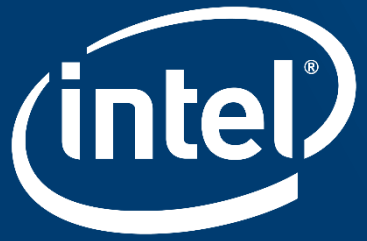


INTELLIGENT STORAGE ACCELERATION LIBRARY (ISA-L)

Jonathan Stern, Solutions Architect

Notices and Disclaimers

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

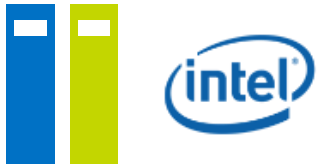
No computer system can be absolutely secure.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit **http://www.intel.com/performance**.

Intel® ISA-L **Value Proposition**

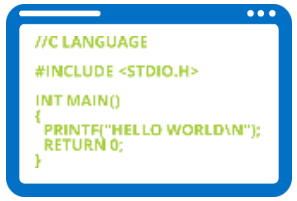
Algorithmic Library
for core storage algorithms where throughput and latency are the most critical factors

Optimized Libraries
for the fundamental building blocks of storage software on Intel® Architecture



Enhances Performance for data integrity, security/encryption, data protection, and compression algorithms

Single API call delivers the optimal implementation for past, present and future Intel processors



Validated on Linux*, BSD, and Windows Server* operating systems

Where is ISA-L used?

Open Source Projects

- Scale-out storage (HDFS*, Ceph* & Swift*)
- Streaming encryption (Netflix*)
- Deduplication software
- File systems

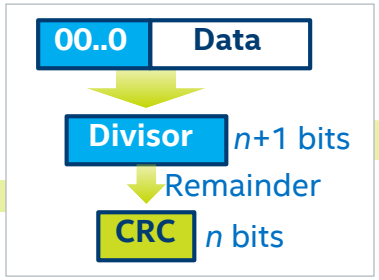
Proprietary Projects

- Hyperscale object storage
- Deduplication & backup solutions
- Multi-cloud backup
- Low-latency scale-up appliances

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

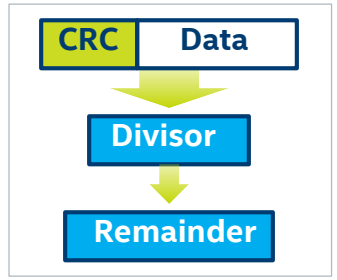
Intel® ISA-L **Functions**

XOR (RAID 5), **P+Q** (RAID 6), Reed-Solomon **Erasure Code**



Sender

CRC-T10, **CRC-IEEE** (802.3),
CRC32-iSCSI, **CRC64**



Receiver

DATA PROTECTION

DATA INTEGRITY

PERFORMANCE OPTIMIZING

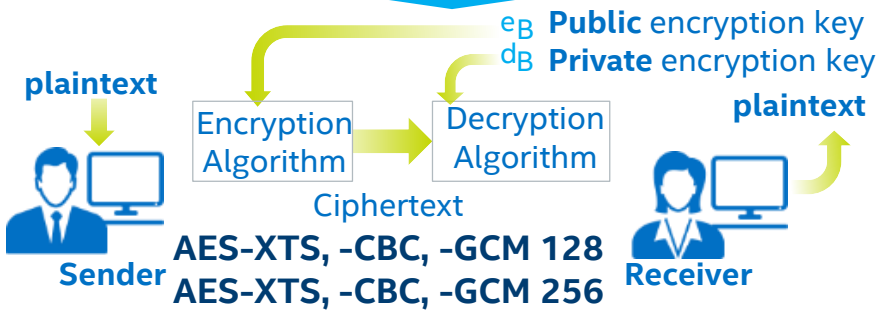
CRYPTOGRAPHIC HASHING

Multi-Buffer: SHA-1, SHA-256, SHA-512, MD5
Multi-Hash: SHA1, SHA1+Murmur

COMPRESSION "DEFLATE"

IGZIP: Faster DEFLATE (zlib)
Compression & Decompression

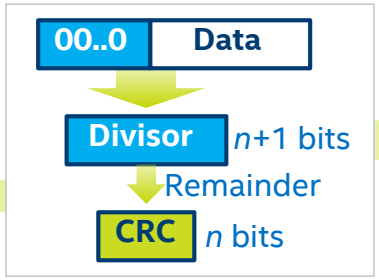
ENCRYPTION



AES-XTS, -CBC, -GCM 128
AES-XTS, -CBC, -GCM 256

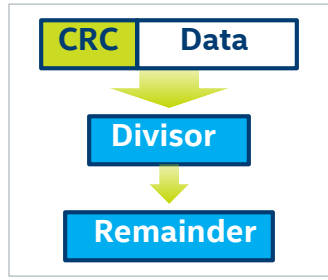
Intel® ISA-L **Functions: Compression**

XOR (RAID 5), **P+Q** (RAID 6), Reed-Solomon **Erasure Code**



Sender

CRC-T10, **CRC-IEEE** (802.3),
CRC32-iSCSI, CRC64



Receiver

DATA PROTECTION

DATA INTEGRITY

PERFORMANCE OPTIMIZING

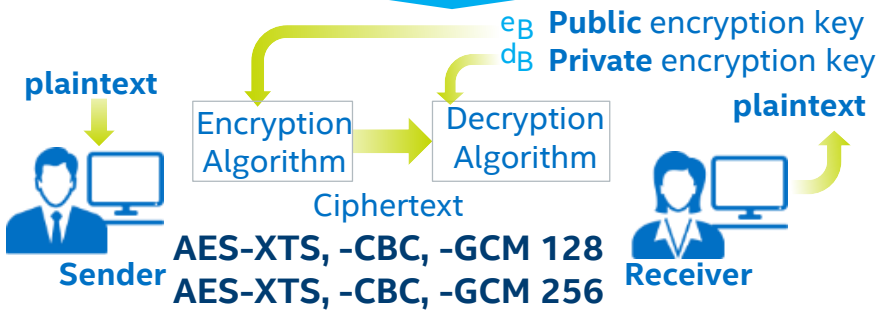
CRYPTOGRAPHIC HASHING

Multi-Buffer: SHA-1, SHA-256, SHA-512, MD5
Multi-Hash: SHA1, SHA1+Murmur

COMPRESSION "DEFLATE"

