

Arquitecturas para tratar grandes volúmenes de información

Procesamiento de Datos a Gran Escala

Arquitecturas para tratar grandes volúmenes de información

Tema1 Arquitecturas de referencia para Big Data

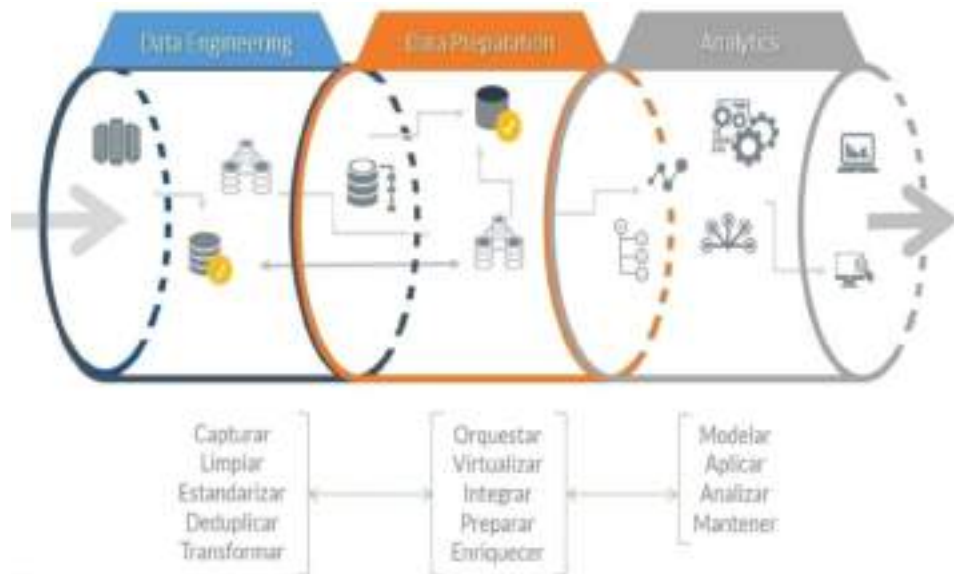
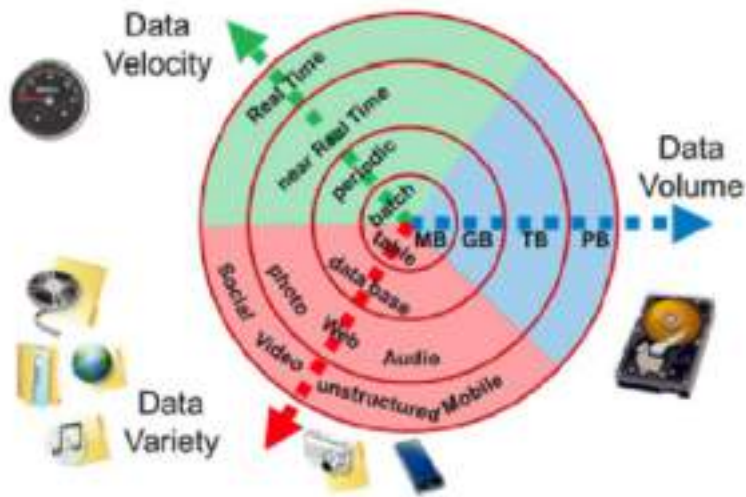
- Necesidades de los entornos de procesamiento para Big Data.
- Infraestructura: servidores físicos (On Premise) versus virtual (Cloud)
- Elementos básicos: CPUs, almacenamiento, interconexión, GPUs, coprocesadores.
- Optimización para sistemas que tratan grandes volúmenes de información.
- Nuevas tendencias de computación.
- Casos de estudios: optimizando el rendimiento.

Necesidades de los sistemas para Big Data

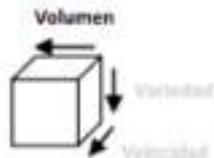
Infraestructura que procesen:

- Gran volumen de información.
- Con variedad de datos.
- Velocidad de llegada.

Gestionando toda la vida del dato desde su **captura/preparación/enriquecimiento** hasta su **análisis/modelización/mantenimiento**.



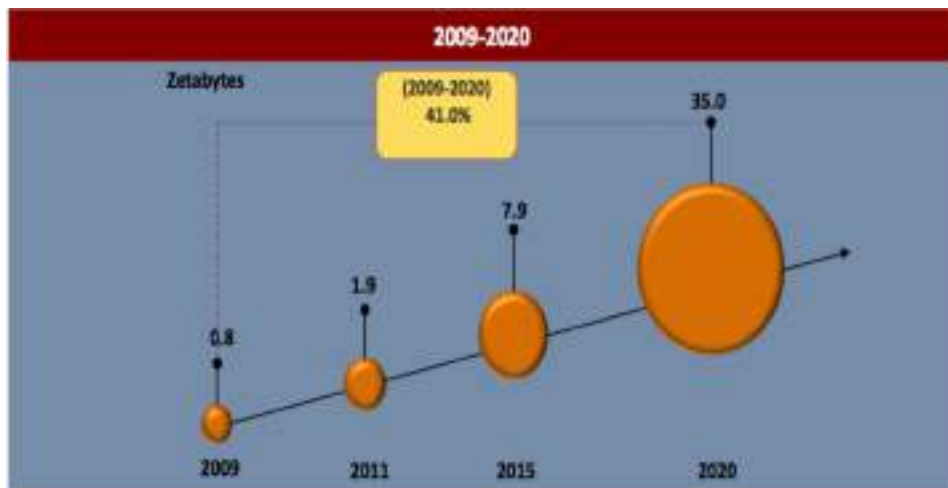
Necesidades para Big Data: Números de crecimiento



La tasa de crecimiento anual prevista es del 41%



El 80% de los datos son desestructurados



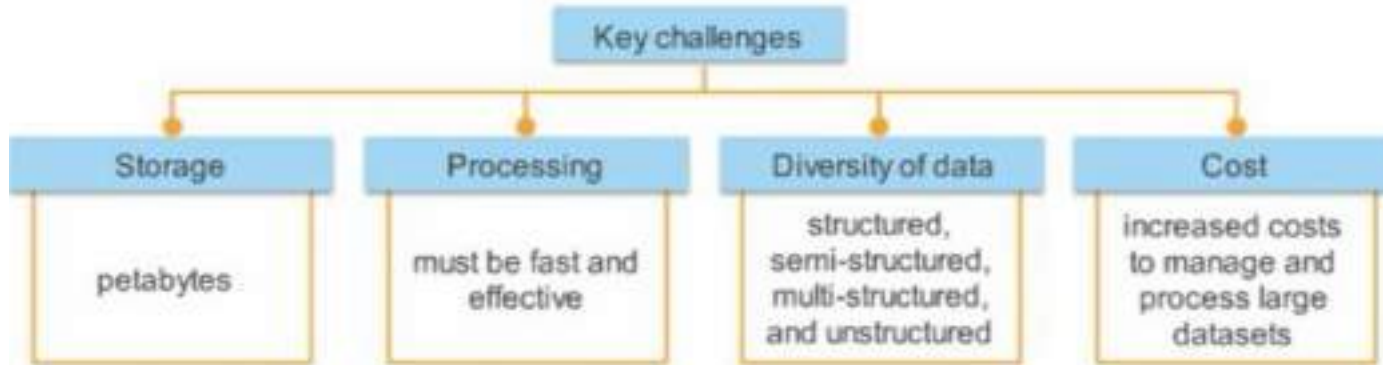
Necesidades para BigData: Retos

Almacenar, procesar de forma efectiva gran cantidad y variedad de datos controlando el coste y de manera escalable para alcanzar conocimiento.

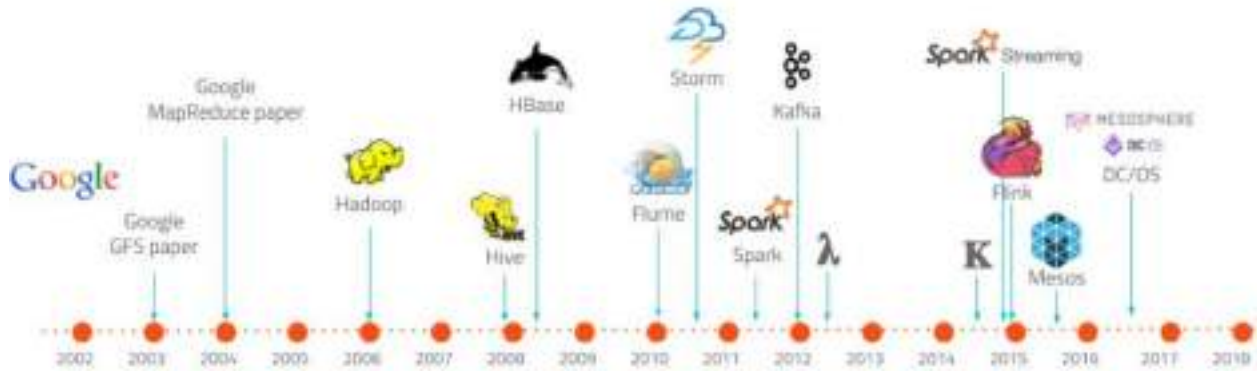


"More infrastructure for more data" is an old concept

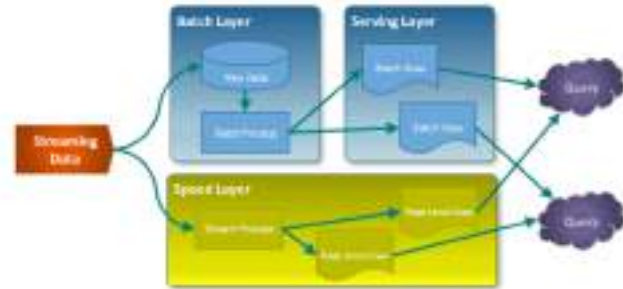
Using legacy techniques, companies cannot efficiently scale up systems as their data grows



Arquitectura de los sistemas para BigData



Línea temporal de las tecnologías para Big data



Sistema robusto tolerante a fallos, que sea linealmente escalable y que permita realizar escrituras y lecturas con baja latencia => **Arquitectura lambda**

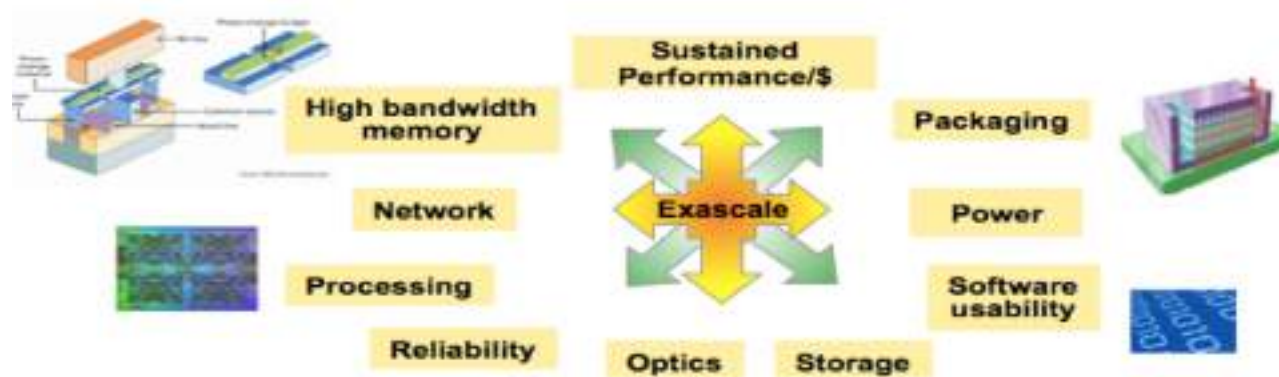
Necesidades actuales de computación

- **Ámbito de aplicabilidad de los sistemas más exigente:**
 - ▣ Big Data and High Performance Analytics
 - ▣ Data-centric Computing
- **¿ Son necesarias nuevas arquitecturas?**
 - ▣ Realizar la computación mas cerca de los datos.



Tendencias en computación de altas prestaciones

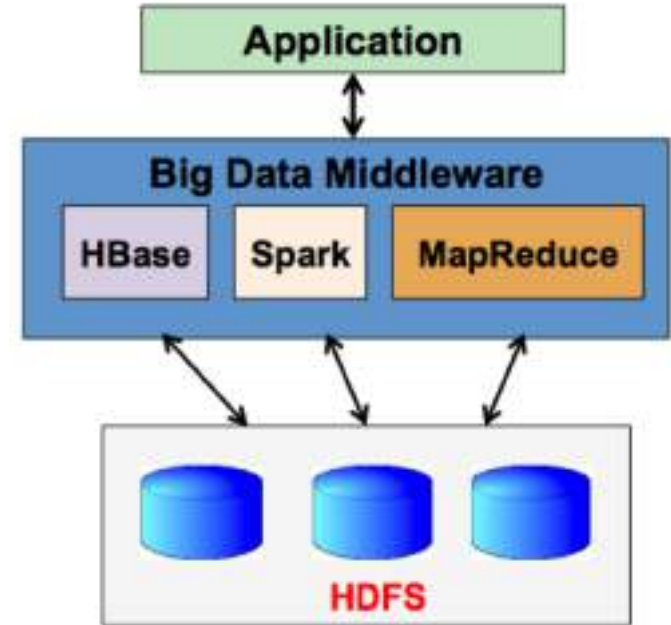
- Evolución de los procesadores para **la era Exascale**
 - Evitar dependencia de factores tecnológicos
 - Arquitecturas caracterizadas por llevar la computación mas cerca de los datos
 - Nuevas tecnologías de memoria y empaquetamiento.



- Nodos con arquitectura heterogénea interconectados con redes.
- Explotar el paralelismo: ILP, TLP, Paralelismo de datos, Cluster, Grid Computing, Cloud Computing
- Eficiencia en coste y consumo de potencia.

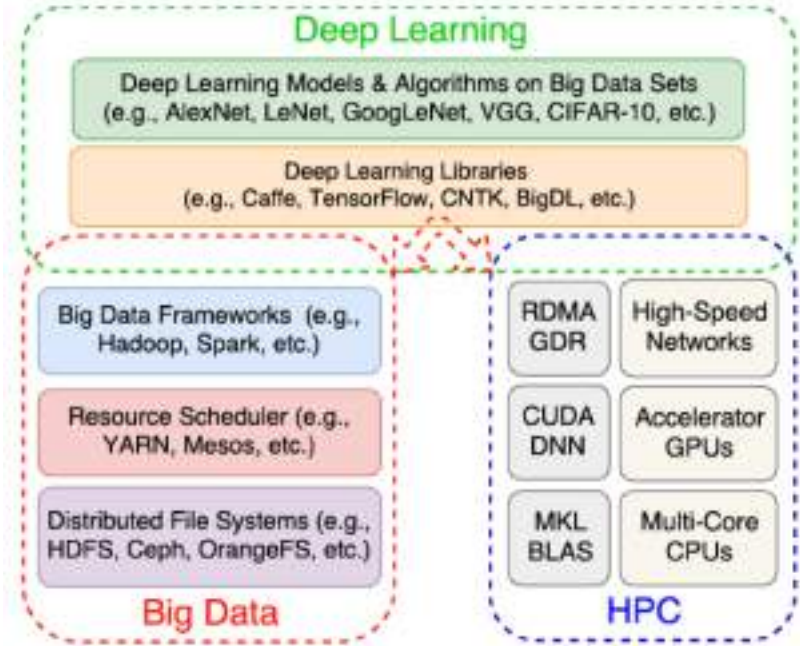
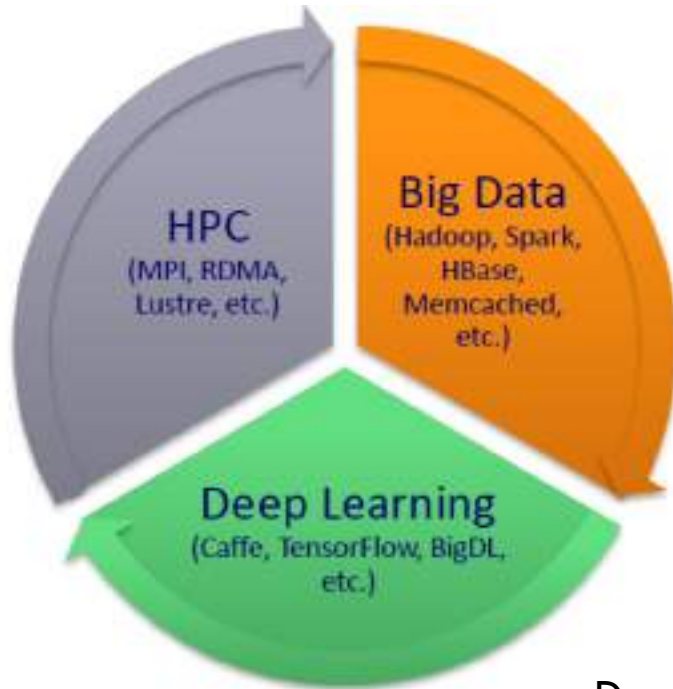
Entornos (Frameworks) para procesamiento de grandes volúmenes de información

- *Frameworks* para Big Data: **Hadoop** **MapReduce** y **Spark** son actualmente los entornos de ejecución más populares
- Hadoop Distributed File System (HDFS) es el sistema de ficheros que está por debajo de Hadoop, Spark, y la base de datos Hbase (Hadoop database)
- Hoy en día, se utilizan a nivel de explotación en organizaciones como: Facebook, Yahoo!,...



