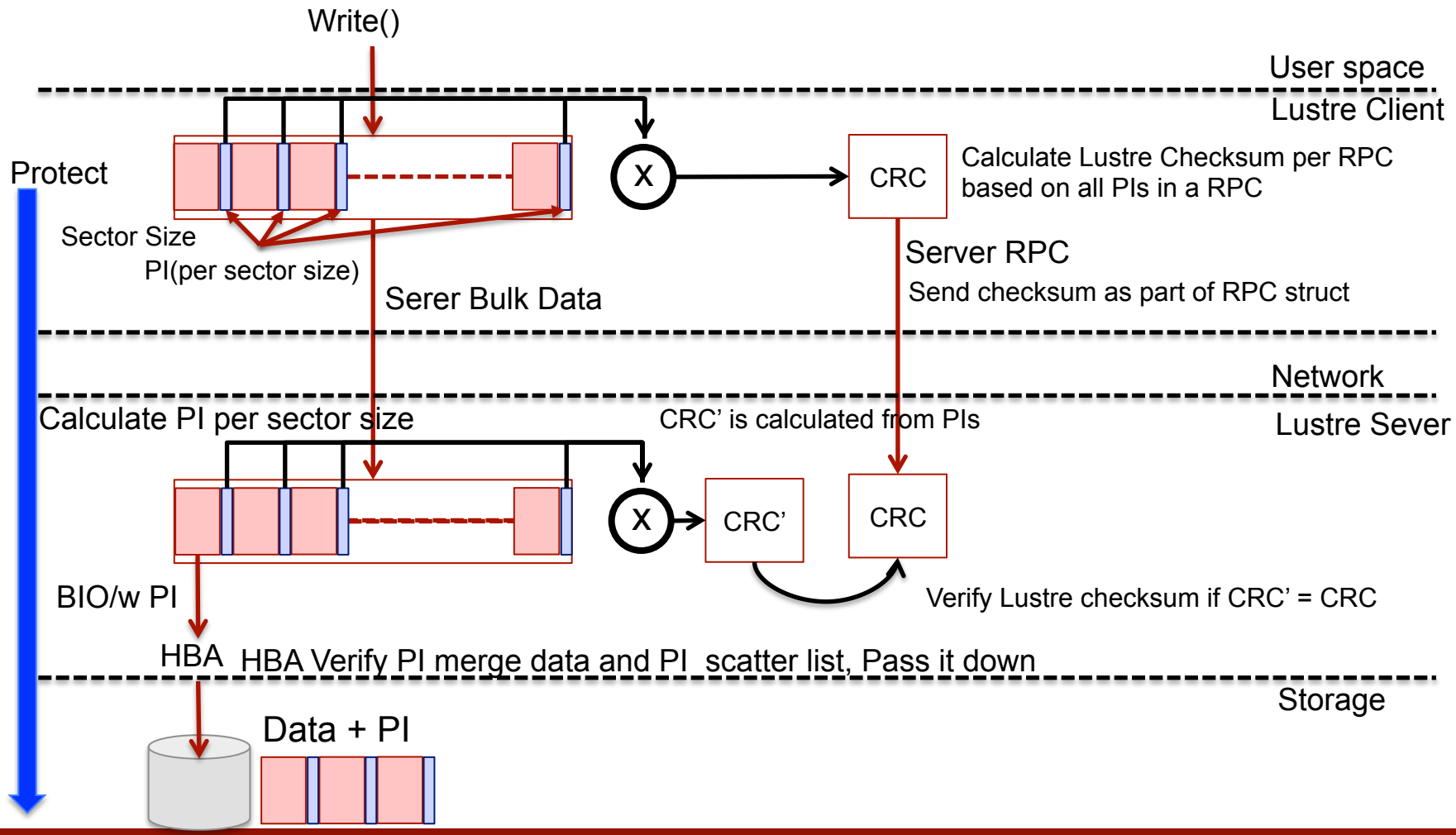
Today's Lustre checksum(Write)