IGZIP: Faster DEFLATE (zlib)
Compression & Decompression

ENCRYPTION



AES-XTS, -CBC, -GCM 128
AES-XTS, -CBC, -GCM 256

IGZIP: What's Old Is New

DEFLATE (aka zlib, gzip, pkzip, etc)
- Lossless compression
- Ubiquitous adoption

v2.18: ISA-L Two-Pass IGZIP
- 5X greater throughput than zlib -1
- 13% better compression ratio than lz4 and lzo
- semi-dynamic compression

v2.17: Optimized Decompression
- >2X throughput vs. zlib, equal to lzo
- Fully compatible with zlib and gzip archives

Compressor Name	Compression Throughput (MB/s)	Ratio
lz4 1.7.3	287.1	52.0%
IGZIP 2.18 -1	**261.6**	**37.5%**
snappy 1.1.3	191.6	51.6%
zstd 1.1.1 -1	149.0	36.0%
brotli 0.5.2 -1	109.0	35.3%
zlib 1.2.8 -1	50.5	38.1%

Compressor Name	Decompression Throughput (MB/s)	Ratio
lz4 1.7.3	1662.32	52.0%
snappy 1.1.3	739.14	51.6%
zstd 1.1.1 -1	464.57	36.0%
IGZIP 2.18 -1	**362.16**	**37.5%**
brotli 0.5.2 -1	206.16	35.3%
zlib 1.2.8 -1	176.63	38.1%

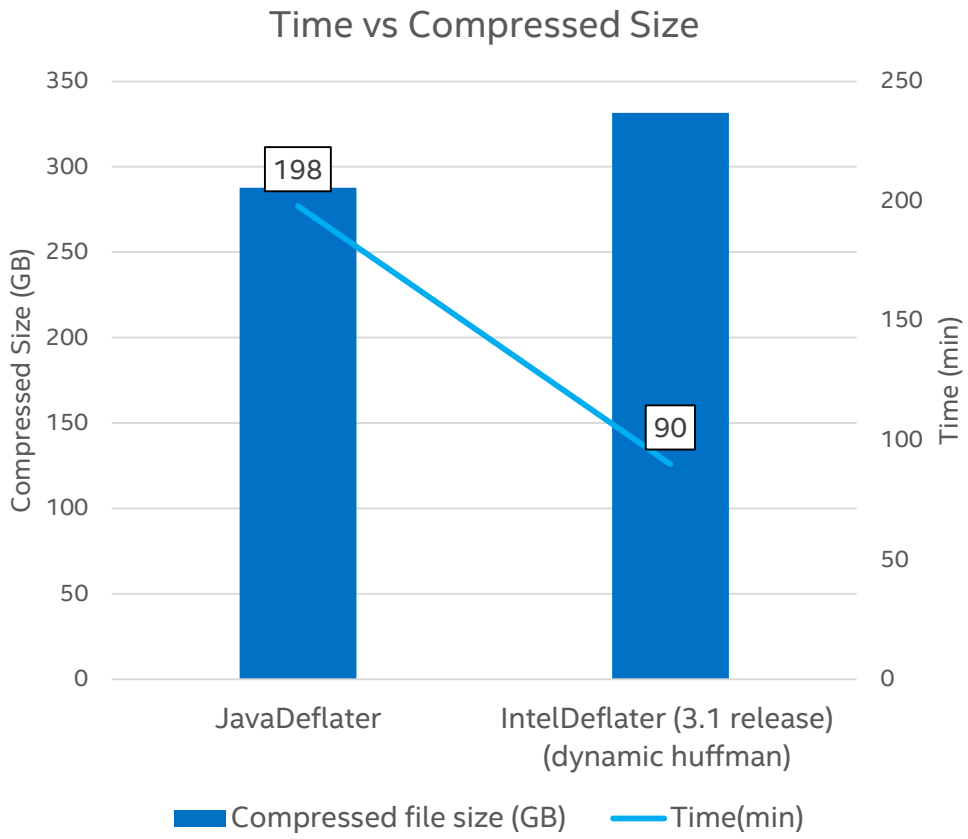
Hardware Configuration: Aztec City CRB, 2x Intel® Xeon® E5-2650v4, 4x 8GB DDR4 2400 MT/s, BIOS GRRFCRB1.86B.0276.R02.1606020546
BIOS configuration: Hyperthreading: disabled; Turbo Boost: disabled; Speed Step: disabled; P- and C-states: disabled. **Calgary Corpus, single core throughput.**

Case Study: Genome Analysis Tool Kit (GATK)

Genomics Data
- Huge volumes of data: 100s of GB per patient
- Cancer Cloud: for each cancer, 100k – 1M patients

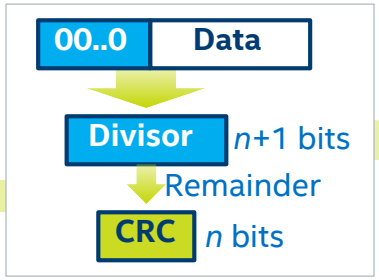
Economics
- Handling that volume is hard!
- DEFLATE great for sequenced genomes
- Industry tools reliant on zlib, usually Java