Sistemas para BigData, HPC y Deep Learning

Influencias entre High Performance Computing(HPC), Big Data, y Deep Learning (DL)



Deep Learning (DL) es un subconjunto de Machine Learning (ML), que está revolucionando los entornos de Big Data

Infraestructura de los sistemas para BigData:

Cluster de ordenadores:



Solución *low-cost*

Opciones de integración de arquitecturas Big Data

Arquitecturas físicas

Plataformas Cloud

Almacenamiento datos en la nube

Ahorro en el hardware

Mantenimiento por parte del proveedor

Menor control del sistema desarrollado

Servidor especializado:



Ejemplo:

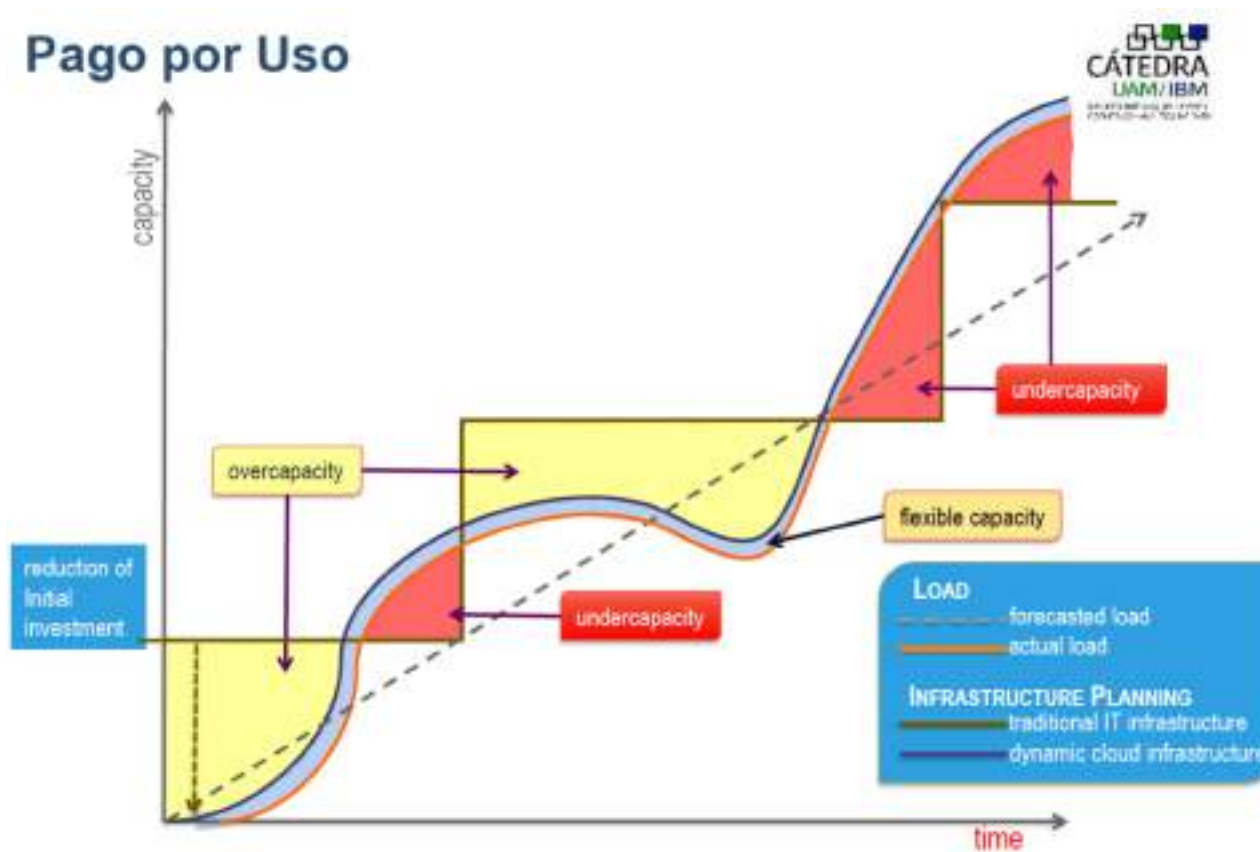


Coste fijo + coste variable

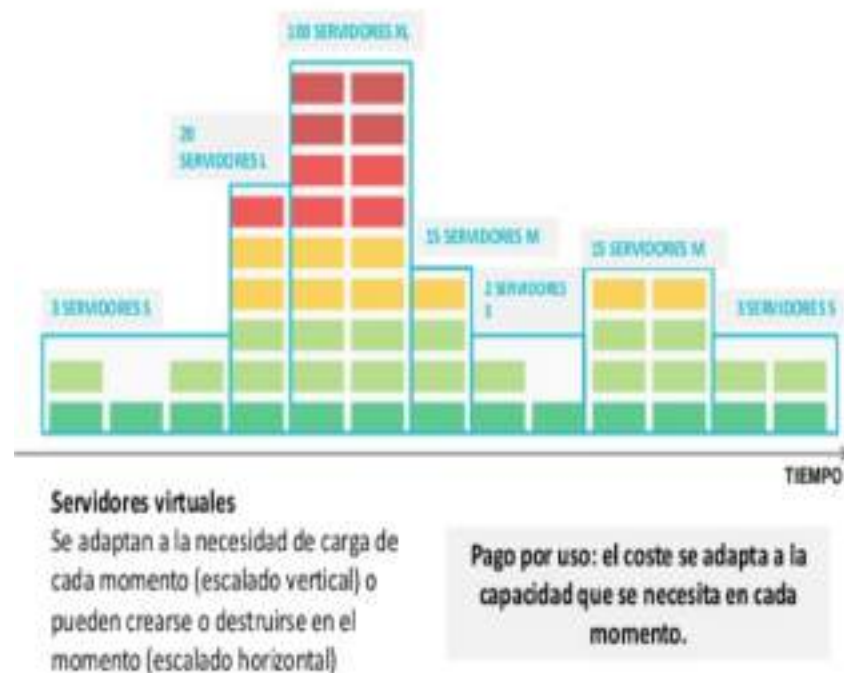
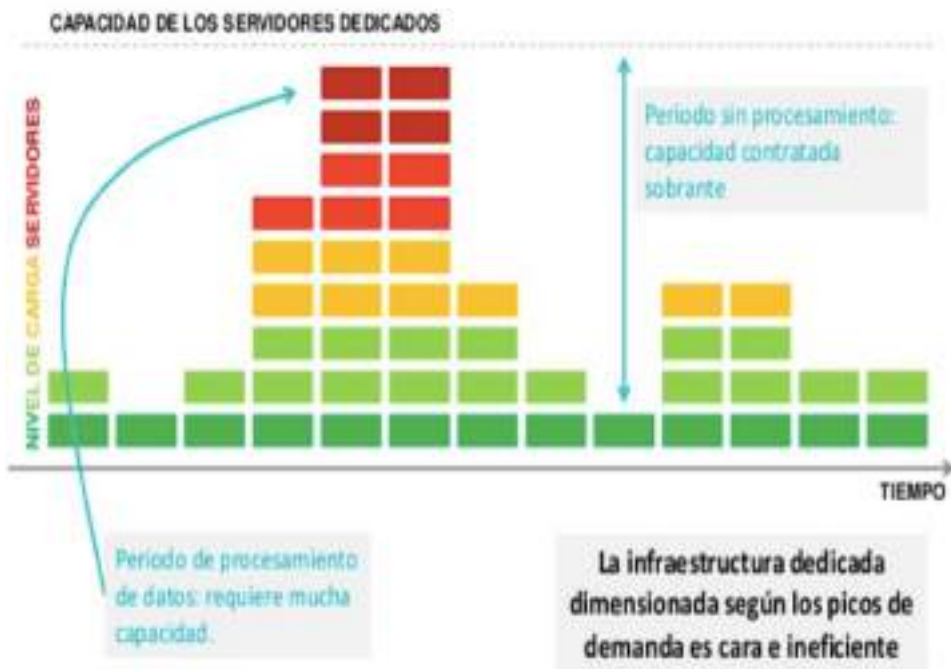
Soluciones adaptables

Escalabilidad automática

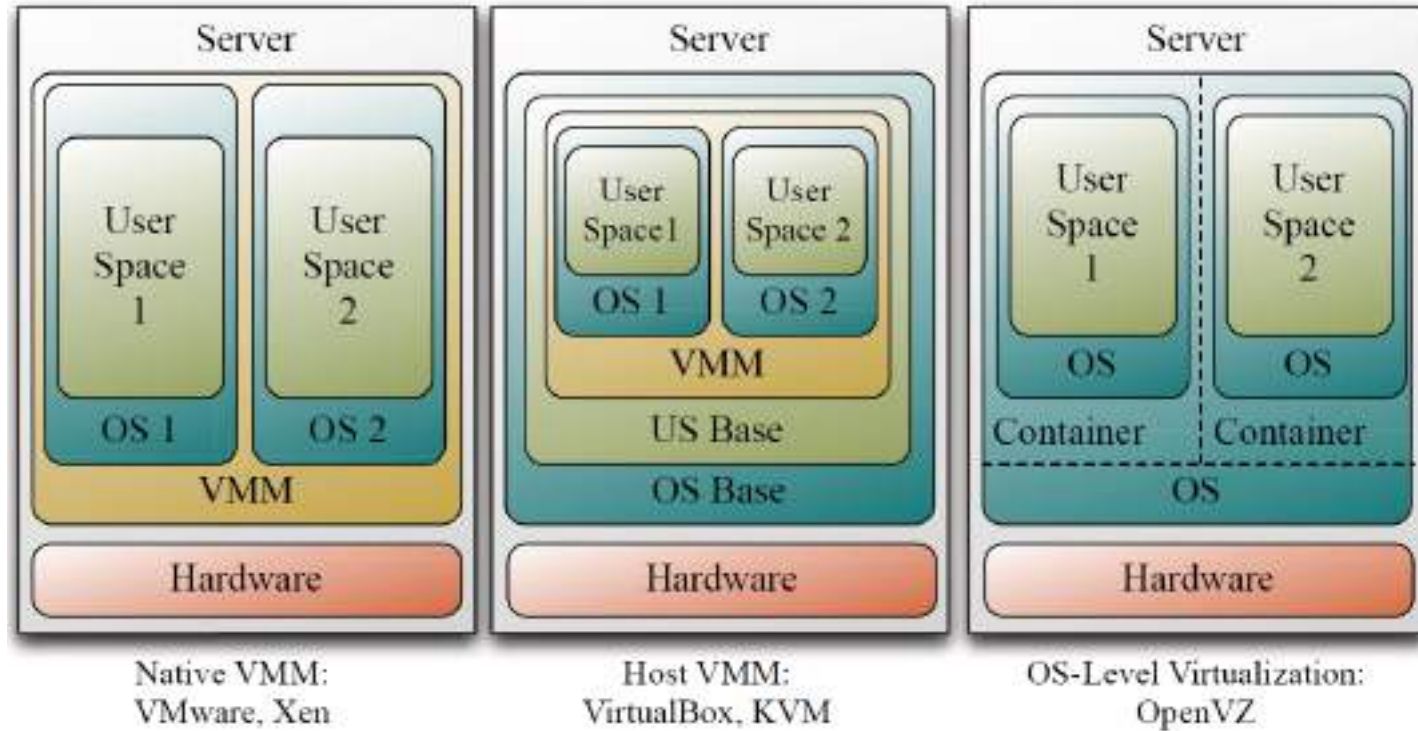
Sistemas para BigData: Flexibilidad de la infraestructura



Sistemas para BigData: Flexibilidad de la infraestructura



Virtualización de plataformas



Performance Comparison of Hardware Virtualization Platforms. Daniel Schlosser et. al. 2011

Sistemas para BigData: Físico vs Virtual

Servidores Físicos

- Servidor físico dedicado (Intel x86) para un cliente, que no es accedido ni compartido por otros.
- Se puede usar con un hipervisor, un sistema operativo, un appliance virtual, o con una imagen cualquiera subida por el consumidor.
- Discos internos de diferentes tipos, y diferentes RAID.
- Desplegado entre 30 minutos y 4 horas.



Servidores Virtuales

- Pueden ser en entorno compartido o dedicado, según si comparten recursos con servidores de otros cliente.
- Desplegado en segundos.
- Responsabilidad sobre el Sistema Operativo.
- Facilidad de gestión.