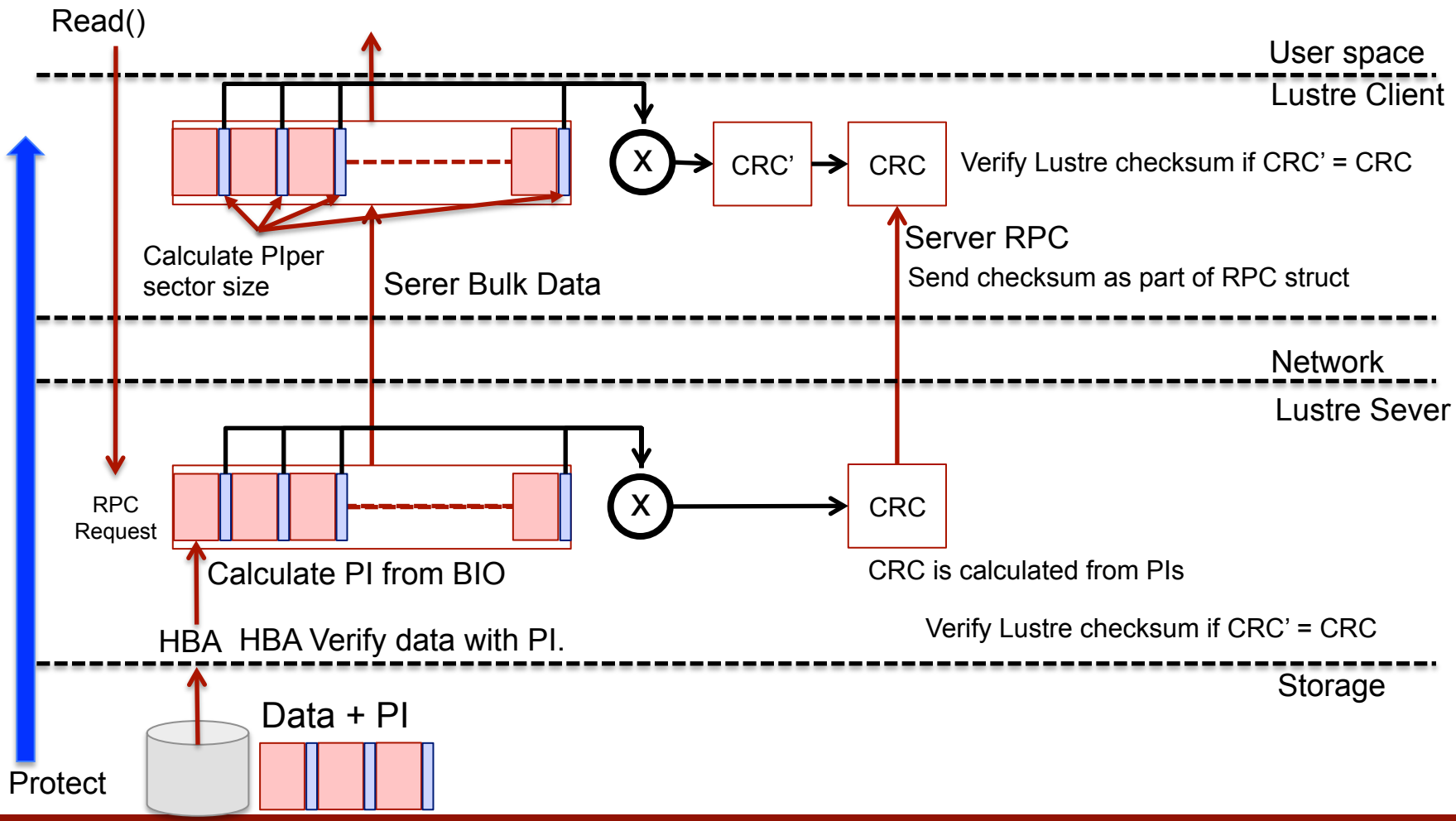
Today's Lustre checksum(Read)



Lustre checksum with T10PI/DIX for Enabling End-to-End Data Integrity(Write)



Lustre checksum with T10PI/DIX for Enabling End-to-End Data Integrity(Read)