GATK Integration
- Throughput is essential
- Diverse hardware platforms



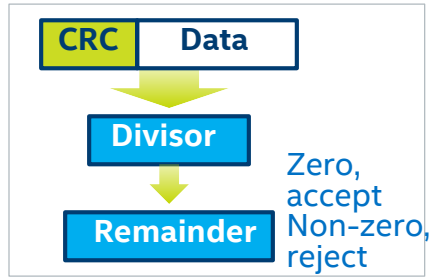
Intel® ISA-L **Functions: Hashing**

XOR (RAID 5), **P+Q** (RAID 6), Reed-Solomon **Erasure Code**



Sender

CRC-T10, **CRC-IEEE** (802.3),
CRC32-iSCSI, CRC64



Receiver

DATA PROTECTION

DATA INTEGRITY

PERFORMANCE OPTIMIZING

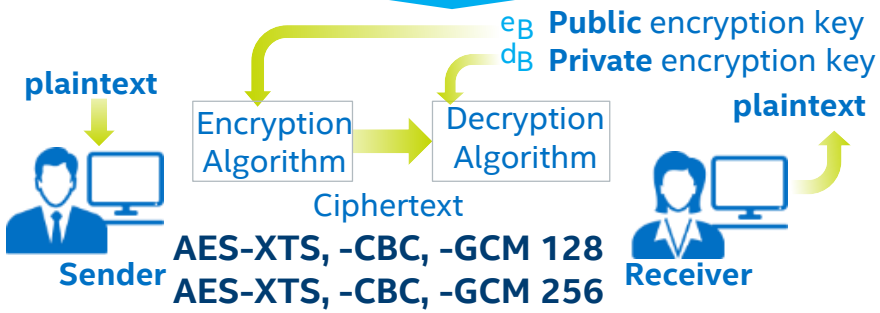
CRYPTOGRAPHIC HASHING

Multi-Buffer: SHA-1, SHA-256, SHA-512, MD5
Multi-Hash: SHA1, SHA1+Murmur

COMPRESSION "DEFLATE"

IGZIP: Faster DEFLATE (zlib)
Compression & Decompression

ENCRYPTION

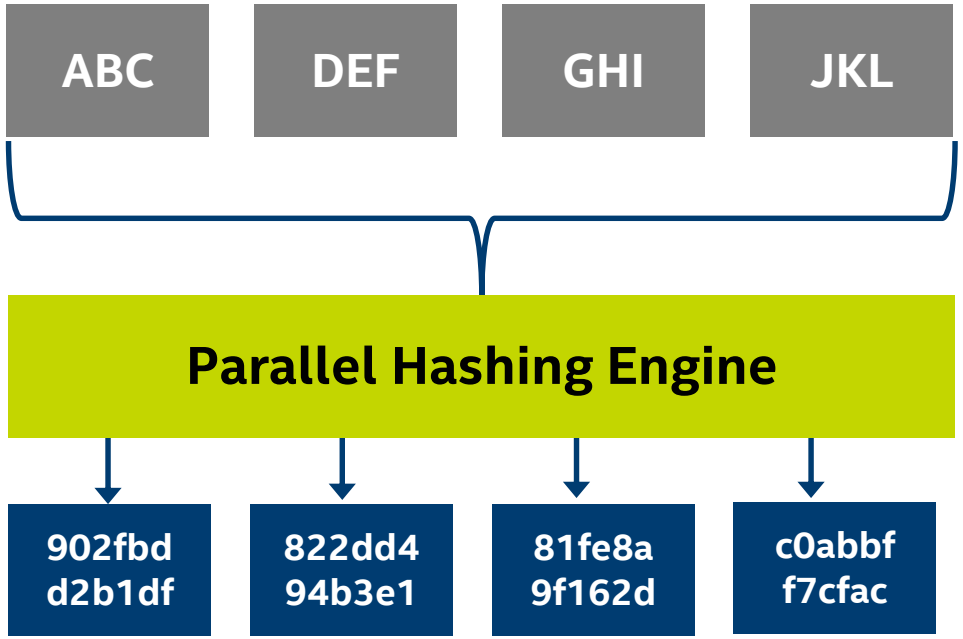


AES-XTS, -CBC, -GCM 128
AES-XTS, -CBC, -GCM 256

Multibuffer Hash

Citizens, Vectorize your Hashes!

- Uses AVX
- MD5, SHA1, SHA2-256, SHA2-512
- Asynchronous interface
- “Four for one”