Máquinas virtuales

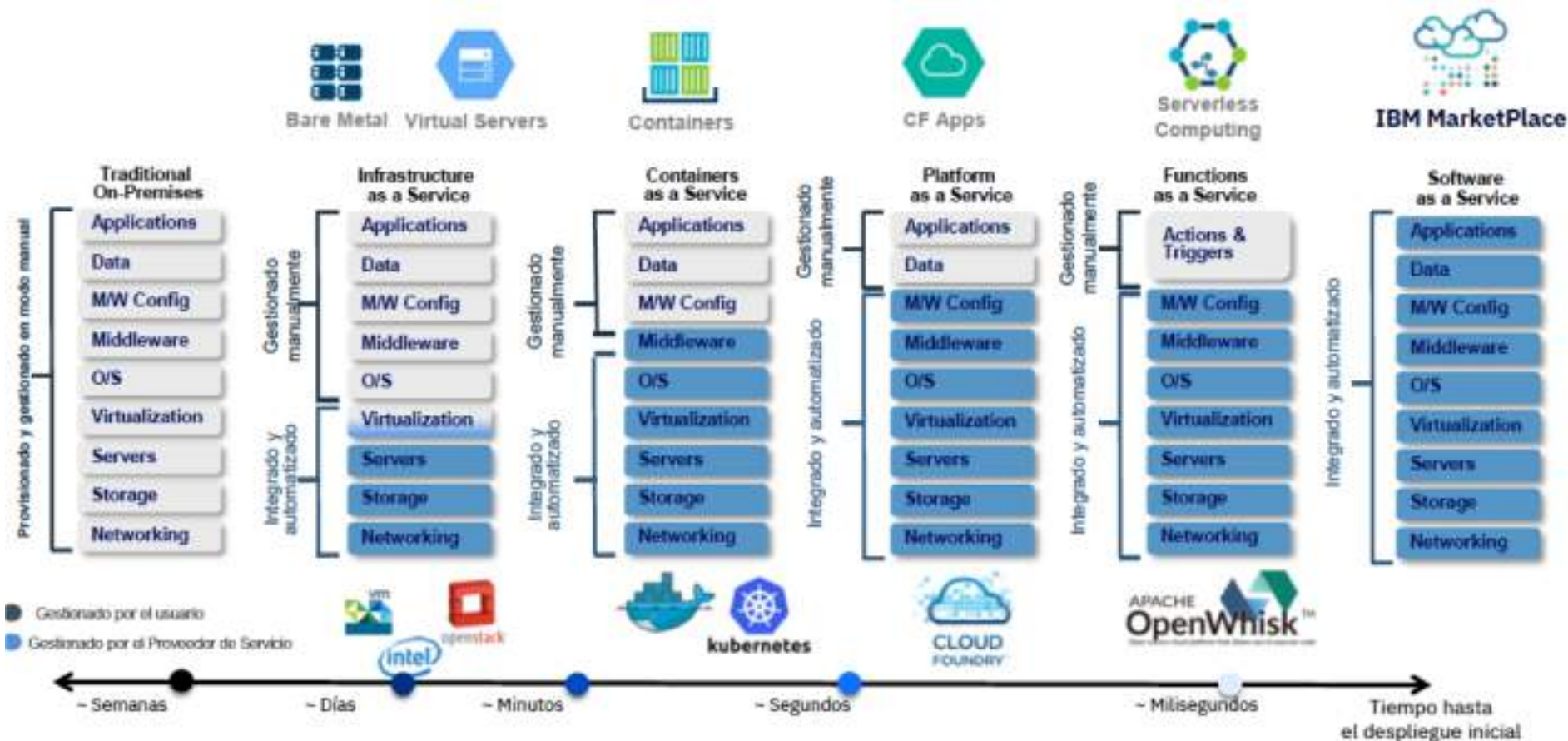


Contenedores Docker

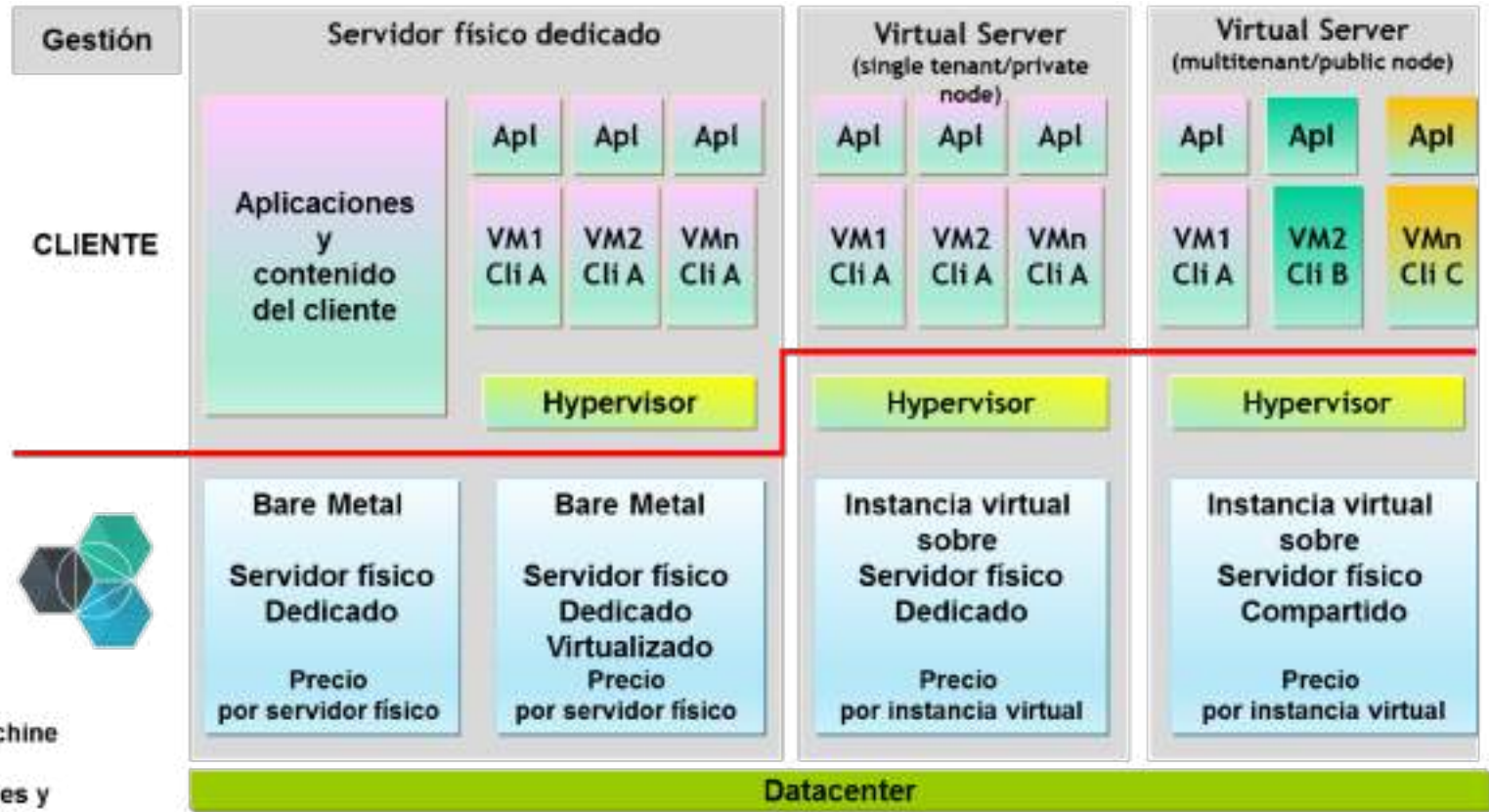
Ventajas de los contenedores:

- Mayor rapidez de despliegue, arranque y escalado
- Menor consumo de recursos
- Máxima portabilidad entre entornos y sistemas

Sistemas para BigData: On Premise vs Cloud



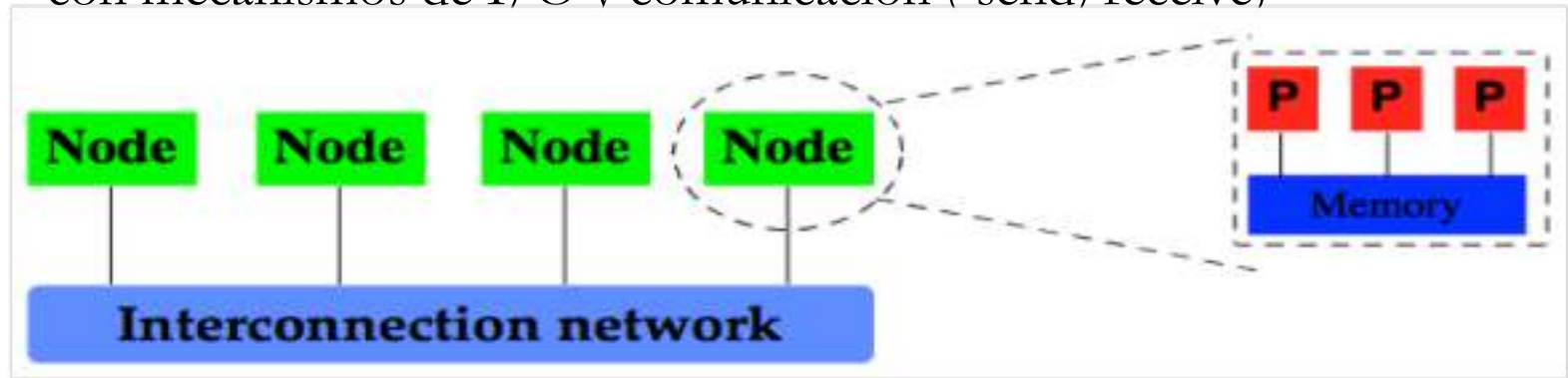
Sistemas para BigData: Servidores en “Bare Metal” o Virtuales



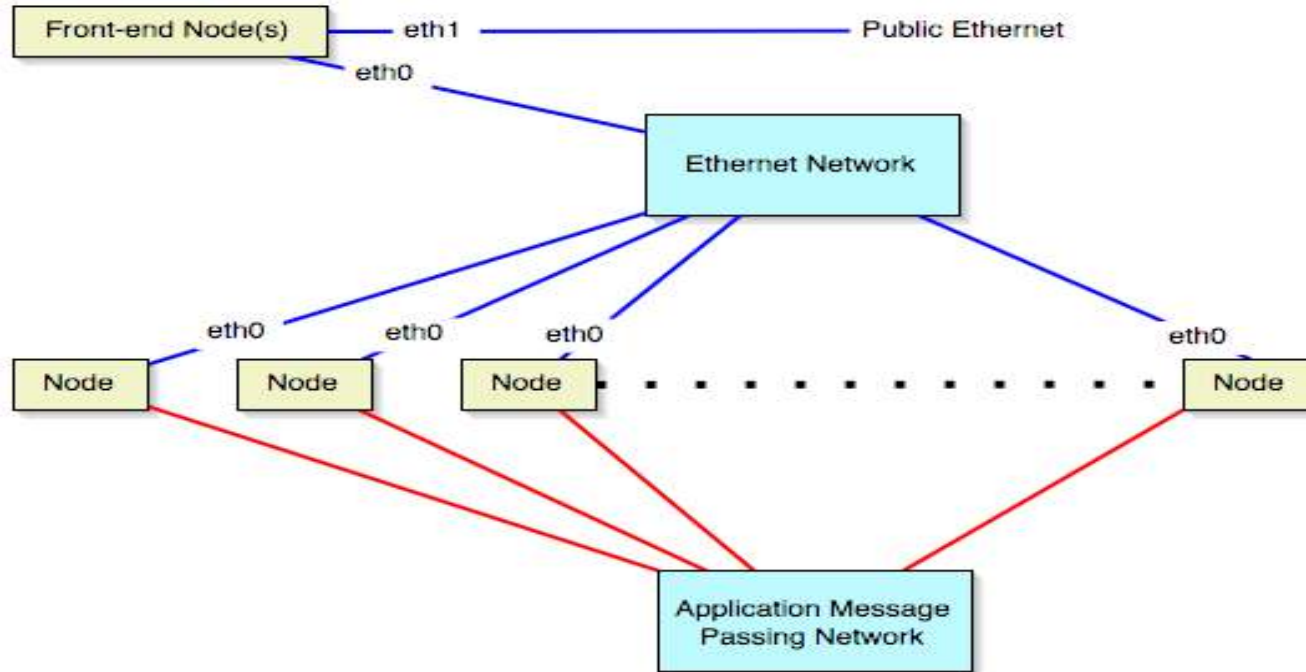
VM: Virtual Machine
Cli: Cliente
Apl: Aplicaciones y contenido del cliente

Multicomputador: Arquitectura de referencia

- Multicomputador = Nodos + Red de Interconexión.
- Nodo = procesador(es) + memoria local
 - El acceso a memoria local es rápido, porque no involucra conexión de red (acceso a memoria convencional en sistema uniprocador)
 - El acceso a memoria remota es lento, involucra conexión de red, con mecanismos de I/O y comunicación (send/receive)

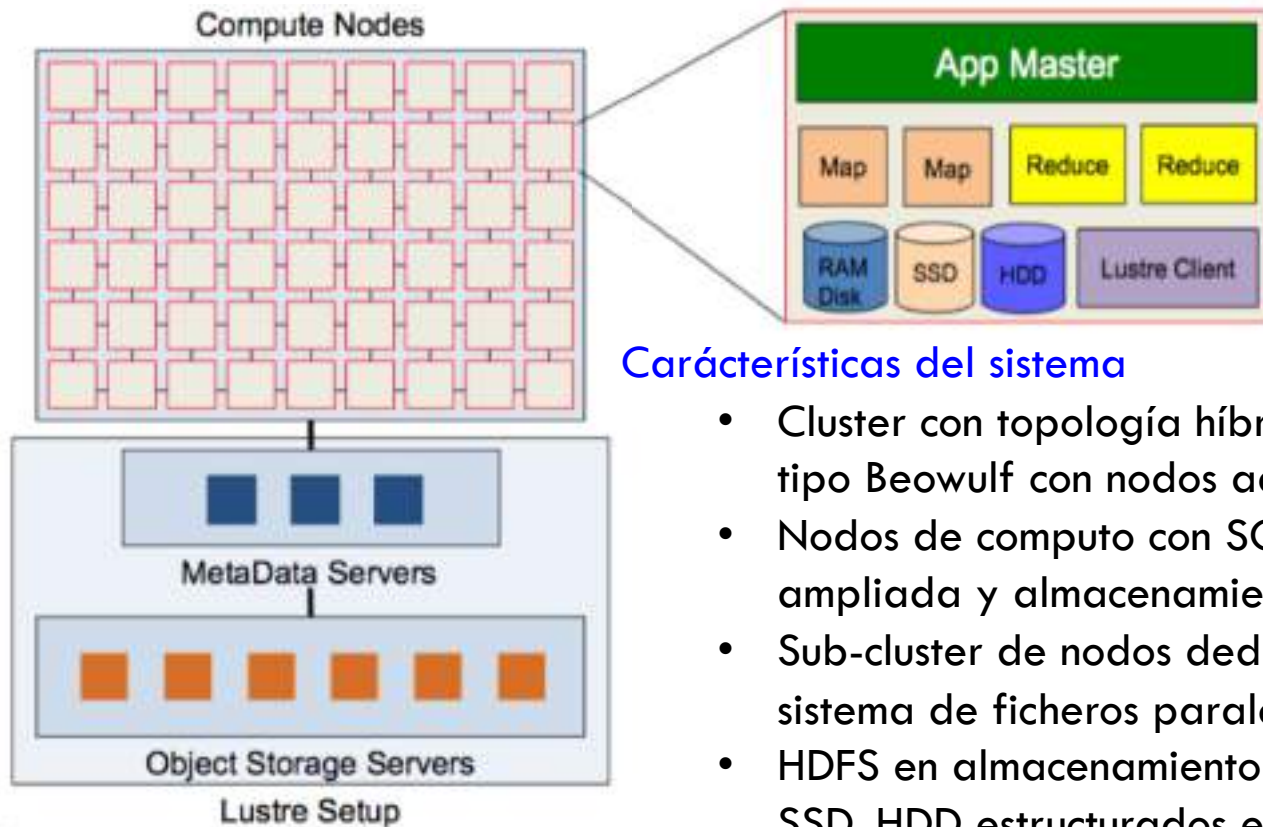


Arquitectura de un cluster para HPC (High Performance Cluster)



- Muchos nodos de computación conectados con red de alto rendimiento

Sistema para Bigdata: Arquitectura de referencia

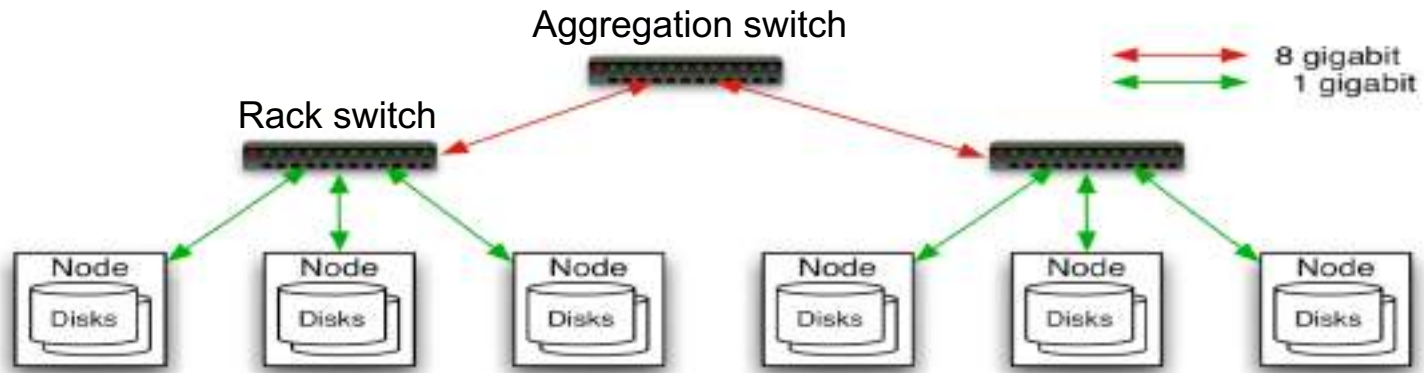
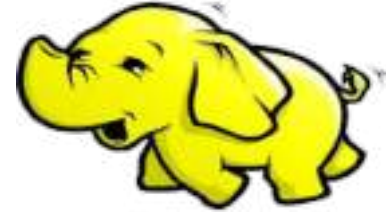


Características del sistema

- Cluster con topología híbrida de una arquitectura tipo Beowulf con nodos adicionales para I/O.
- Nodos de computo con SO versión ligera; memoria ampliada y almacenamiento local pequeño.
- Sub-cluster de nodos dedicados para I/O con un sistema de ficheros paralelo, (en la figura Luster)
- HDFS en almacenamiento heterogéneo: RAMDisk, SSD, HDD estructurados en RAID, JBOD,...

Cluster Hadoop

- Cluster creado con commodity Hardware;
 - Nodos inicialmente eran PCs
 - 30-40 nodos/rack
 - Red a 1 gigabit/s en rack



HDFS : Hadoop Distributed File System



- Sistema de Ficheros distribuido muy grande
 - 10K nodos, 100 millones de ficheros 10PB
- Realizado con “*Commodity Hardware*”
 - Ficheros replicados para tolerancia a fallos
 - Detecta fallos y recupera los datos.
- Optimizado para proceso por lotes (“*Batch Processing*”).
 - Expone la localización de los datos y así permite que la computación se pueda llevar cerca de los datos.
 - El ancho de banda agregado es muy alto.