Status

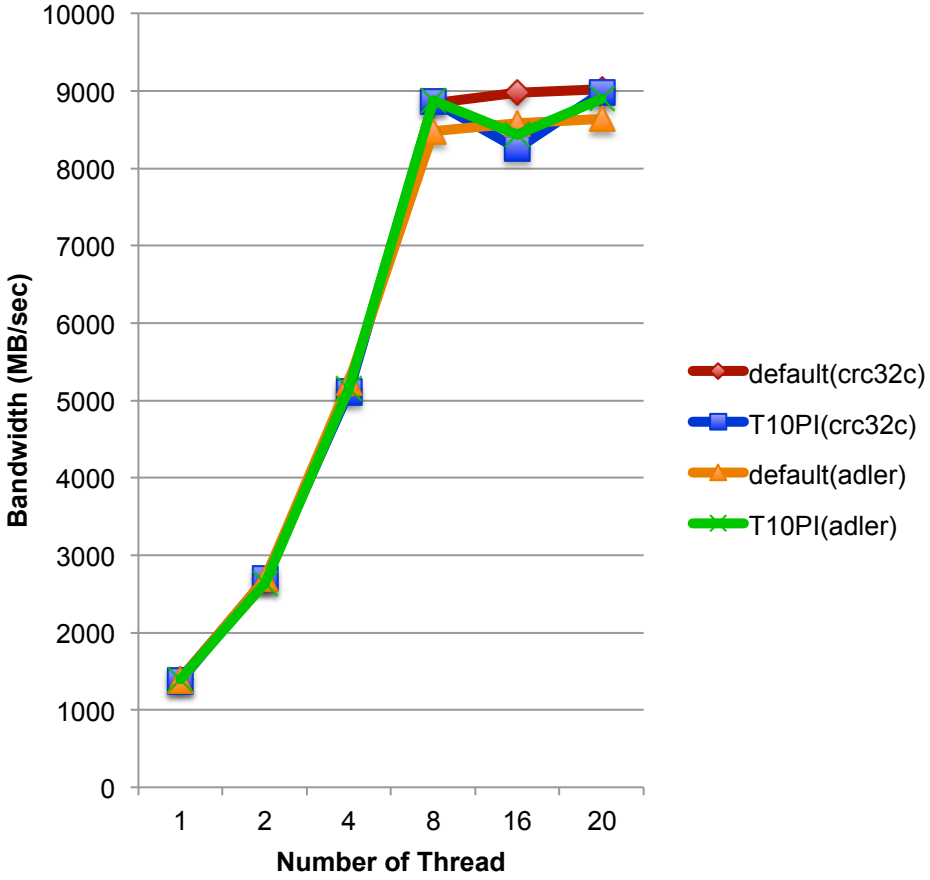
- **Task is tracked under LU-10472**
  - Patch being to submit for review
    - T10PI support for BIO https://review.whamcloud.com/#/c/31513
    - T10PI support for Lustre checksum (https://review.whamcloud.com/#/c/30980)
    - T10PI support for page cache (https://review.whamcloud.com/#/c/30792)
  - Cleanup and optimization are ongoing to finalize patches
- **Started function test and benchmark**
  - Adding test codes
  - Fault injection
  - Comparing performance against today's Lustre checksum