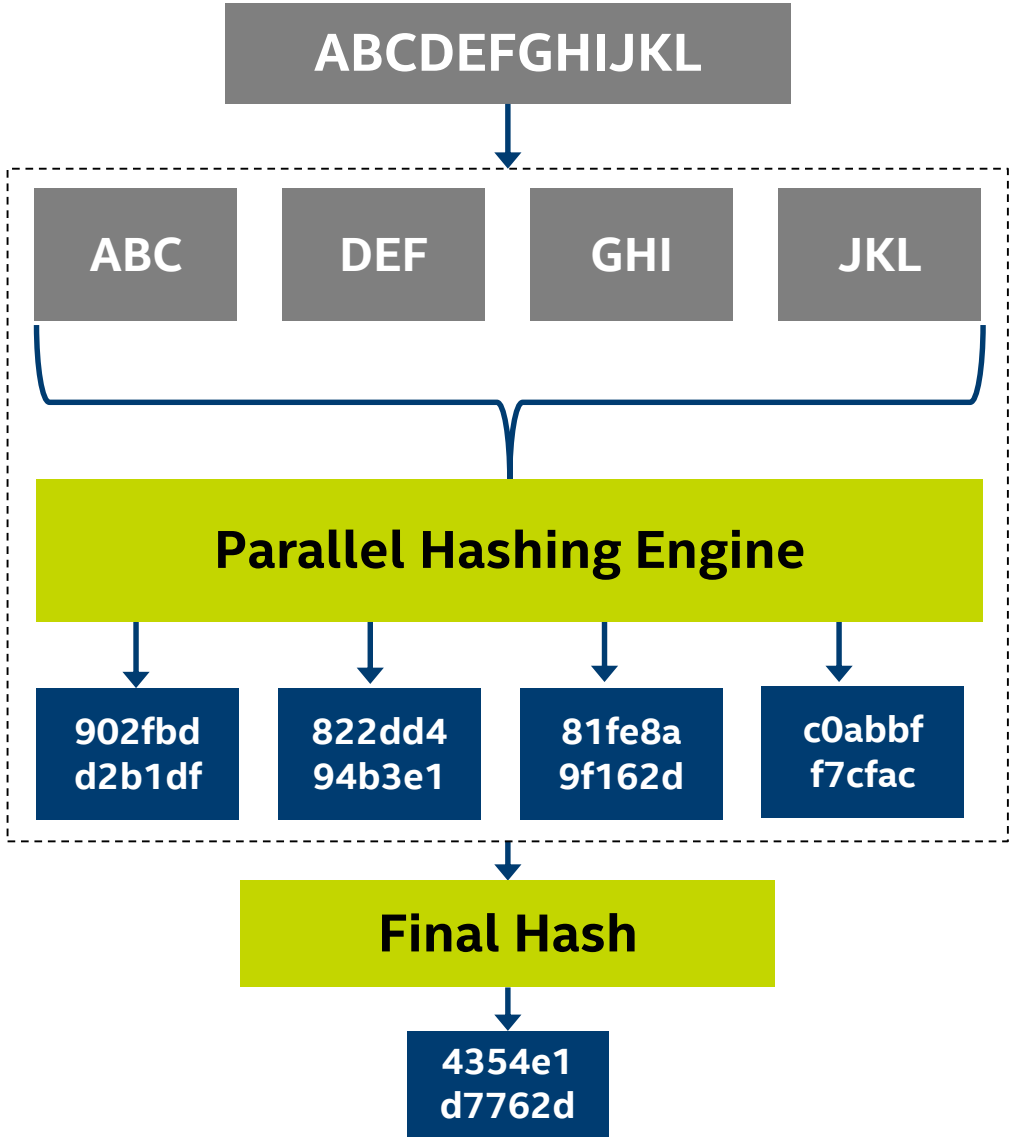
Multihash

What is ISA-L Multihash?