Infraestructura: Características de los componentes base

- Sistema multiprocesador/multicore con memoria compartida NUMA.
- Componentes de un cluster:
 - Procesador: Multi-core/many-core con Hyperthreading.
 - Almacenamiento:
 - Memoria (DDR4 , Flash, 3D Xpoint)
 - HDDs, Solid State Disks (SSDs),
 - Non-Volatile Random-Access Memory(NVRAM), y NVMe SSD.
 - Red de Interconexión con RDMA (Remote DirectMemoryAccess) networking
 - InfiniBand y RoCE (RDMA over Converged Enhanced Ethernet)
 - Aceleradores
 - NVIDIA GPGPU,
 - IntelXeon Phi,
 - FPGA



Multi-core Processors



Accelerators / Coprocessors
high compute density, high
performance/watt
>1 TFlop DP on a chip



High Performance Interconnects -
InfiniBand
:1usec latency, 100Gbps Bandwidth



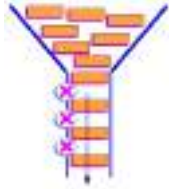
SSD, NVMe-SSD, NVRAM

Arquitecturas para BigData: El procesador

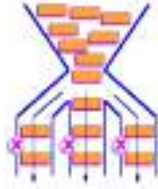
Paralelismo en la ejecución de instrucciones:



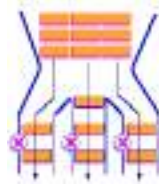
Escalar



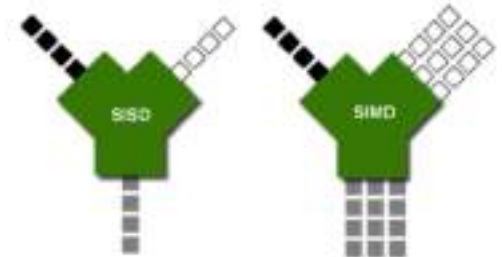
Segmentado



Superescalar



VLIW

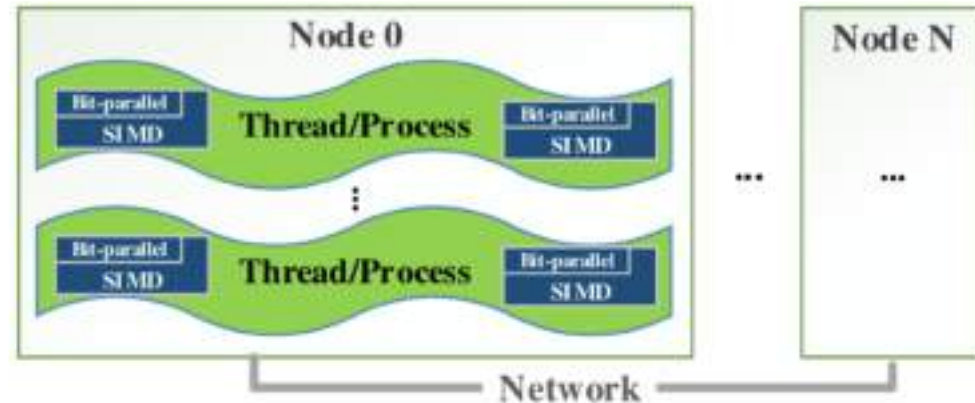


SIMD (vectorial)

■ Instructions
□ Data
■ Results

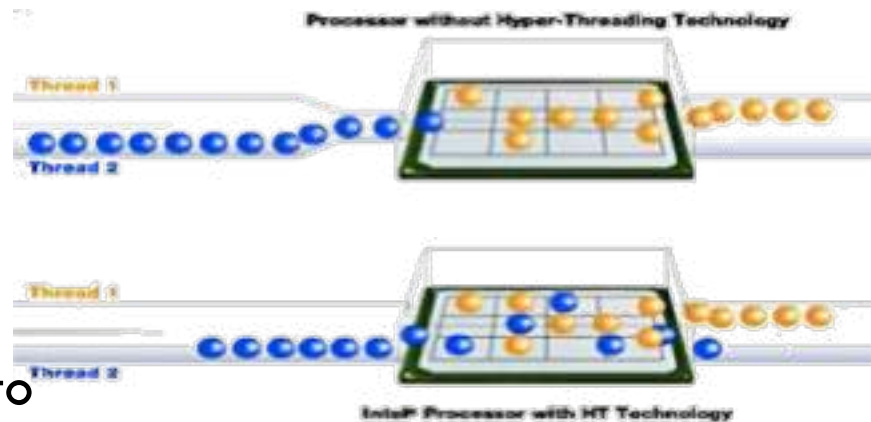
- Procesador con tecnología Multi-core/many-core con tres niveles de paralelismo:

- Nodo/core
- Muti-Thread (SMT, HT)
- Instrucciones SIMD



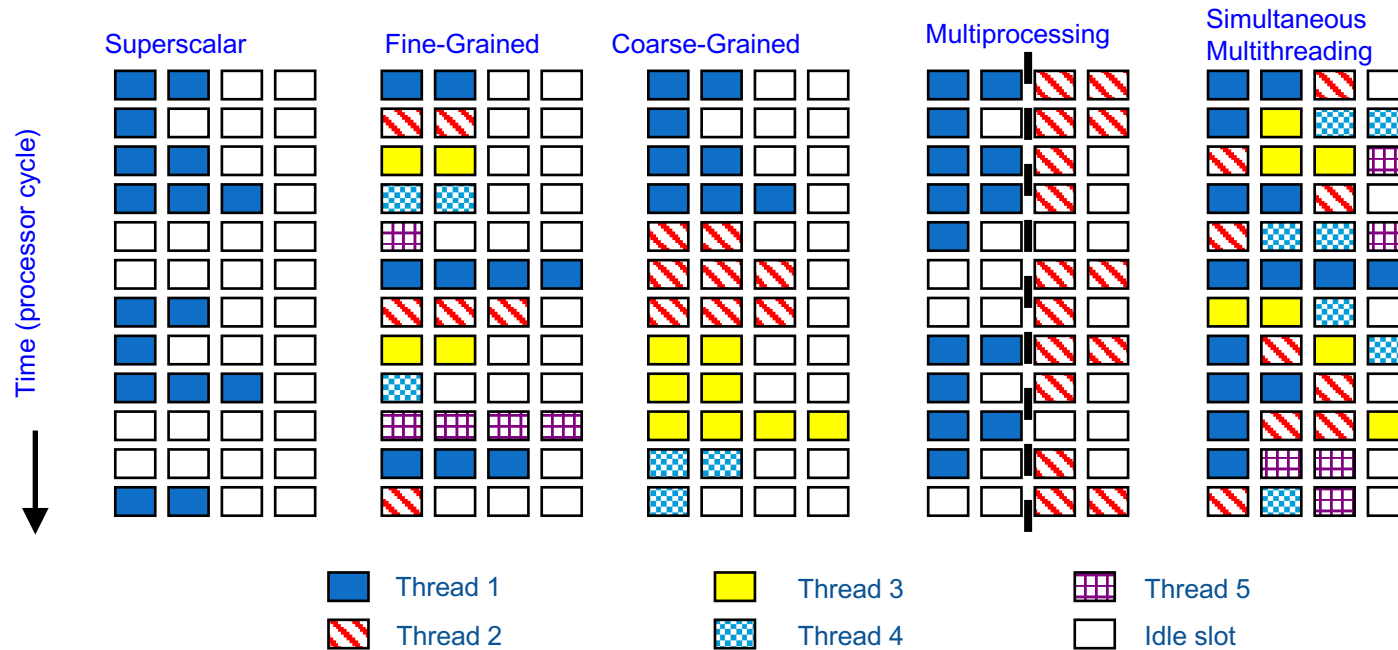
Arquitecturas para BigData: procesador con hyperthreading

- **Cada procesador maneja dos *threads***
 - Cuando el que está ejecución se bloquea, entra el otro
 - Estructura HW para hacer un cambio de contexto del procesador (registros, ...)
- **Número de procesadores x 2**
 - ¡No es real!
 - Útil en sistemas de sobremesa
 - Muchos bloqueos
 - No siempre útil en alto rendimiento
 - N° bloqueos mínimo



Procesador: Hyperthreading (HT) o Multi-Thread simultaneo (SMT)

- Procesador superescalar con tecnología Multi-Thread simultaneo (SMT)



Arquitecturas para BigData: Optimizando el procesador

Accelerating Apache Spark machine learning with Clear Linux* OS ...

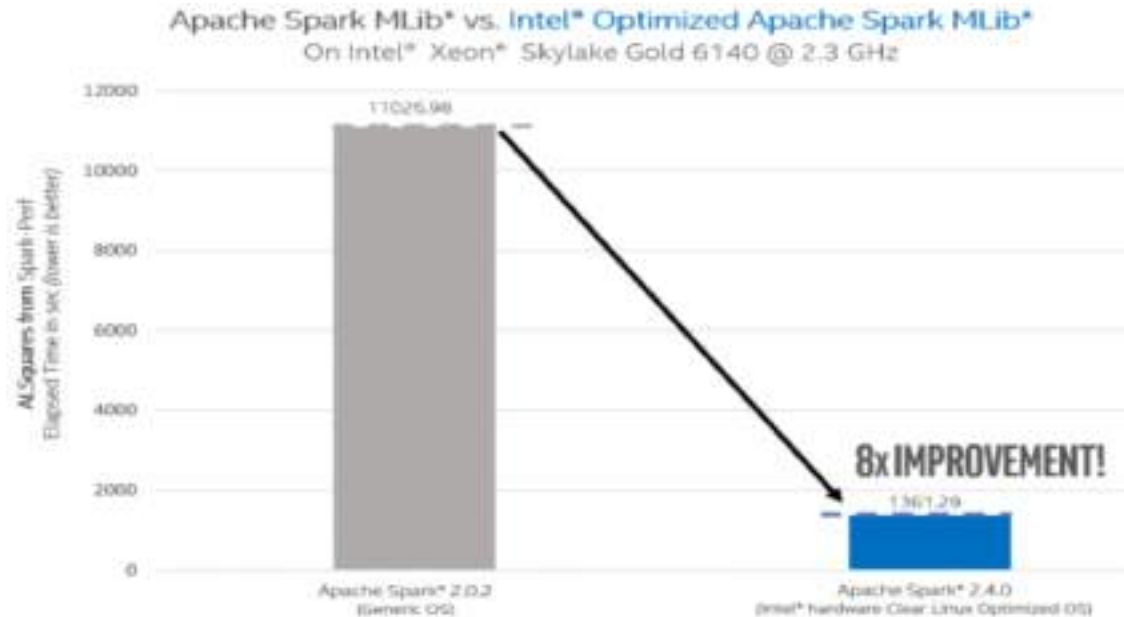
<https://01.org/blogs/2018/apache-spark-clear-linux>

Características:

- Intel® Advanced Vector Extensions 512 (Intel® AVX-512)
- Intel® Memory Protection Extensions (Intel® MPX)
- Intel® Ultra Path Interconnect (Intel® UPI)

Math LIB:

Intel MKL 2018.3.222 vs F2JBLAS



Hyper-Threading (HT) technology was disabled to achieve better performance !

Arquitecturas para BigData: El procesador optimizado

[1] Architectural Impact on Performance of In-memory Data Analytics: Apache Spark Case Study

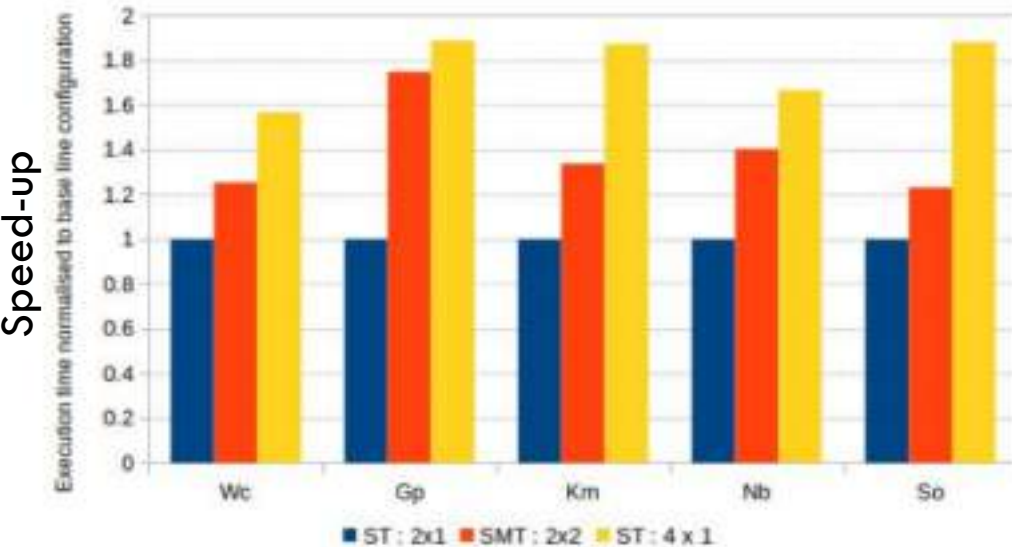
TABLE VII: Machine and Spark Configurations to evaluate Hyper Threading

		ST:2x1	SMT:2x2	ST:4x1
Hardware	No of sockets	1	1	1
	No of memory nodes	1	1	1
	No. of cores	2	2	4
	No. of threads	1	2	1
Spark	spark.driver.cores	2	4	4
	spark.default.parallelism	2	4	4
	spark.driver.memory (GB)	24	24	24

TABLE III: Machine Details.

Component	Details	
Processor	Intel Xeon E5-2697 V2, Ivy Bridge micro-architecture	
	Cores	12 @ 2.7GHz (Turbo up 3.5GHz)
	Threads	2 per Core (when Hyper-Threading is enabled)
	Sockets	2
	L1 Cache	32 KB for Instruction and 32 KB for Data per Core
	L2 Cache	256 KB per core
Memory	L3 Cache (LLC)	30MB per Socket
	2 x 32GB, 4 DDR3 channels, Max BW 60GB/s per Socket	
	OS	
JVM	Linux Kernel Version 2.6.32	
Spark	Oracle Hotspot JDK 7u71	
	Version 1.5.0	

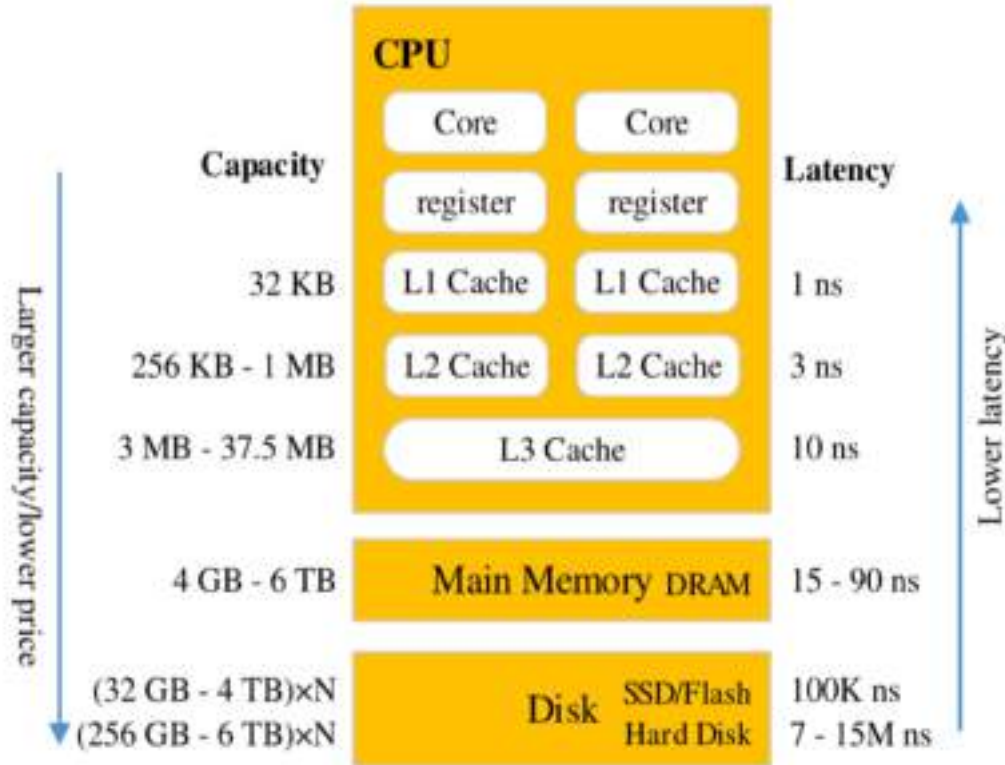
Arquitecturas para BigData: El procesador optimizado



(a) Multi-core vs Hyper-Threading

- Word Count (Wc): counts the number of occurrence of each word in a text file
- Grep (Gp): searches for the keyword The in a text file and filters out the lines with matching strings to the output file
- K-Means (Km): uses K-Means clustering algorithm from Spark MLlib. The benchmark is run for 4 iterations with 8 desired clusters
- NaiveBayes (Nb): runs sentiment classification
- Sort (So): ranks records by their key

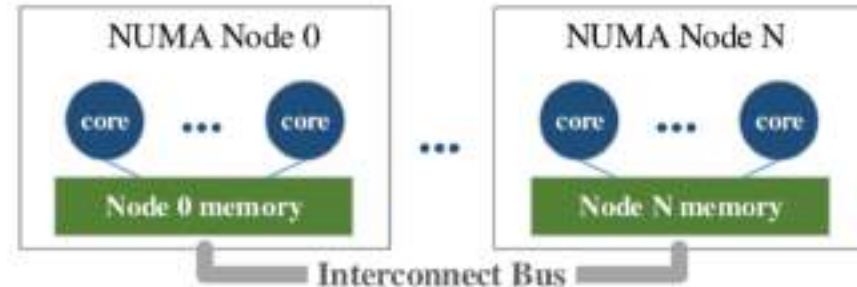
Arquitecturas para BigData: Sistema de memoria