Test Environment

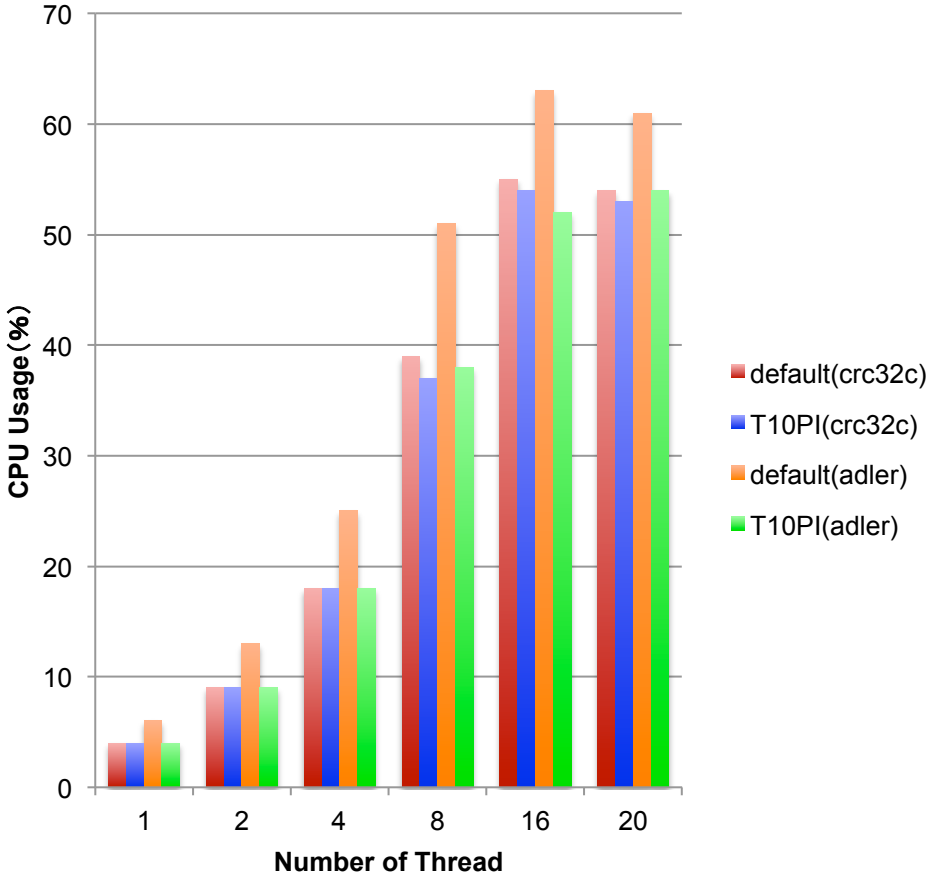
- **1 x MDS**
  - 2 x E5-2640v3, 256GB Memory, 1 x EDR Infibanind
  - 1 x LSI SAS3008(Enabled T10PI/DIX)
- **1 x OSS**
  - 2 x E5-2640v3, 256GB Memory, 1 x EDR Infibanind
  - 1 x LSI SAS3008(Enabled T10PI/DIX)
- **1 x SS8462**
  - 8 x NL-SAS and 2 x SAS disks connected to OSS/MDS with SAS
- **6 x Client**
  - 2 x E5-2660v3, 128GB Memory, 1 x EDR Infibanind
- **Use IOR with Lustre Fake-IO**

Performance Comparison – Single Client (FPP, Sequential, Write)

Single Client Performance(Write)