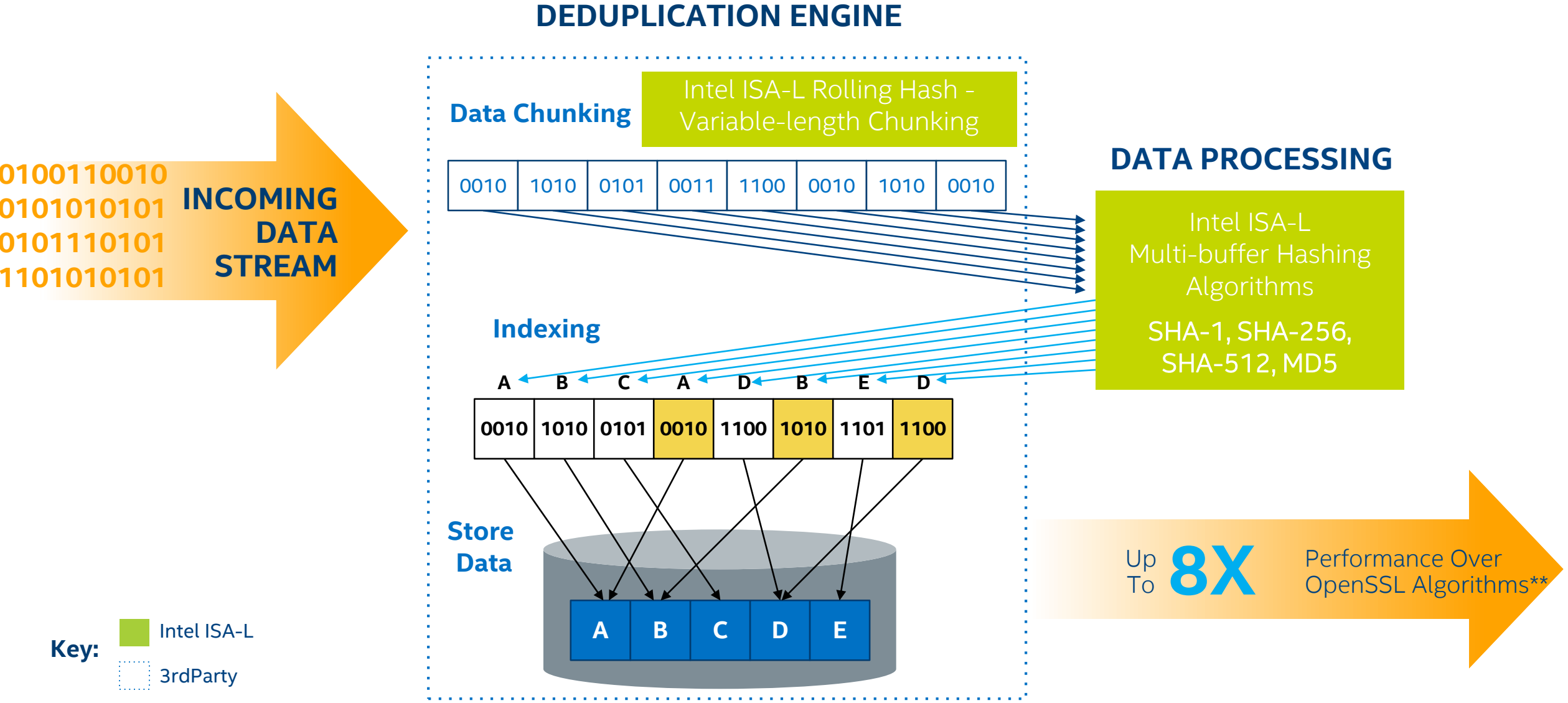
- Synchronous interface
- SHA1 != SHA1

Use Cases

- Data integrity
- Encryption
- Deduplication



Hashing Usage: Data Deduplication Optimizations

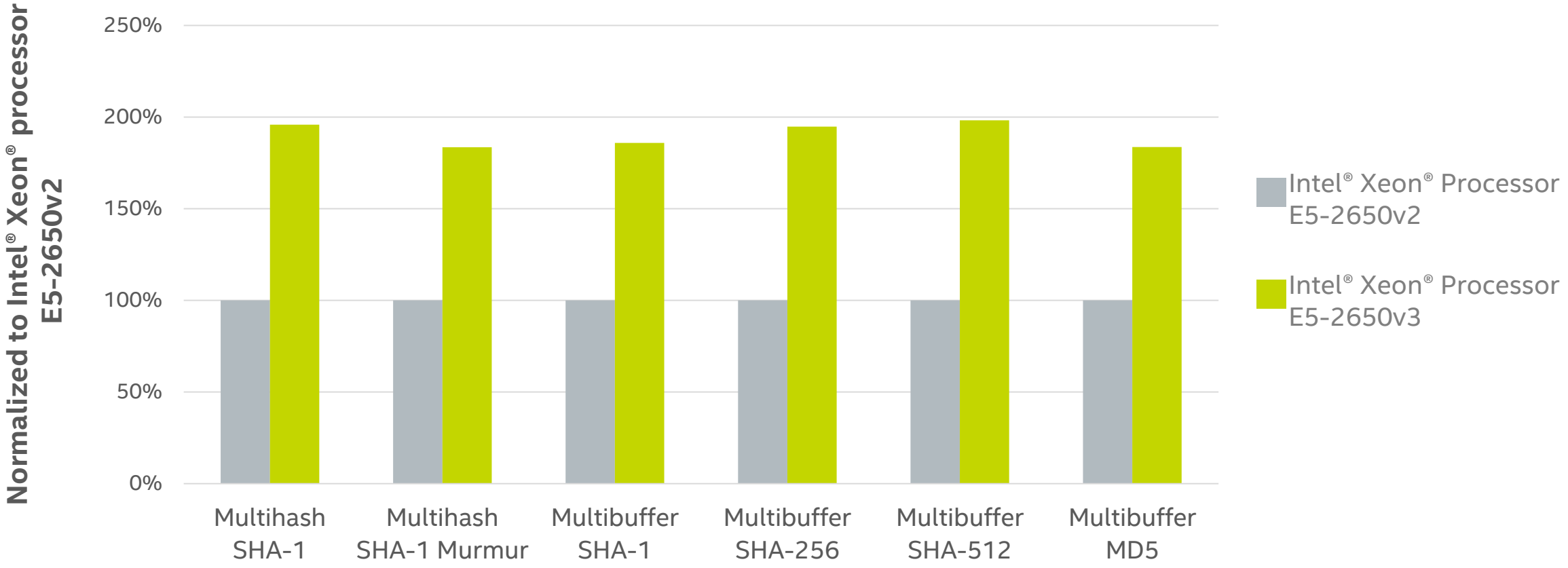


**** Hardware Configuration:** Aztec City CRB, 2x Intel® Xeon® E5-2650v4, 4x 8GB DDR4 2400 MT/s, BIOS GRRFCRB1.86B.0276.R02.1606020546
BIOS configuration: Hyperthreading: disabled; Turbo Boost: disabled; Speed Step: disabled; P- and C-states: disabled. **OpenSSL 1.0.2g MD5 hash**

Performance on the **Intel® Xeon® Processor**

Generational Cycle/Byte Comparison

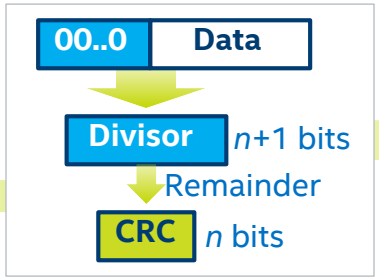
(higher is better)



E5-2560v2 Configuration: Rose City CRB, 2x Intel® Xeon® E5-2650v2, 4x 8GB DDR3 1600 MHz ECC RDIMM
E5-2650v3 Configuration: Aztec City CRB, 2x Intel® Xeon® E5-2650v2, 4x 8GB DDR4 2133 MHz ECC RDIMM
BIOS configuration: Hyperthreading: disabled; Turbo Boost: disabled; Speed Step: disabled; P- and C-states: disabled.

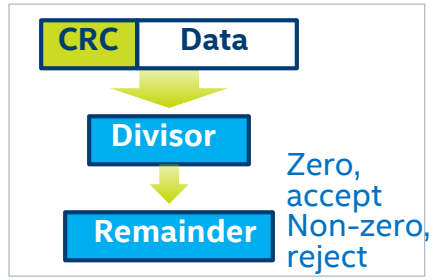
Intel® ISA-L **Functions: Encryption**

XOR (RAID 5), **P+Q** (RAID 6), Reed-Solomon **Erasure Code**



Sender

CRC-T10, **CRC-IEEE** (802.3), **CRC32-iSCSI**, **CRC64**



Receiver

DATA PROTECTION

DATA INTEGRITY

PERFORMANCE OPTIMIZING

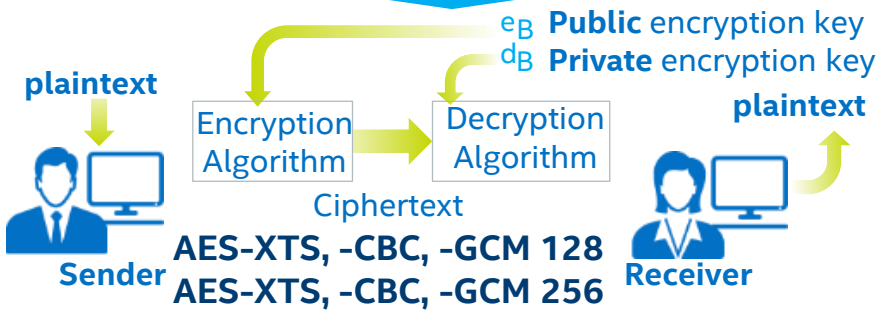
CRYPTOGRAPHIC HASHING

Multi-Buffer: SHA-1, SHA-256, SHA-512, MD5
Multi-Hash: SHA1, SHA1+Murmur

COMPRESSION "DEFLATE"