➤ Sistema de Jerarquía de Memoria de un Nodo:

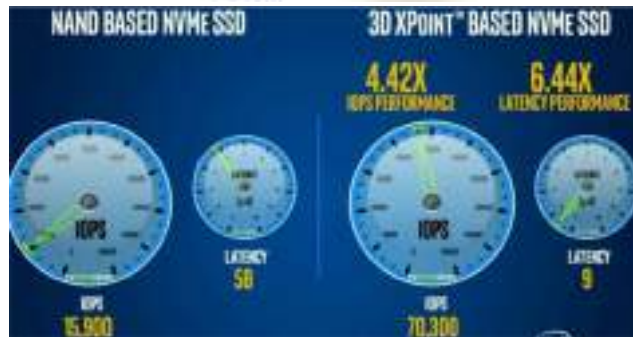
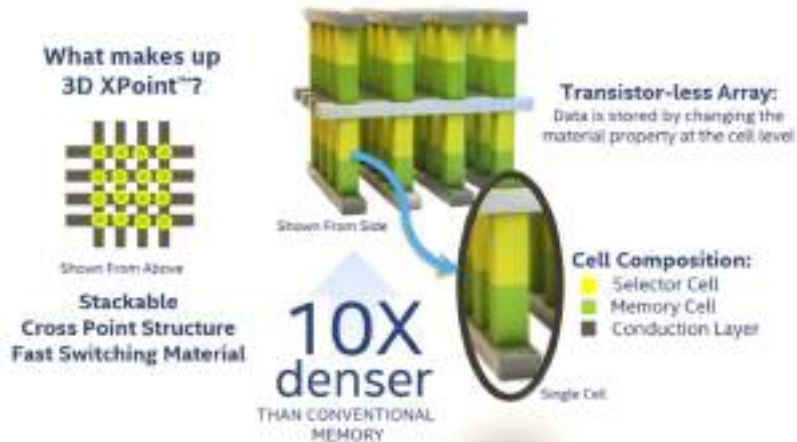
- Registro interno del procesador
- Cache multinivel L1, L2 y L3
- DRAM
- Almacenamiento externo

➤ Sistema Multiprocesador con acceso a memoria compartida NO UNIFORME (NUMA)

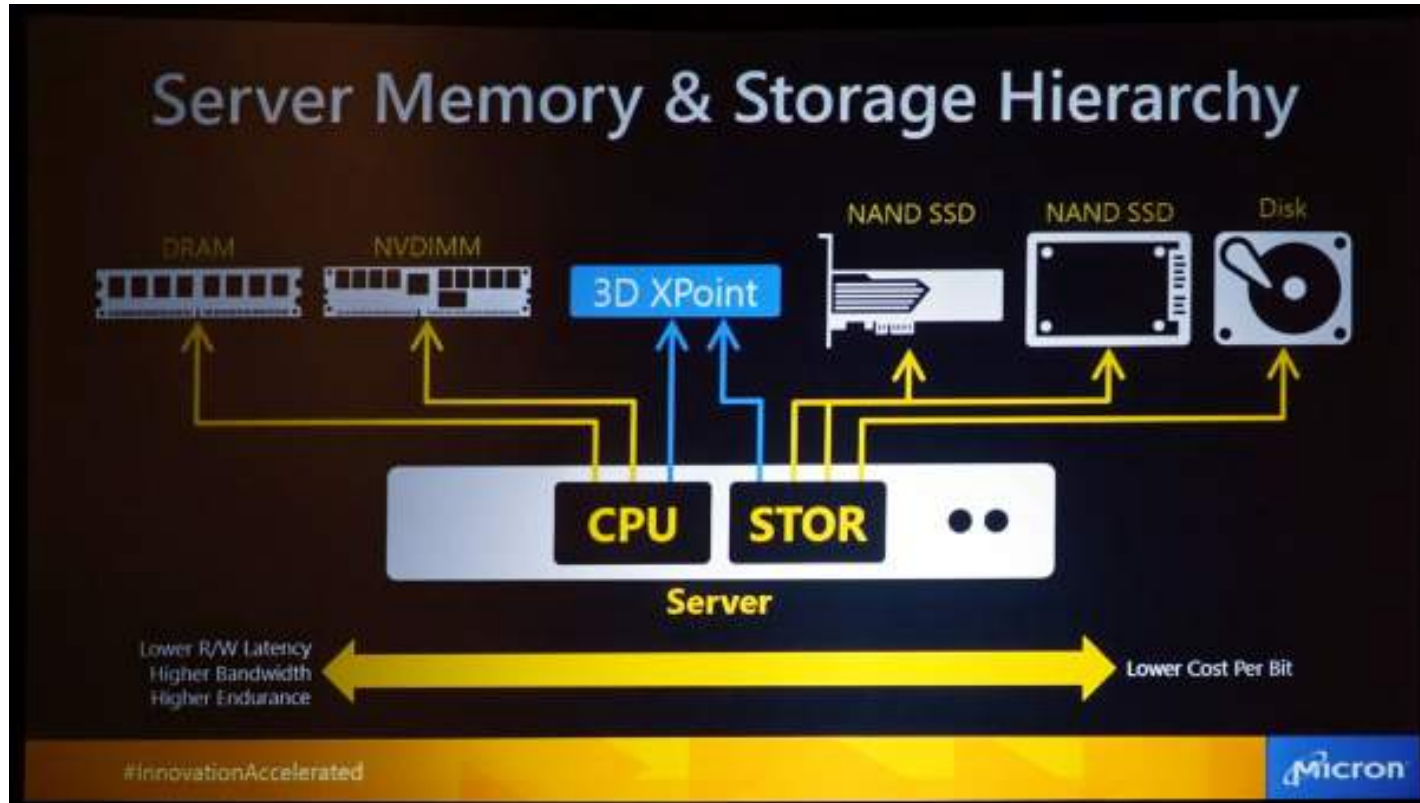


Arquitecturas para BigData: Memoria

Memoria: DDR(RAM) a 3D Xpoint (no volátil)



Convergencia de memoria y almacenamiento



Tamaños y latencias en memoria/almacenamiento

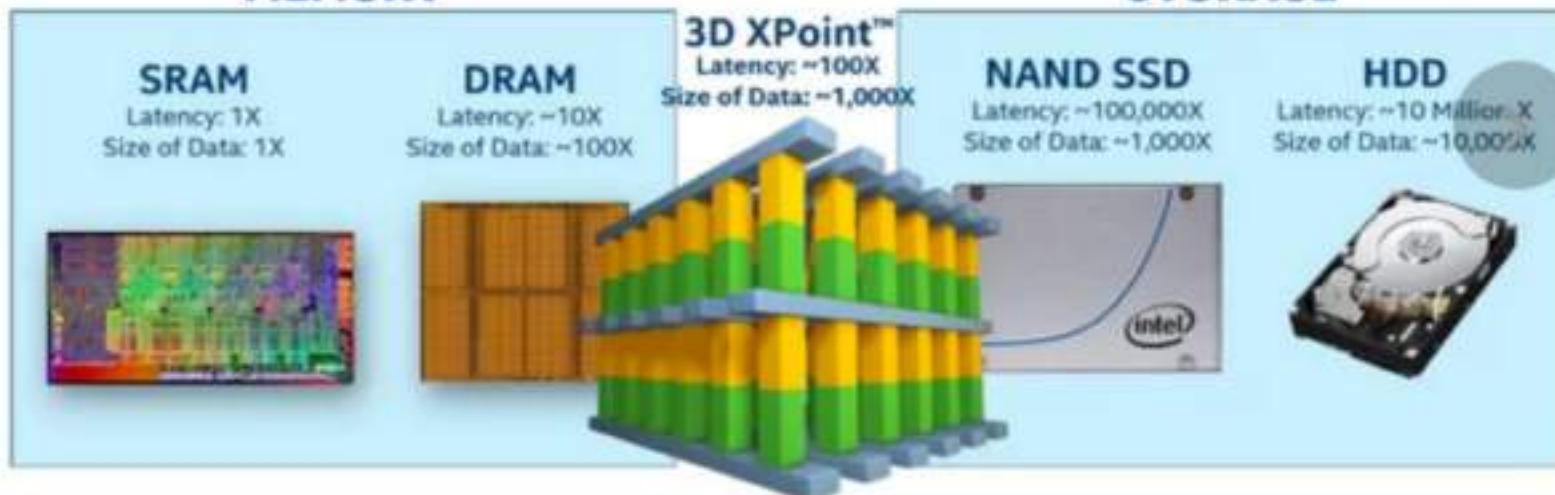
3D XPOINT™ MEMORY MEDIA

Breaks the memory/storage barrier

MEMORY

+

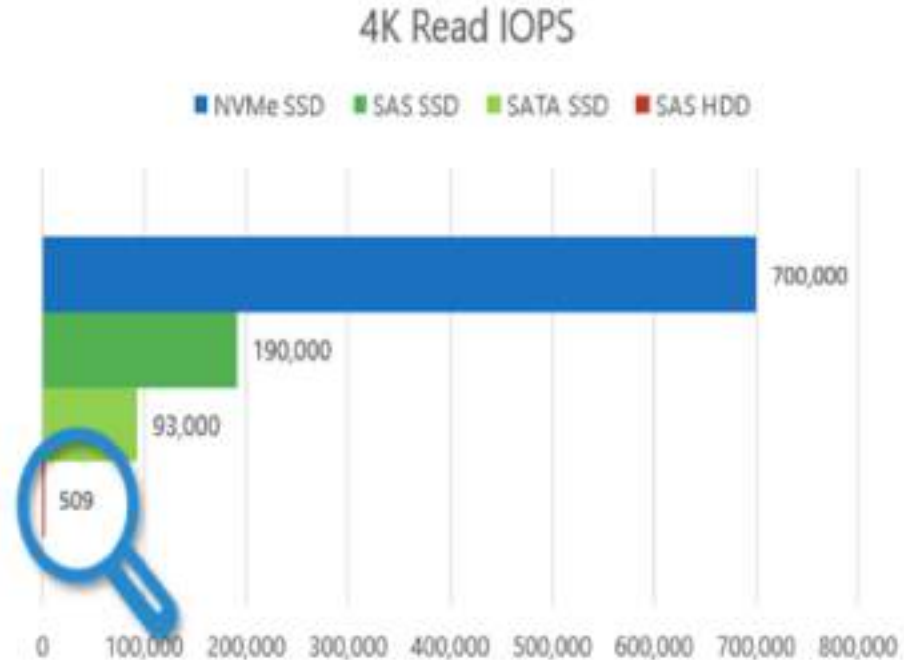
STORAGE



Almacenamiento NVMe Express (NVMe SSD)

- NVMe es la especificación de la interfaz del dispositivo a nivel lógico (*logical device interface*) para acceder a medios de almacenamiento no volátiles (*non-volatile storage media*) conectados al bus PCI Express (PCIe) o por conexión M.2.

NVMe está optimizado para las operaciones de unidades de almacenamiento con latencias muy bajas y que son comúnmente ofrecidos por SSD con memoria no volátil.



Infraestructura BigData: Posibilidades de almacenamiento



Disco Local

SSD, SAS, SATA

Rendimiento ajustable

Config RAID

Efímero



Servidores de Almacenamiento

Dedicado y gestionado por cliente

100% Personalizable

Acceso Concurrente



Almacenamiento de ficheros

NFS

Fácil gestión y administración

Resiliente

Rendimiento alto



Almacenamiento de Bloques

iSCSI

Replicación entre CPDs

Resiliente

Rendimiento muy alto



Almacenamiento de Objetos

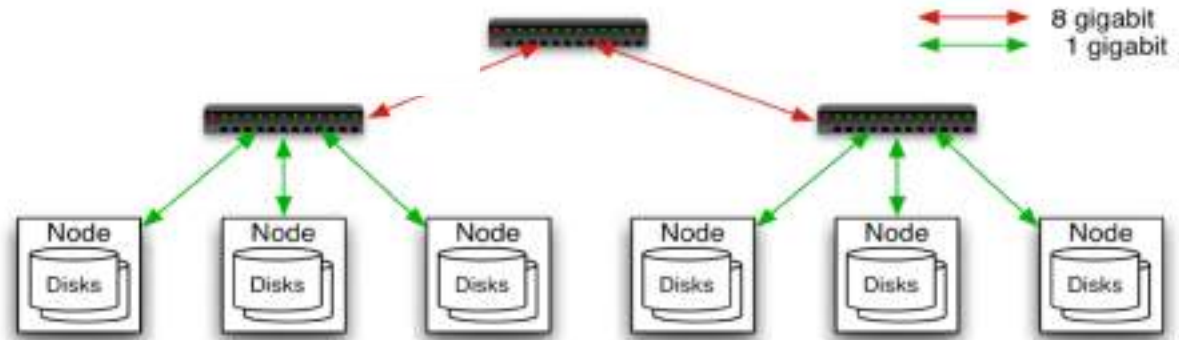
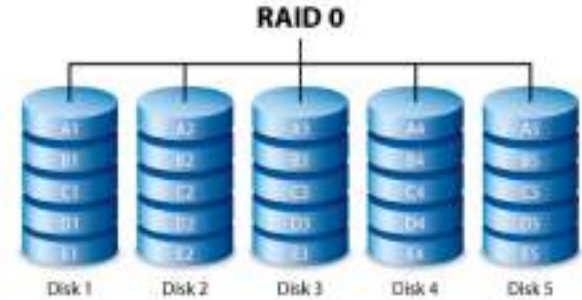
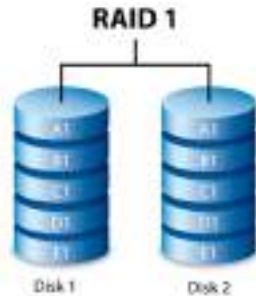
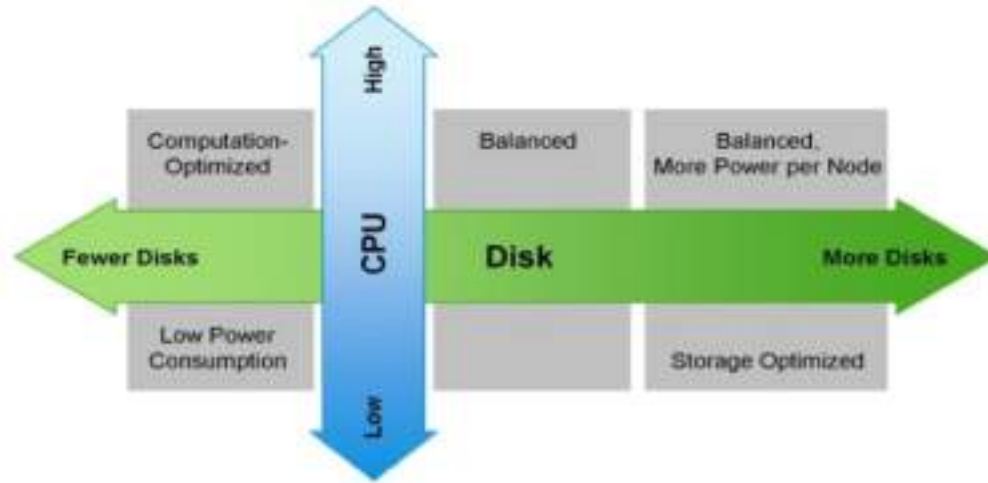
Datos No estructurados

PAYG

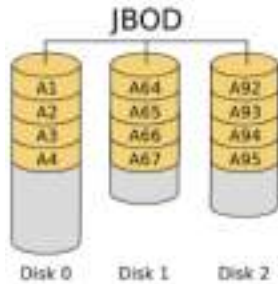
Escalable

Integración con CDN

Infraestructura para BigData: Requisitos del sistema



Infraestructura para BigData: Almacenamiento en RAID



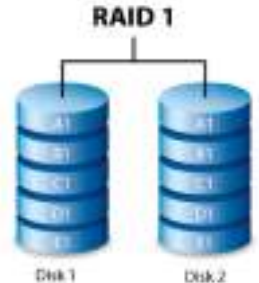
JBOD: Just a Bunch of Disks (un puñado de discos), este tipo de RAID configura los discos para que cada uno funcione de manera independiente como si se trataran de discos duros conectados de manera individual al ordenador.

RAID: (Redundant array of independent disks) sistema de almacenamiento de datos que utiliza múltiples unidades de disco (HDD/SSD) entre las cuales se distribuyen o replican los datos.