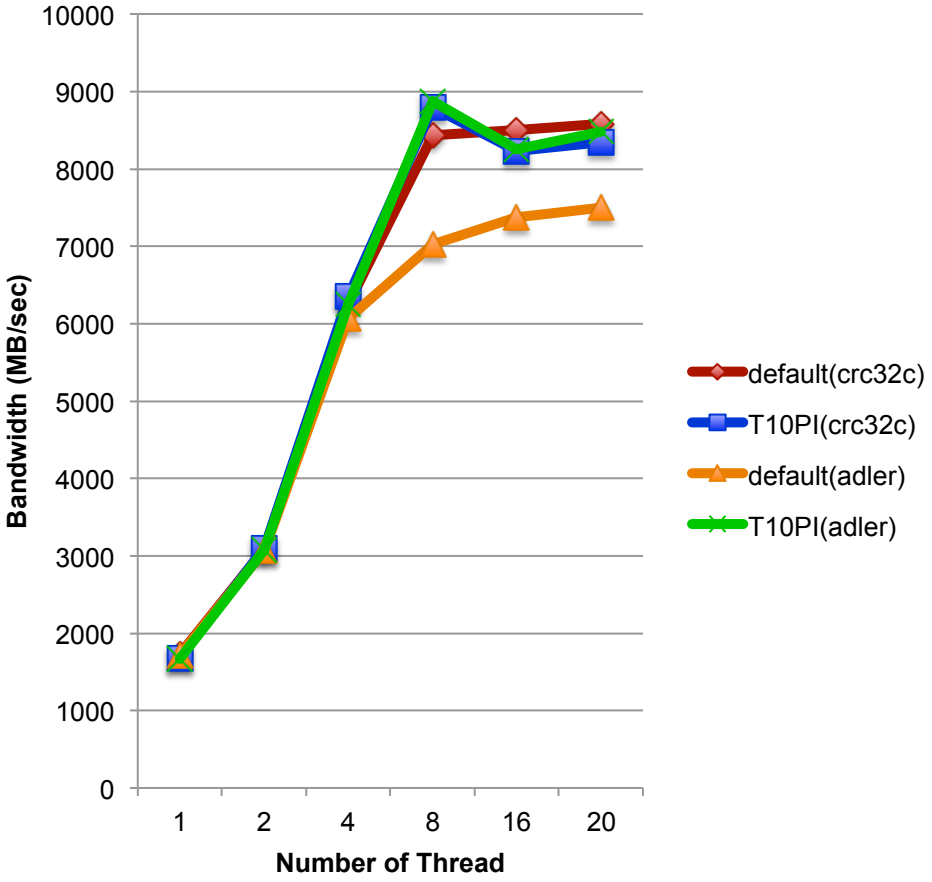


Client CPU Usage

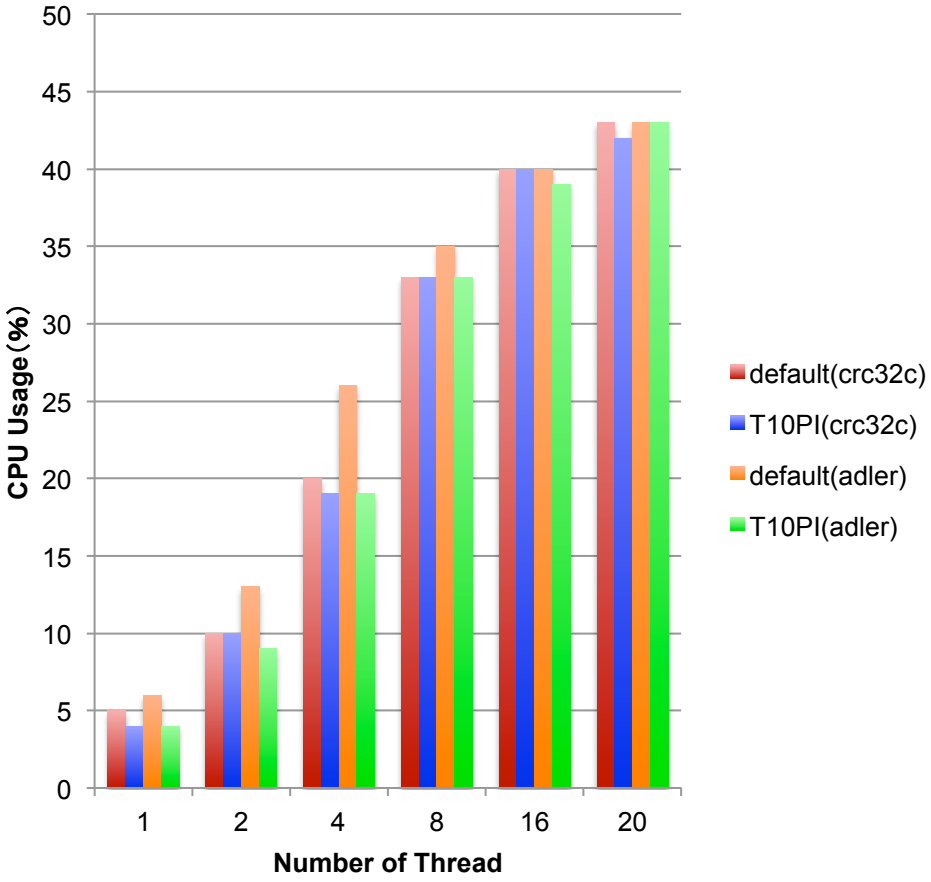


Performance Comparison- Single Client (FPP, Sequential, Read)