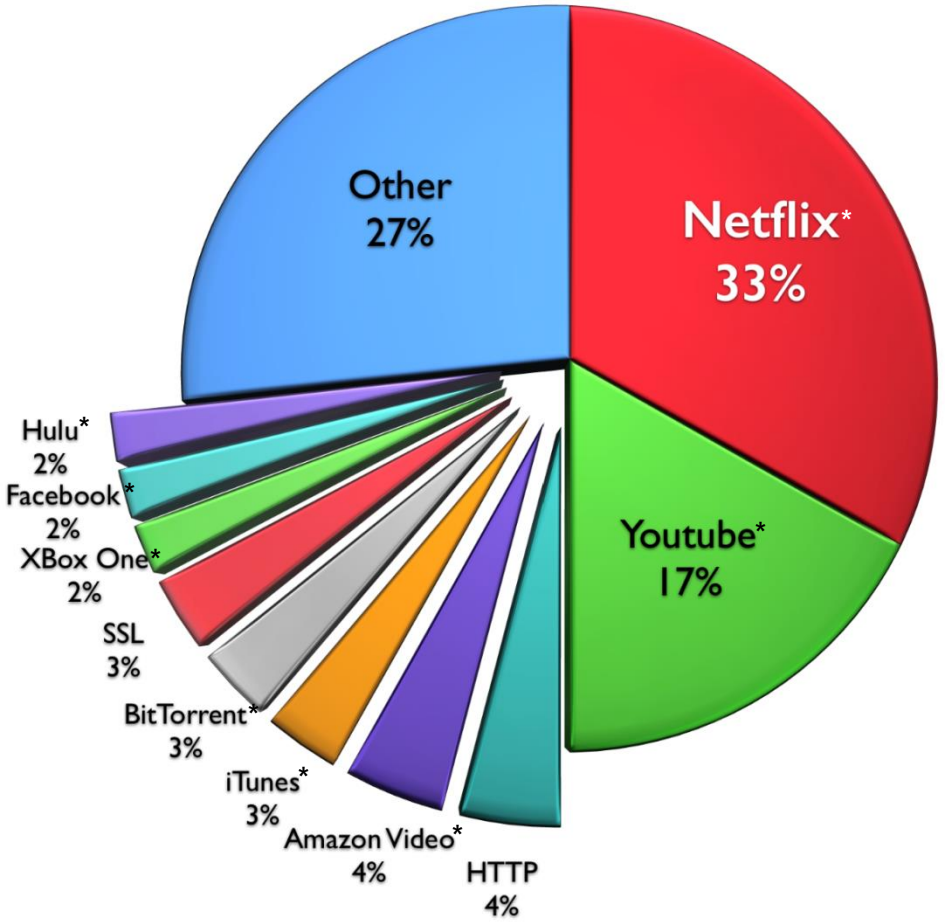
IGZIP: Faster DEFLATE (zlib)
Compression & Decompression

ENCRYPTION



AES-XTS, -CBC, -GCM 128
AES-XTS, -CBC, -GCM 256

Netflix* & Intel: Background



North American Aggregate Internet Traffic
Sandvine 2016 Global Internet Phenomenon Report
https://www.sandvine.com/trends/global-internet-phenomena

Netflix pushes how many bits?

- Average of 35Tb/s all day, every day, and rising

And how do they do it?

- Built their own custom Content Delivery Network
- Vast majority of the library is served from boxes living in your local ISP/IXP
- Heterogeneous hardware, but all single socket, all FreeBSD based

How come?

- Saves vast amounts of backbone traffic
- Gives Netflix direct control at both ends of the wire
- Improves user experience

The Challenge

Design Goal:
Upgrade to 100Gbps NIC per Open Connect Appliance

Curveball:
Add encryption (HTTPS/TLS) to streaming video for user privacy

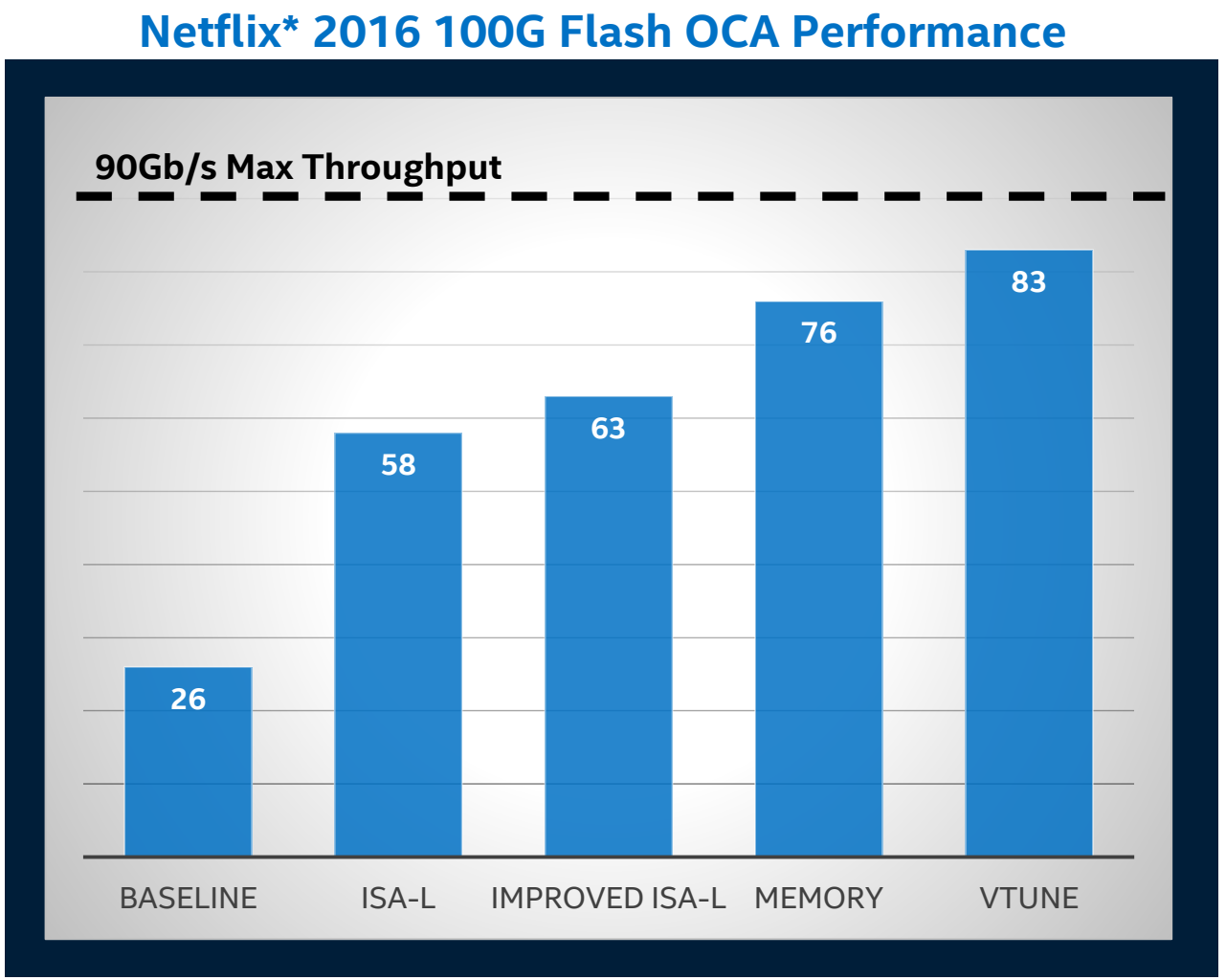
Budget:
Do it cost effectively

Before and After

Started with OpenSSL
- Required compromises in their data path

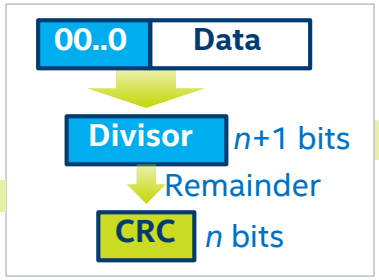
Tried all the alternatives: BoringSSL, etc
- ISA-L was the fastest on the market[2]
- Long-lived connections, only in the data path