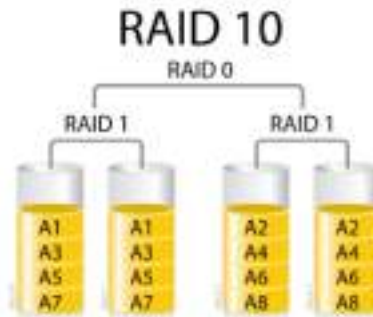
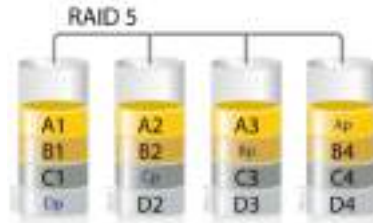


RAID 0. Todos los discos duros funcionan como un único volumen, y su espacio total es la suma del espacio de todos los discos duros. **Mayor ($\times N$) velocidad de lectura y escritura. No hay paridad de datos ni volumen de respaldo.**

RAID 1 Los datos se duplican en los discos duros como si fuese un espejo. **Velocidad de lectura $\times 2$. Sin mejora en la velocidad de escritura. Si falla un disco se puede reemplazar sin perder datos. Perdemos el 50% del espacio total de los discos.**



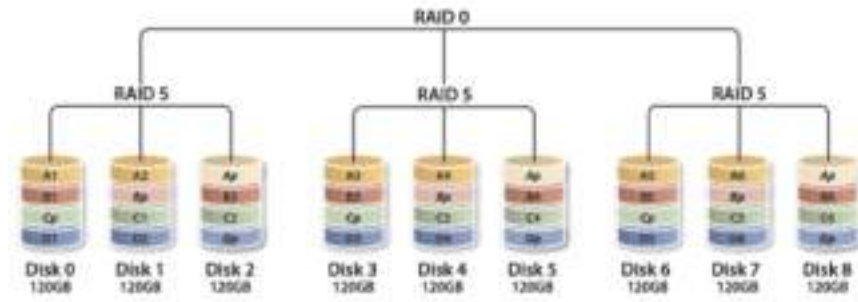
Infraestructura para BigData: Almacenamiento en RAID



RAID 5, los datos se distribuyen a lo largo de todos los discos duros. En una de las unidades se guarda la paridad. La paridad se reparte entre todos los discos duros.

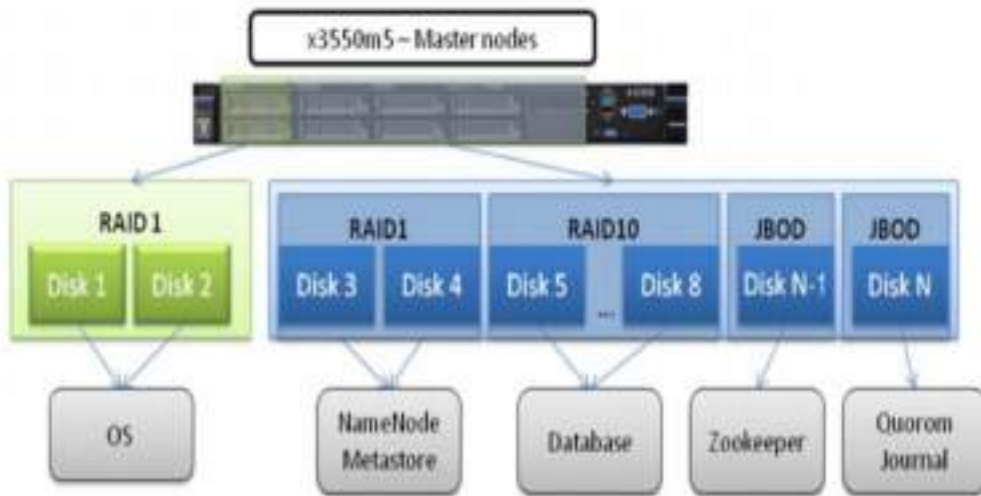
- El espacio total de los discos es $N-1$, igual que la mejora de la velocidad de lectura.
- No hay mejora en velocidad de escritura.
- Si falla uno de los discos duros, cualquiera de ellos, se puede reemplazar y recuperar todos los datos.
- Si fallan 2 no.

RAID 30/50/100



Infraestructura para BigData: Configuración de nodos

Nodo Maestro



Component	Master node configuration
System	System x3550 M5
Processor	2 x Intel Xeon processor E5-2650 v4 2.2 GHz 12-core
Memory - base	128 GB – 8 x 16 GB 2133 MHz RDIMM (minimum)
Disk (OS / local storage)	OS: 2x 2.5" HDD or SSD Data: 8 x 2TB 2.5" HDD
HDD controller	ServeRAID M5210 SAS/SATA Controller
Hardware management network adapter	Integrated 1GBaseT IMM interface
Data network adapter	Broadcom NetXtreme Dual Port 10GbE SFP+ Adapter

Infraestructura para BigData: Configuración de nodos

Nodo de datos

x3650m5 -- Datanodes



Component	Data node configuration
System	System x3650 M5
Processor	2 x Intel Xeon processor E5-2680 v4 2.4GHz 14-core
Memory - base	256GB: 8x 32GB 2400MHz RDIMM
Disk (OS)	2x 2.5" HDD or SSD
Disk (data)	4 TB drives: 14x 4TB NL SATA 3.5 inch (56 TB Total) 6TB drives: 14x 6TB NL SATA 3.5 inch (84 TB total) 8 TB drives: 12x 8TB NL SATA 3.5 inch (96 TB Total)
HDD controller	OS: ServeRAID M1215 SAS/SATA Controller HDFS: N2215 SAS/SATA HBA
Hardware storage protection	OS: RAID1 HDFS:None (JBOD). By default, Hortonworks maintains a total of three copies of data stored within the cluster. The copies are distributed across data servers and racks for fault recovery.

Infraestructura para BigData: Red de comunicación



Figure 7. Lenovo RackSwitch G8272

The enterprise-level Lenovo RackSwitch G8272 has the following characteristics:

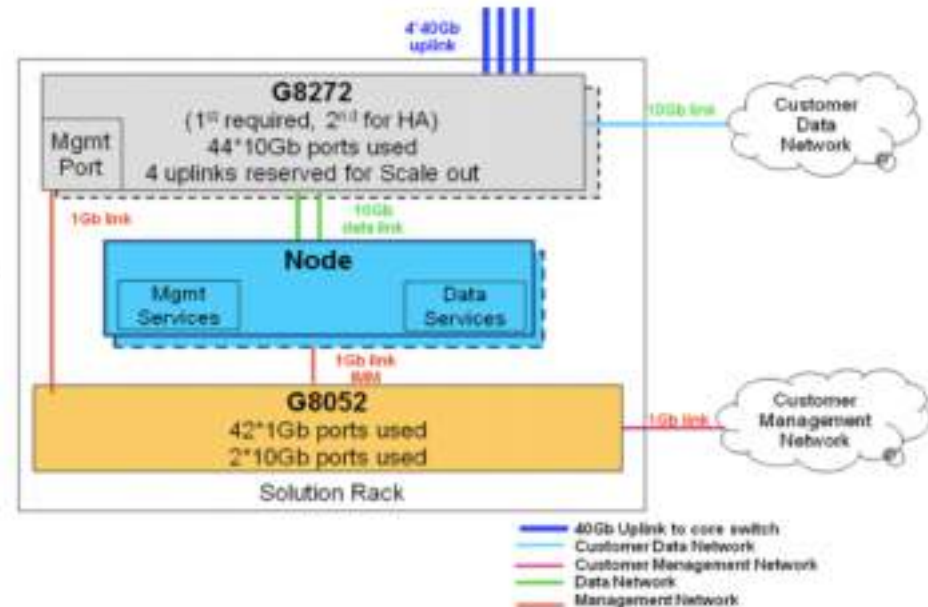
- 48 x SFP+ 10GbE ports plus 6 x QSFP+ 40GbE ports
- Support up to 72 x 10Gb connections using break-out cables
- 1.44 Tbps non-blocking throughput with low latency (~ 600 ns)
- Up to 72 1Gb/10Gb SFP+ ports
- OpenFlow enabled allows for easily created user-controlled virtual networks



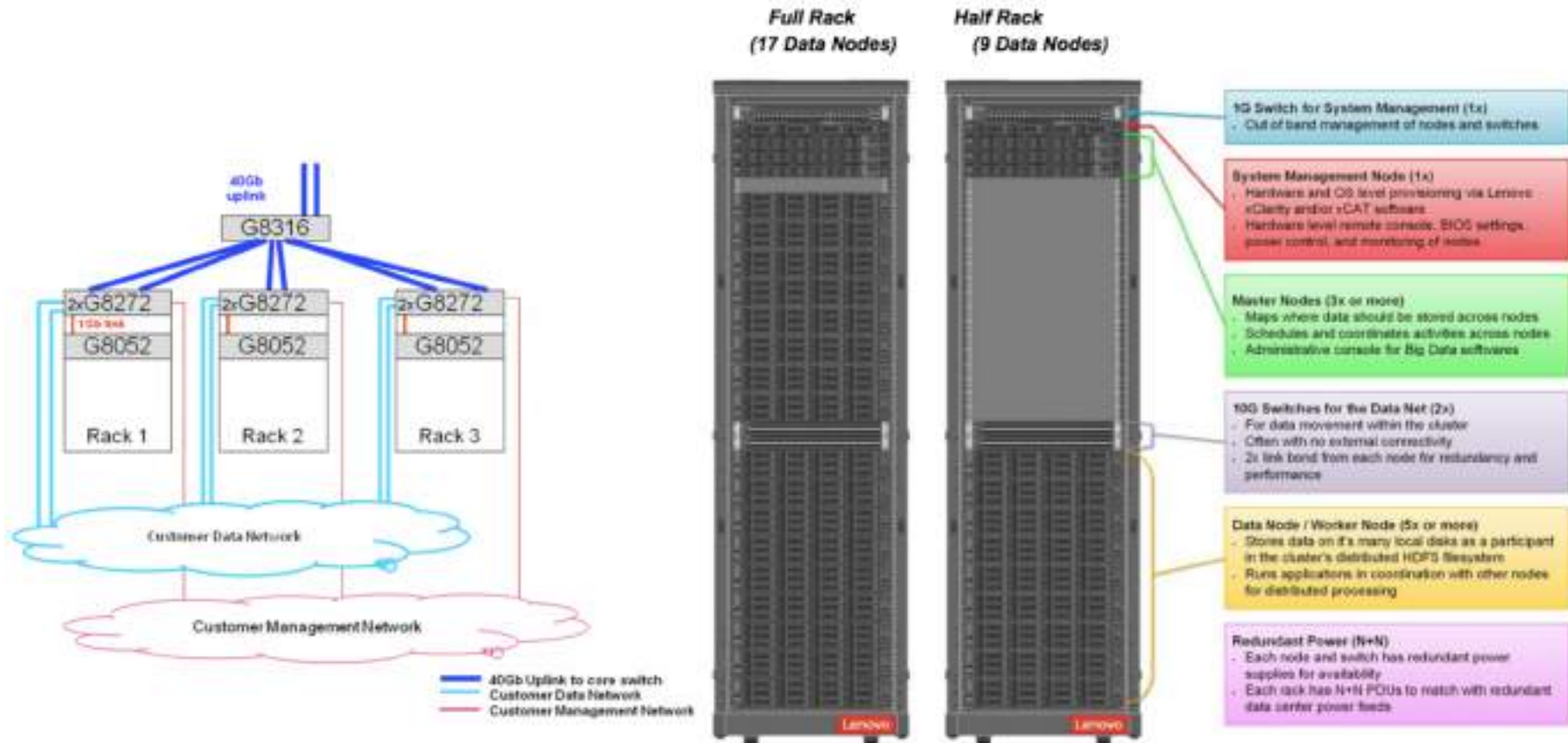
Figure 6. Lenovo RackSwitch G8052

Lenovo RackSwitch G8052 has the following characteristics:

- A total of 48 1 GbE RJ45 ports
- Four standard 10 GbE SFP+ ports
- Low 130W power rating and variable speed fans to reduce power consumption



Infraestructura para BigData: Cluster



Sistemas para BigData: Disk-based vs in-memory based .



Ref: Running Apache Spark on a High-Performance Cluster Using RDMA and NVMe Flash por Patrick Stuedi, IBM Research

Infraestructura para BigData: Repaso de conceptos

Debe entender y ser capaz de responder :

- ¿Qué es?
- ¿Qué se mejora?
- ¿Cuándo tiene sentido usarlo y que implica?

Para los siguientes conceptos:

- ✓ Servidor Físico, Virtual, Contenedor
- ✓ Hyperthreading (HT), SMT
- ✓ Cache, Memoria principal, NUMA
- ✓ SSD, HDD, NVMe SSD
- ✓ RAID, JBOD

Tipos de Red de Interconexión

SK-9821

Muchas posibilidades:

ATM, Myrinet, Gigabit Ethernet, Fast Ethernet, Infiniband

➤ Fast Ethernet (para gestion)

- La red barata más rápida disponible
- Ofrece un ancho de banda suficiente para la mayoría de situaciones.
- Hasta 100-1000 Mbps

➤ Gigabit Ethernet:

- Muy rápida (10, 40 y 100 Gbps)
- Coste decreciendo rápidamente.

➤ Infiniband:

- Muy rápida
- baja LATENCIA
- coste mas alto

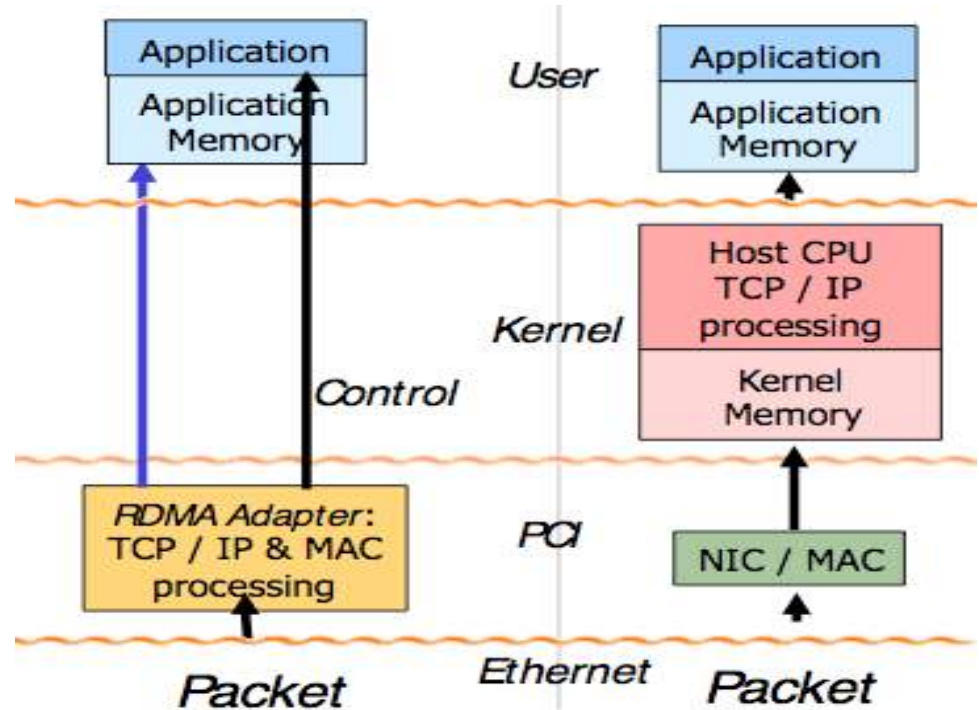
Caudal de Infiniband, bruto / eficaz

	SDR	DDR	QDR
1X	2,5 / 2 Gbps	5 / 4 Gbps	10 / 8 Gbps
4X	10 / 8 Gbps	20 / 16 Gbps	40 / 32 Gbps
12X	30 / 24 Gbps	60 / 48 Gbps	120 / 96 Gbps

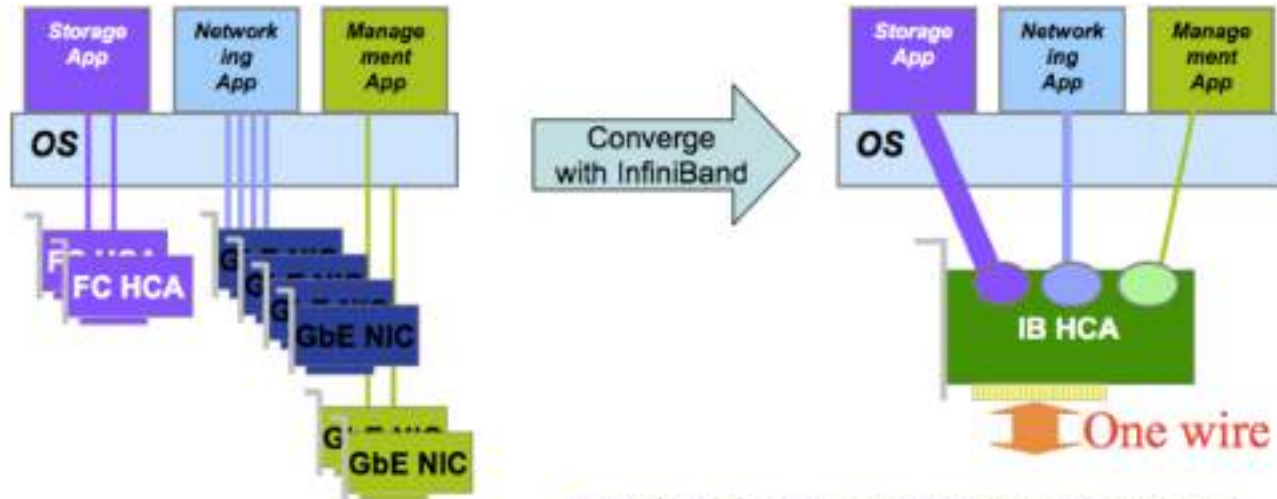


Inteconexión: Redes de baja latencia (Low Latency Interconnects)

- Objetivo: Disminuir la latencia para un paquete reduciendo el número de copias por paquete.