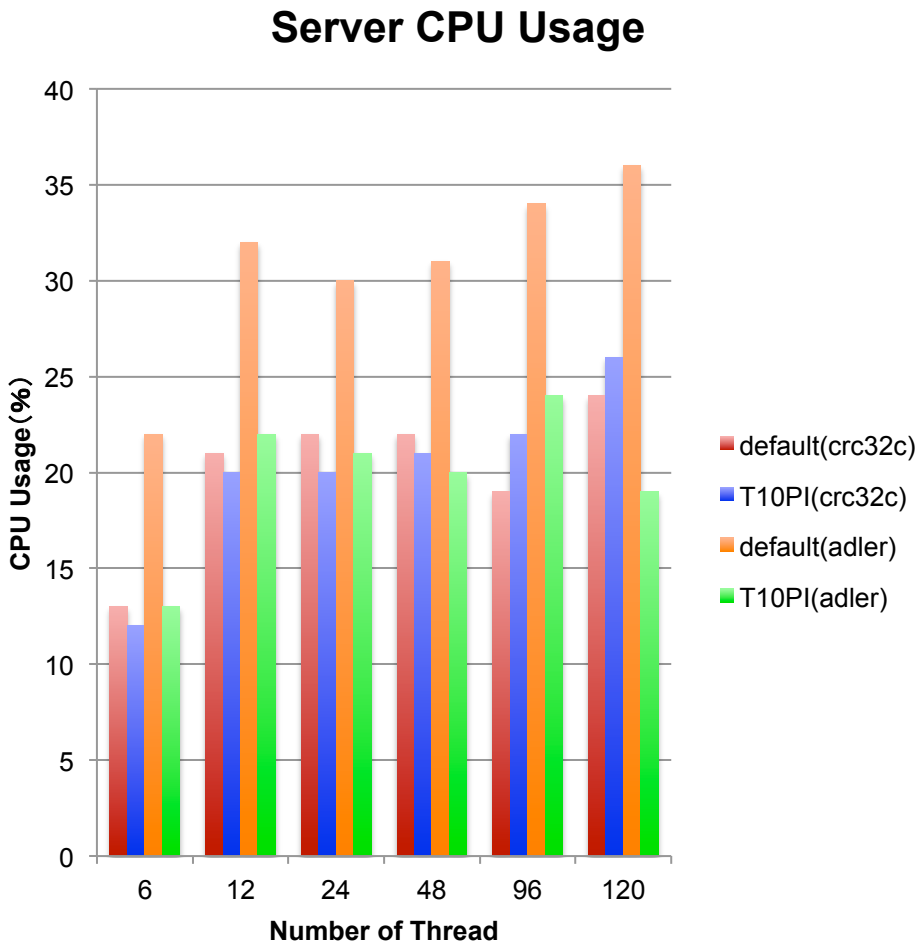
Single Client Performance(Read)