ISA-L was tweakable
- Asked for non-temporal instructions: eureka!
- Identified the bottleneck: memory bandwidth
- Tuned the hardware
- … but it also fit the entire deployed infrastructure



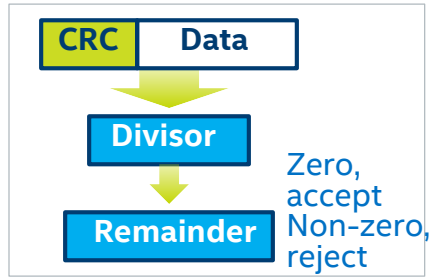
Intel® ISA-L **Functions: Erasure Coding**

XOR (RAID 5), **P+Q** (RAID 6), Reed-Solomon **Erasure Code**



Sender

CRC-T10, **CRC-IEEE** (802.3), **CRC32-iSCSI**



Receiver

DATA PROTECTION

DATA INTEGRITY

PERFORMANCE OPTIMIZING

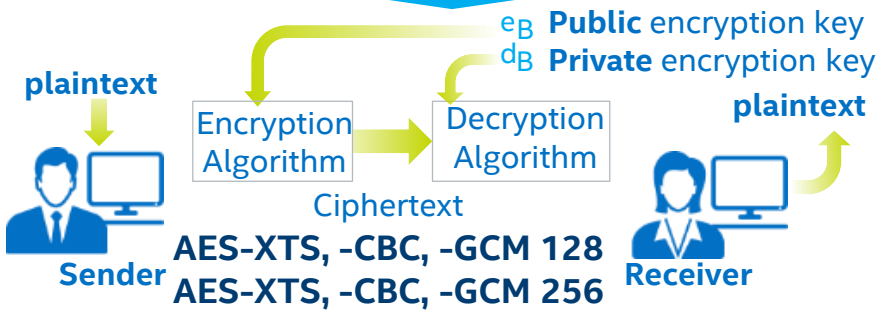
CRYPTOGRAPHIC HASHING

Multi-Buffer:SHA-1, SHA-256, SHA-512, MD5

COMPRESSION "DEFLATE"

IGZIP: Fast DEFLATE (zlib)

ENCRYPTION



AES-XTS, -CBC, -GCM 128
AES-XTS, -CBC, -GCM 256

ISA-L: Erasure Codes that Fly

Who is using Erasure Codes?

- "All the clouds" – distributed storage frameworks
- Hadoop HDFS, Ceph, Swift, hyperscalers…

Why are they using Erasure Codes?

- Irresistible economics: (at least) as much redundancy as triple replication with half the raw data footprint
- Half the storage media costs = big capex and opex savings

Why wasn't everyone using them before?

- Until ISA-L, EC was computationally prohibitive
- E5-2600v4, ISA-L can generate ~5GB/s of EC!

Integration Points

Debian* (as libisal2):
https://packages.debian.org/sid/libs/libisal2

Ceph*: ISA-L Erasure Code Integrated 2015
http://docs.ceph.com/docs/jewel/rados/operations/erasure-code-isa/

Swift*: Policies framework allows liberasure (ISA-L wrapper in Python)
http://docs.openstack.org/developer/swift/overview_erasure_code.html

HDFS*: ISA-L Erasure Code Patches in 3.0.0-alpha1, Compression in progress
https://issues.apache.org/jira/browse/HADOOP-11887
https://blog.cloudera.com/blog/2016/02/progress-report-bringing-erasure-coding-to-apache-hadoop/

ZFS*: Deduplication using ISA-L
http://www.snia.org/sites/default/files/SDC/2016/presentations/capacity_optimization/Xiadong_Qihau_Accelerate_Finger_Printing_in_Data_Deduplication.pdf

EFFICIENCY

Easing compute bottlenecks

SIMPLICITY

Low-level & easy to integrate

FLEXIBILITY

Native software-defined APIs

Intel® ISA-L: Learn More

- **License:** Algorithms are available under BSD license:
  https://github.com/01org/isa-l
  https://github.com/01org/isa-1_crypto
- **Customer Story - Netflix:** via BrightTalk
  "The Journey To Efficiently Securing Your Video Stream"
  https://www.brighttalk.com/webcast/10773/230519/
  [1] https://people.freebsd.org/~rrs/asiabsd_2015_tls.pdf
  [2] https://people.freebsd.org/~rrs/asiabsd_tls_improved.pdf
- **Detailed ISA-L Webinar:** via BrightTalk
  "Storage Algorithms Built for Speed"
  https://www.brighttalk.com/webcast/10773/179977
- **To use it:** see the included Getting Started Guide, API Guide, and C language reference applications.