Infraestructura para BigData: Convergencia con InfiniBand



- Slower I/O
- Different service needs – different fabrics
- No flexibility

- High bandwidth pipe for capacity provisioning
- Dedicated I/O channels enable convergence
 - For Networking, Storage, Management
 - Application compatibility
 - QoS - differentiates different traffic types
 - Partitions – logical fabrics, isolation

Remote Direct Memory Access

❖ Remote

- data transfers between nodes in a network

❖ Direct

- no Operating System Kernel involvement in transfers
- everything about a transfer offloaded onto Interface Card

❖ Memory

- transfers between user space application virtual memory
- no extra copying or buffering

❖ Access

- send, receive, read, write, atomic operations

Arquitecturas para BigData: RDMA

Similitudes y diferencias entre TCP y RDMA

- ❖ Ambas utilizan el modelo cliente-servidor
- ❖ Ambas requieren de una conexión para transporte fiable
- ❖ Ambas proporcionan un modo de transporte fiable
 - TCP garantiza secuencias en orden de **bytes**
 - RDMA garantiza secuencias en orden de **mensajes**

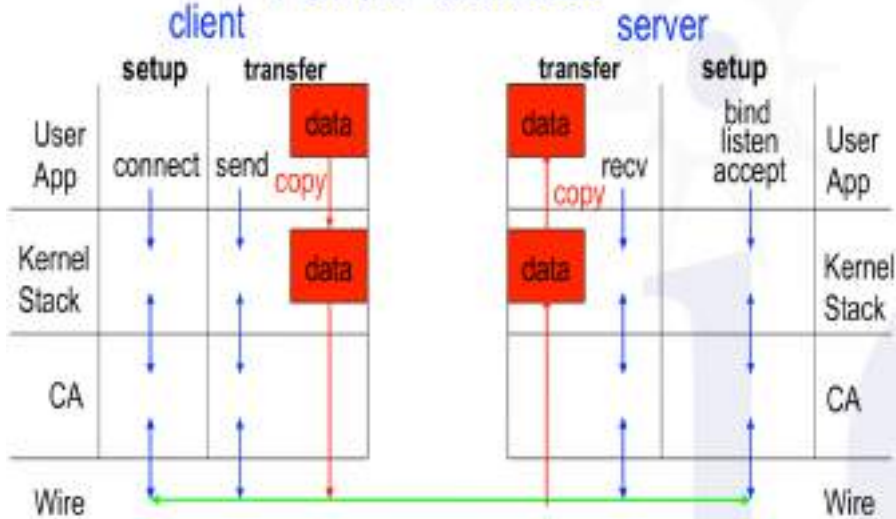
RDMA aporta :

- ❖ “zero copy” – datos transferidos directamente de memoria virtual de un nodo a memoria virtual de otro nodo
- ❖ “kernel bypass” – no involucra al sistema operativo en las transferencias de datos
- ❖ Operación asíncrona – Los threads no se bloquean durante la transferencia de I/O

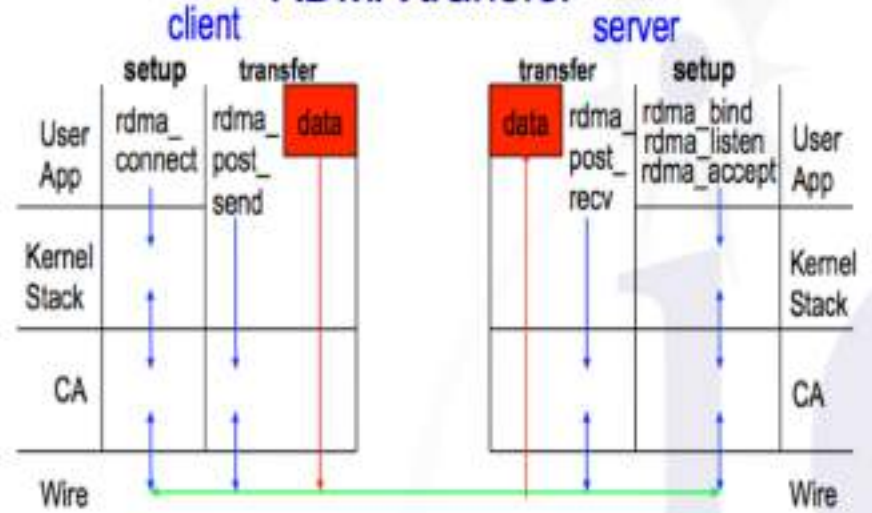
Arquitecturas para BigData: RDMA

Diferencias entre RDMA y TCP/IP

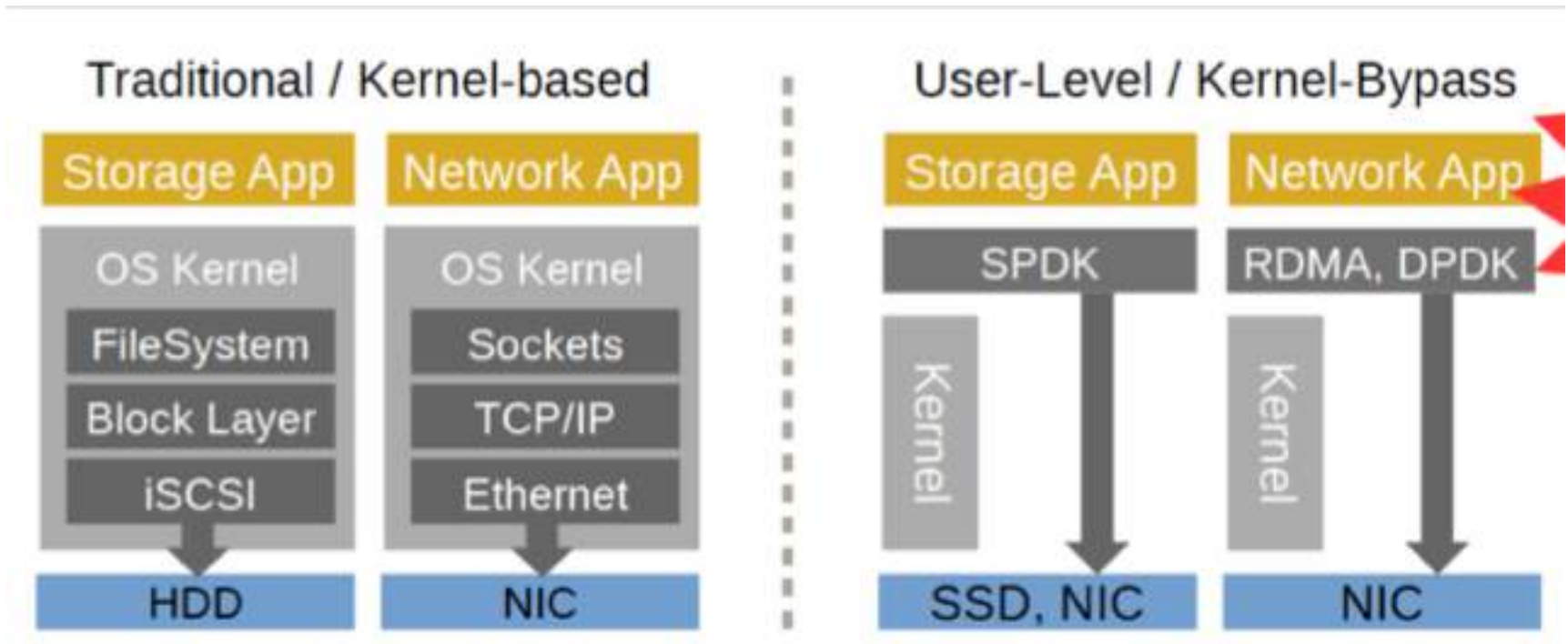
TCP/IP transfer



RDMA transfer

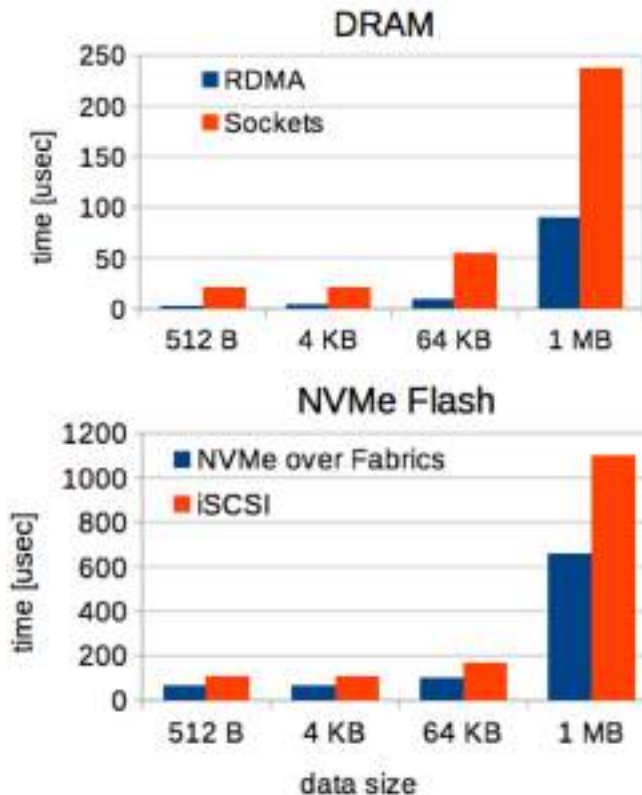
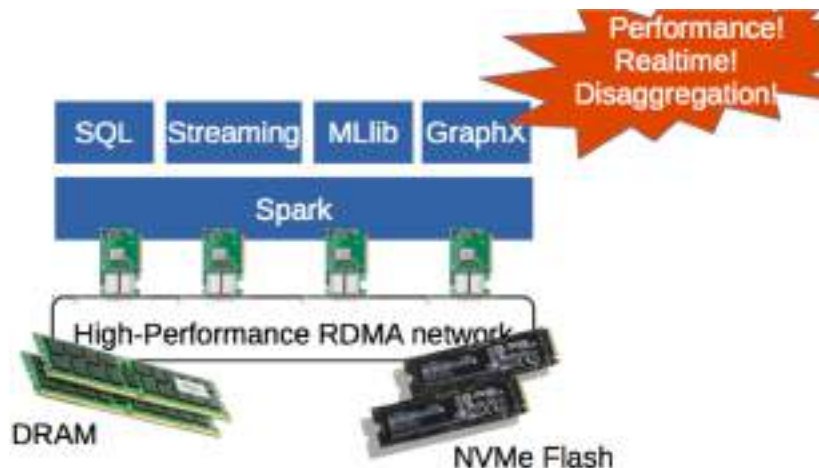


Arquitecturas para BigData: Liberar la CPU



Ref: Running Apache Spark on a High-Performance Cluster Using RDMA and NVMe Flash por Patrick Stuedi, IBM Research

Mejoras: Red con RDMA y Almacenamiento con NVMe



Ref: Running Apache Spark on a High-Performance Cluster Using RDMA and NVMe Flash por Patrick Stuedi, IBM Research



Infraestructura para BigData: Repaso de conceptos

Debe entender y ser capaz de responder :

- ¿Qué es?
- ¿Qué se mejora?
- ¿Cuándo tiene sentido usarlo y que implica?

Para los siguientes conceptos:

- ✓ Transferencias RDMA
- ✓ RDMA vs transferencia TCP/IP
- ✓ Zero copy
- ✓ Latencia Infiniband/Omnipath
- ✓ Redes en sistemas Cloud

Evolución de las Tecnologías para BigData

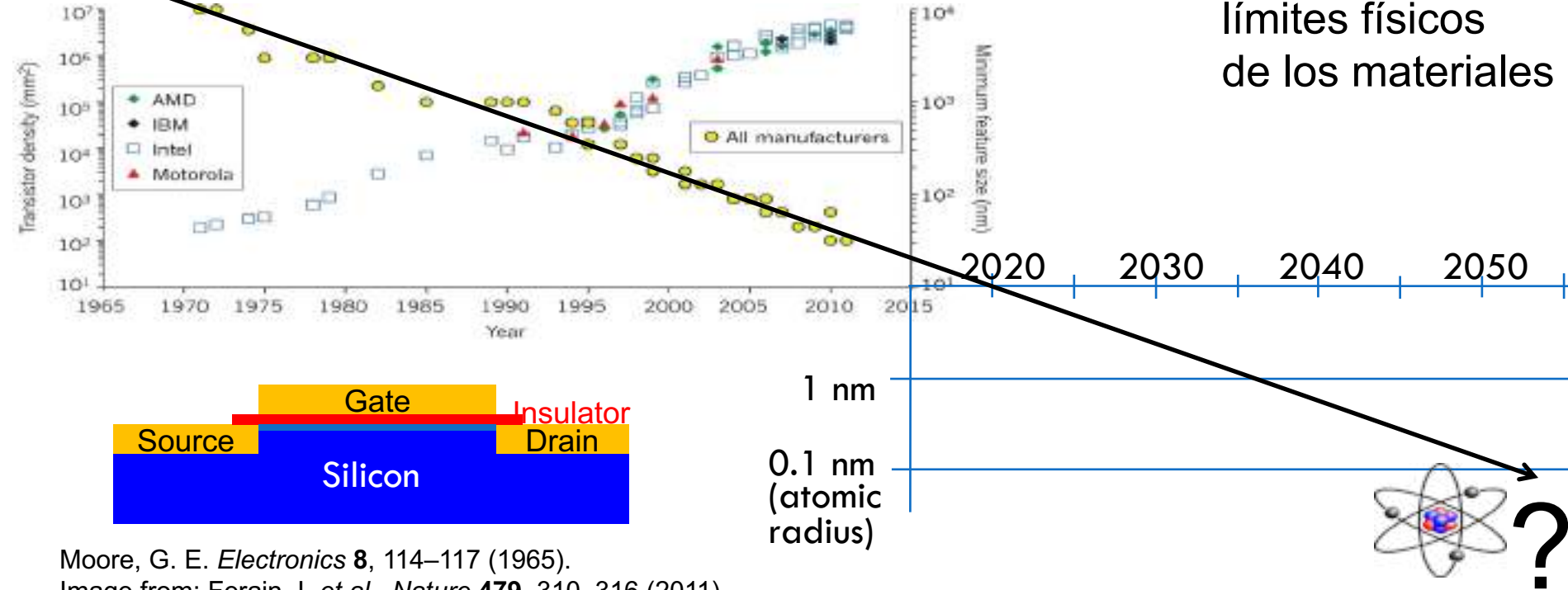
- ◆ Tecnologías actuales de computación para BigData
 - Sistemas Multicore
 - **Coprocesadores: GPUs, FPGA**
- Tecnologías disruptivas:
 - Neurocomputación
 - ◆ Computación Cuántica

Futuro de los procesadores

Ley de Moore

...

pronto entraremos en los
límites físicos
de los materiales

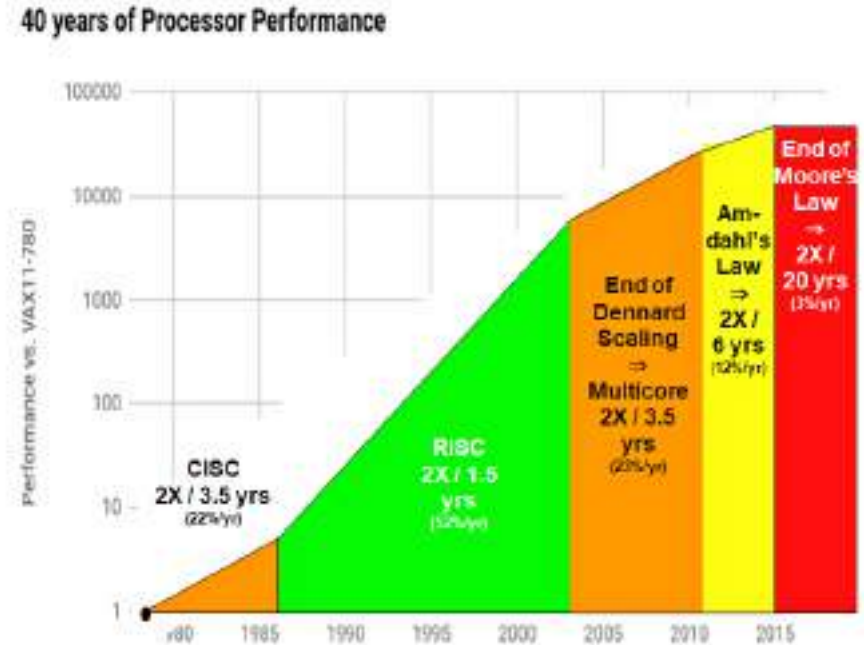


Procesadores : Evolución del rendimiento

- El rendimiento de los procesadores de propósito general se está estancando:

Se necesita nuevas tendencias para dar soluciones:

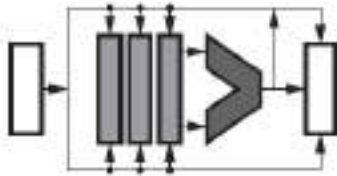
- Aceleradores con arquitecturas específicas para cada dominio.
- Tecnologías más disruptivas: procesadores cuánticos, neurocomputación.
- Computación aproximada.