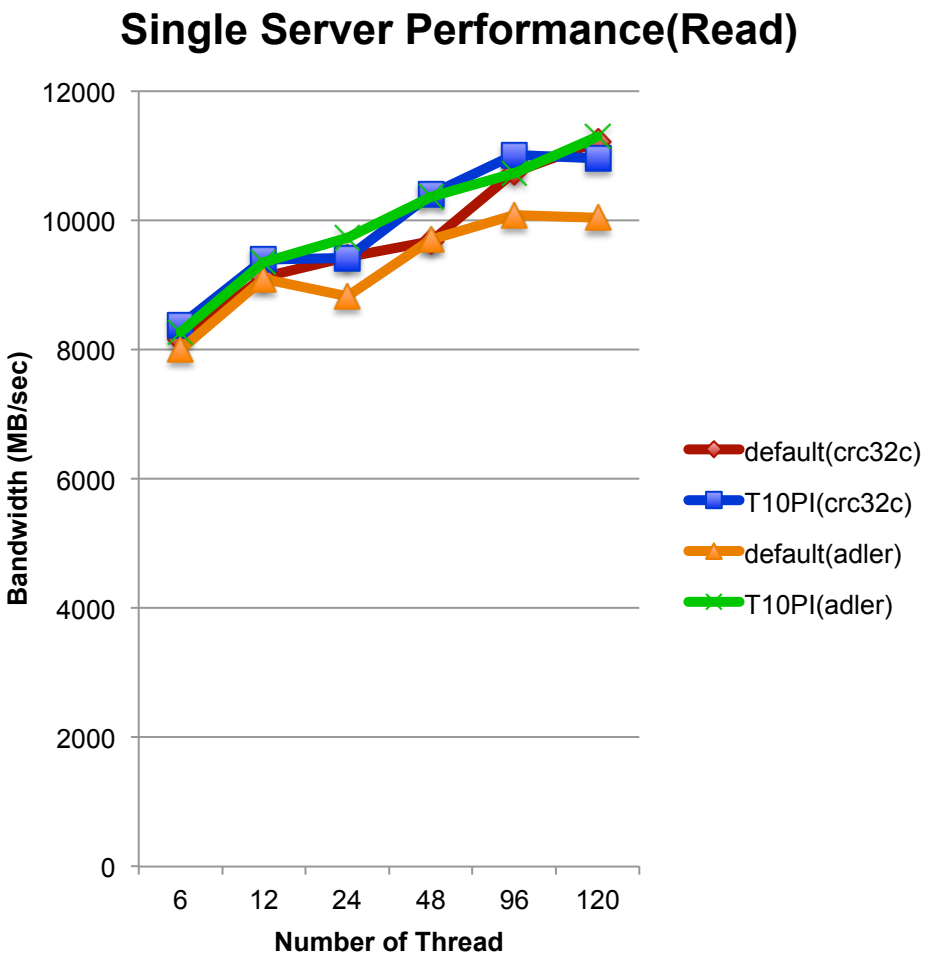
Client CPU Usage

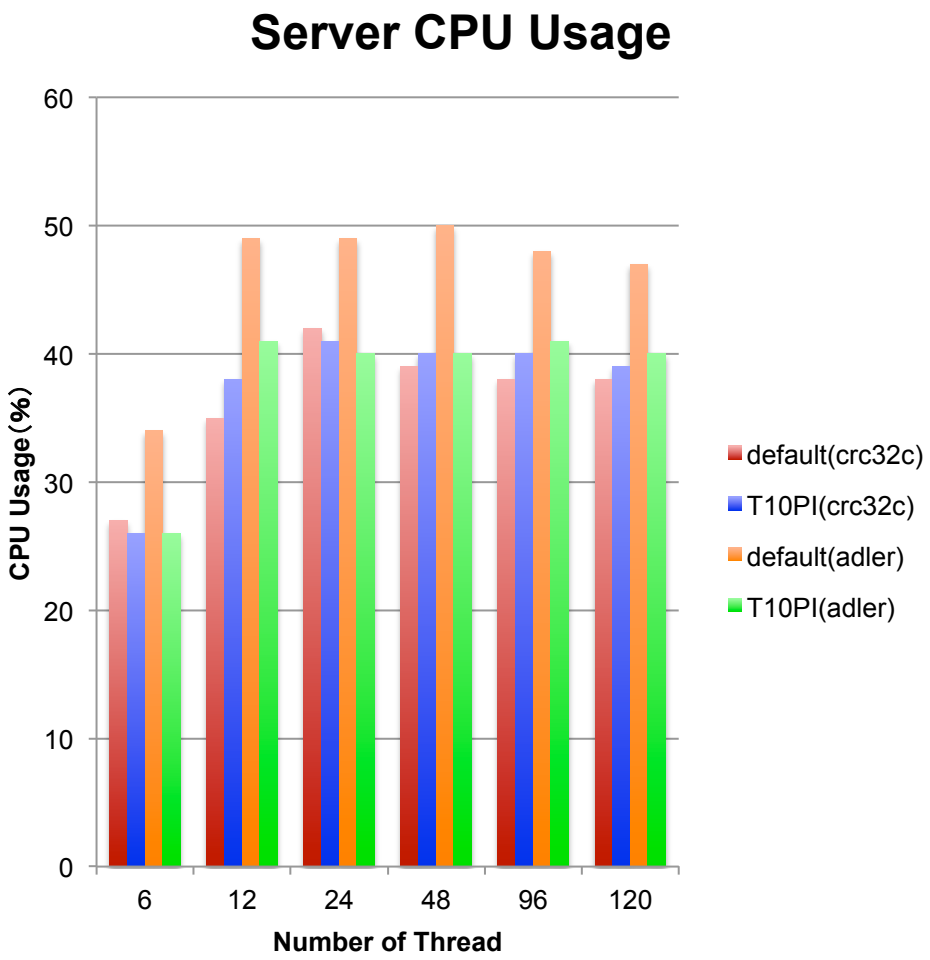


Performance Comparison - Multi Client/ Single Server(FPP, Sequential, Write)



Performance Comparison- Multi Client/ Single Server(FPP, Sequential, Read)





Conclusions

- **Designed Lustre End-to-End Data integrity**
  - Reused current Lustre checksum design and expended with T10PI/DIX.
  - Flexible and adaptable to any T10PI/DIX supported hardware and software.
  - Very minimum performance impacts.
- **Further Work**
  - Cleanup and shape the codes and add additional test codes.
  - Continue benchmark and test many failure scenarios on entire End-to-End comportment.