BACKUP

ROADMAP

Design Considerations: **ISA-L vs. QAT**

Intel ISA-L

- does not consume PCIe lanes
- not hardware dependent: SW-defined apps can't assume platform
- "fast enough" throughput for certain performance targets
- latency savings of avoiding PCIe transaction for encryption/hash
- directional roadmap
- zero cost beyond CPU cores

Intel QAT

- huge advantage in high-throughput (>1GB/s) compression
- solid roadmap for both comms and storage use cases
- broad support of many protocols in HW
- stable API

ISA-L 5Q Roadmap

	Q4'16	Q1'17	Q2'17	Q3'17	Q4'17
Compress & Decompress	**Compression igzip** Increased to 5x performance over zlib, fast custom Huffman code generation **Decompression (NEW)** 2x performance over zlib	**Compression (NEW)** 2 Pass Compression, 5x faster than zlib-1 at same ratio			
Hashing		**Multi-Hash 256** 256 bit Output digest though-put performance increase over standard SHA256	**Multi-Buffer Hash** Denverton performance improvements for all multi-buffer hashing, perf TBD		
Data Integrity & Protection	**CRC64** Very large object CRC for multi-terabyte objects	**RAID** AVX512 Latency improvements for RAID5/6 calculations		**TBD** AVX512 Follow-on Updates →	
Crypto			**AES-GCM** Super small packet performance improvements		
Integration		**Hadoop** ISA-L Erasure Code Integration (~30x performance improvement, capacity ~doubles)	**Genomics Acceleration Tool Kit(GATK)** Igzip integration, ~50% improvement on object creation	**ISA-L Interface to QAT** Interface to QAT through ISA-L →	

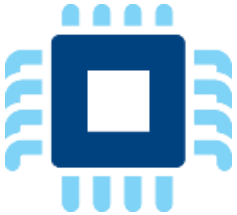
PERFORMANCE METRICS

Intel® ISA-L Performance **Overview**

Functional Library Comparisons

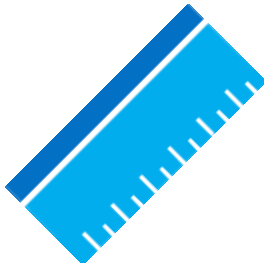
(performance vs. other libraries available)

- ISA-L 2.17
- OpenSSL 1.0.2g
- zlib 1.2.8



CPU Gen over Gen Performance

- Intel® Xeon® processor generation over generation performance metrics



Units of Measurement

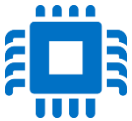
- Cycles/Byte
- Throughput (MB/s, GB/s)
- Calgary Corpus Weighted Ave
- Compression Ratio

Intel® ISA-L Performance **Overview**

Platform configuration details

Intel® Xeon® Processor E5-2600v4

- E5-2650v4, 12C, 2.2 GHz, M0
- Aztec City CRB
- 4x8 GB DDR4 2400 MT/s ECC RDIMM



BIOS Configuration

- **P-States:** Disabled
- **Turbo:** Disabled
- **Speed Step:** Disabled
- **C-States:** Disabled
- **Power Performance Tuning:** Disabled
- **ENERGY_PERF_BIAS_CFG:** PERF
- **Isochronous:** Disabled
- **Memory Power Savings:** Disabled

Cold Cache Tests

- Pick large data set by default (larger than last-level cache)
- Ensures memory fetch/put included

Turbo Off for Repeatability

Loop to Reduce Timer Latencies and Transients

- Start timer
- Iterate over data set
- Stop timer
- Report total bytes processed/time

Cycle/Byte Performance on the Intel® Xeon® Processor E5-2600v4 Product Family (cache cold cycle/byte)

ISA-L Function	Intel® Xeon® Processor E5-2650v4 @ 2.1 GHz 1 Socket			
	ISA-L		OpenSSL 1.0.2g	
	Cycle/Byte Performance (lower is better)	Single Core Throughput (higher is better)	Cycle/Byte Performance (lower is better)	Single Core Throughput (higher is better)
Rolling Hash 32 bit	4.16	529 MB/s	-	-
Rolling Hash 64 bit	2.67	823 MB/s	-	-
Multihash SHA-1	1.09	2.0 GB/s	-	-
Multihash SHA-1 Murmur	1.36	1.6 GB/s	-	-
Multibuffer SHA-1	1.14	1.9 GB/s	4.22	521 MB/s
Multibuffer SHA-256	2.62	840 MB/s	12.44	177 MB/s
Multibuffer SHA-512	3.26	676 MB/s	7.95	277 MB/s
Multibuffer MD5	0.61	3.5 GB/s	4.96	443 MB/s
AES-XTS 128	0.72	3.0 GB/s	0.86	2.5 GB/s
AES-XTS 256	0.93	2.3 GB/s	1.15	1.9 GB/s
AES-CBC 128 Decode	0.65	3.3 GB/s	0.81	2.7 GB/s
AES-CBC 192 Decode	0.76	2.8 GB/s	0.93	2.3 GB/s
AES-CBC 256 Decode	0.89	2.4 GB/s	1.06	2.0 GB/s
AES-GCM 128	0.80	2.7 GB/s	1.97	1.1 GB/s
AES-GCM 256	1.05	2.1 GB/s	2.26	973 MB/s

Up to 5X bandwidth boost

Up to 8X bandwidth boost

Cycle/Byte Performance on the Intel® Xeon® Processor E5-2600v4 Product Family (cache cold cycle/byte)

ISA-L Function	Intel® Xeon® Processor E5-2650v4 @ 2.1 GHz 1 Socket			
	ISA-L		OpenSSL 1.0.2g	
	Cycle/Byte Performance (lower is better)	Single Core Throughput (higher is better)	Cycle/Byte Performance (lower is better)	Single Core Throughput (higher is better)
PQ Gen (16+2)	0.11	19.0 GB/s	-	-
XOR Gen (16+1)	0.10	21.5 GB/s	-	-
Reed Solomon EC (10+4)	0.41	5.3 GB/s	-	-
CRC T10	0.18	12.0 GB/s		
CRC IEEE (802.3)	0.18	12.0 GB/s		
CRC32 iSCSI	0.18	11.7 GB/s		
CRC64 Normal	0.18	12.0 GB/s		
CRC64 Reflective	0.18	12.0 GB/s		
Compress - Stateless	7.86 CC WT AVE ratio 40.52 6.75 Silesia WT AVE ratio 41.35	280 MB/s 325 MB/s		
Decompress "Inflate"	6.07 CC WT AVE 5.20 Silesia WT AVE	362 MB/s 422 MB/s		

Cycle/Byte Performance (lower is better)	Single CoreThroughput (higher is better)
zlib 1.2.8 - Deflate	
50.89 CC WT AVE ratio 39.24% 48.59 Silesia WT AVE ratio 38.33%	43 MB/s 45 MB/s
zlib 1.2.8 - Inflate	
12.48 CC WT AVE 12.04 Silesia WT AVE	176 MB/s 182 MB/s