Necesidades de computación

CPU

Scalar Processing



Complex Algorithms
and Decision Making

FPGA

Adaptable Hardware



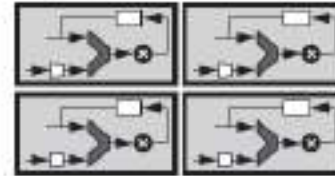
Processing of
Irregular Data Structures
Genomic Sequencing

Latency
Critical Workloads
Real-Time Control

Sensor Fusion
Pre-processing, Programmable I/O

GPU

Vector Processing (e.g., GPU, DSP)



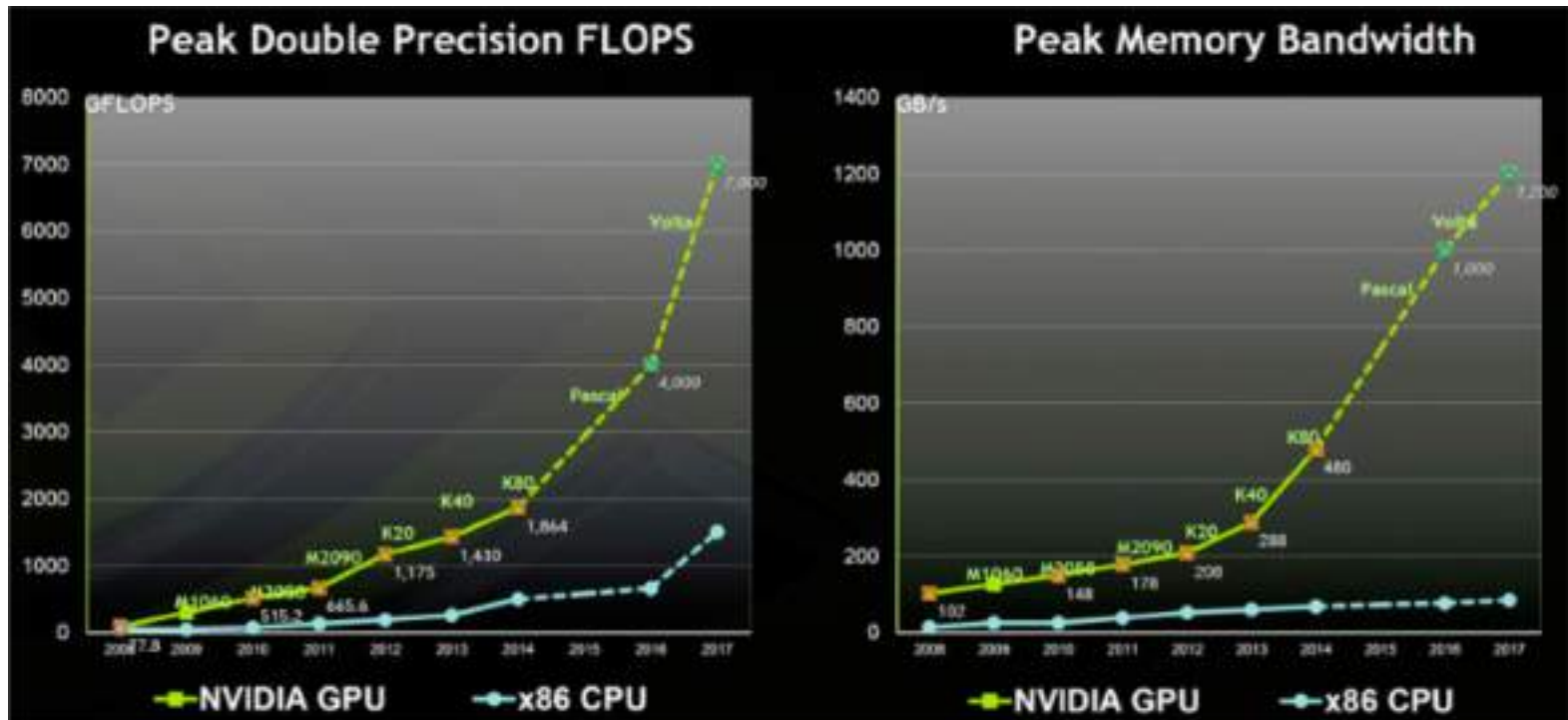
Domain-specific
Parallelism

Signal Processing
Complex Math, Convolutions

Video and
Image Processing

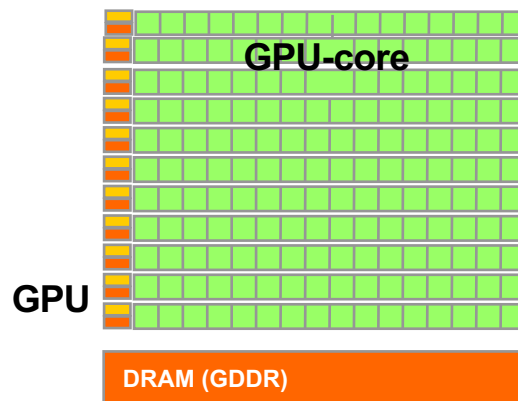
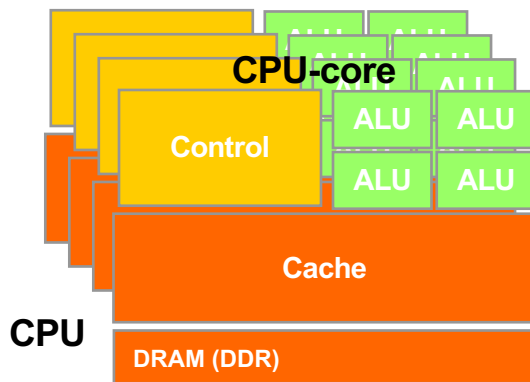
WP505_02_092918

CPU vs GPUs Performance Trends



CPU vs GPUs

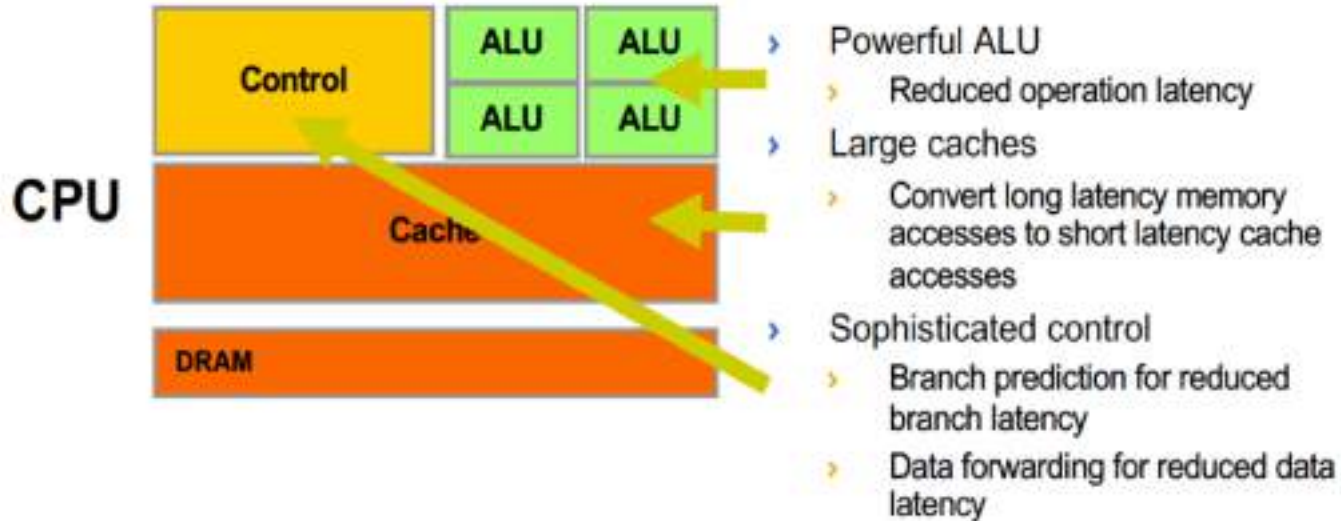
- The GPU is specialized for compute-intensive, highly data parallel computation (what graphics rendering is about)
- So, more transistors can be devoted to data processing rather than data caching and flow control



Different design philosophies

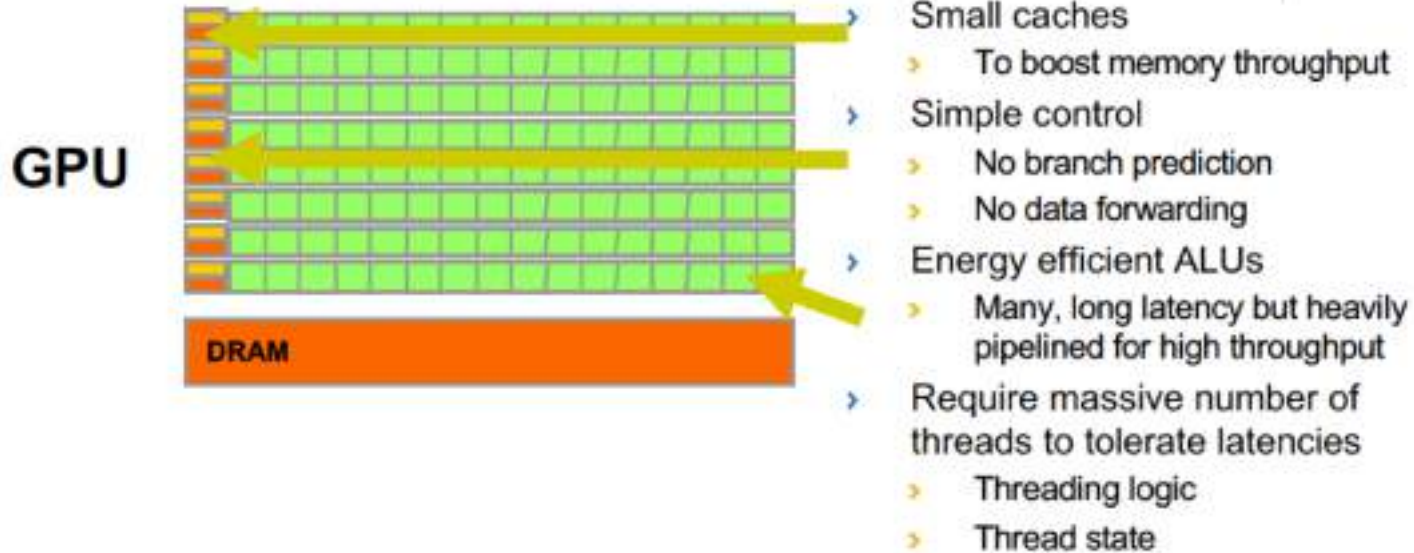
CPU: A few out-of-order cores vs GPU: Many in-order cores

CPU: Latency Oriented Design



CPU for sequential parts where latency matters CPU can be 10X+ faster than GPU for sequential code

GPUs: Throughput Oriented Design



GPUs for parallel parts where throughput wins GPUs can be 10X+ faster than CPUs for parallel code

Infraestructura BigData: Aceleradores/Coprocesadores GPU

CPU

vs

GPU



Optimized for **low latency**

- + Large main memory
- + Fast clock rate
- + Large caches
- + Branch prediction
- + Powerful ALU
- Relatively low memory bandwidth
- Cache misses costly
- Low performance per watt



Optimized for **high throughput**

- + High bandwidth main memory
- + Latency tolerant (parallelism)
- + More compute resources
- + High performance per watt
- Limited memory capacity
- Low per-thread performance
- Extension card

Review of GPU Architecture

Streaming Multiprocessors (SM)

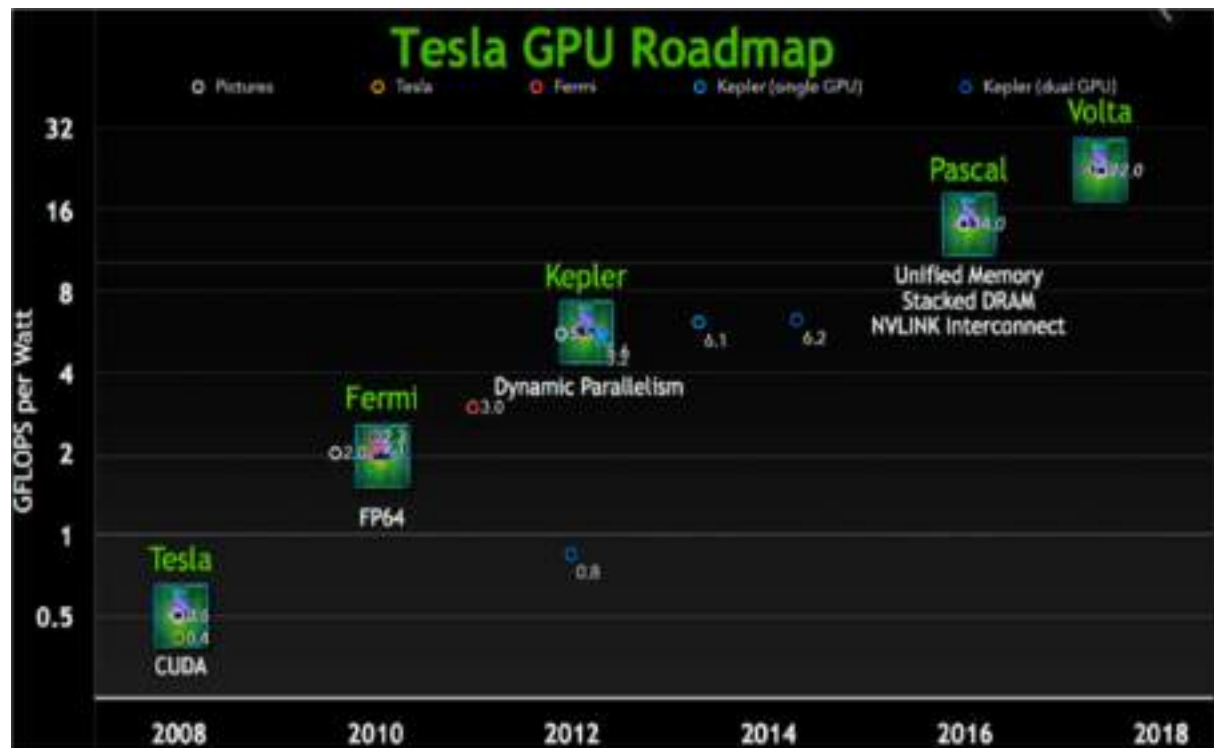
- ▣ Compute Units (CU)

Streaming Processors (SP) or CUDA cores

- ▣ Vector lanes

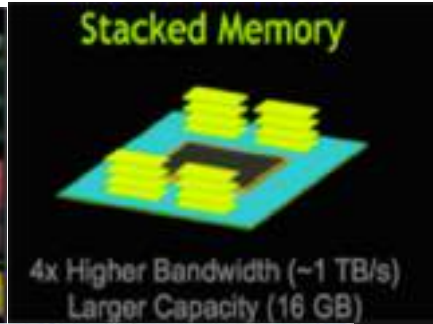
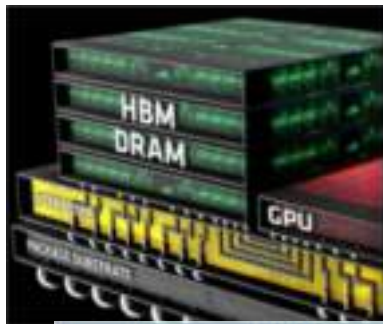
Number of SMs x SPs

- ▣ Tesla (2007): 30 x 8
- ▣ Fermi (2010): 16 x 32
- ▣ Kepler (2012): 15 x 192
- ▣ Maxwell (2014): 24 x 128
- ▣ Pascal (2016): 56 x 64
- ▣ Volta (2018): 80 x 64



GPU: Improving Features

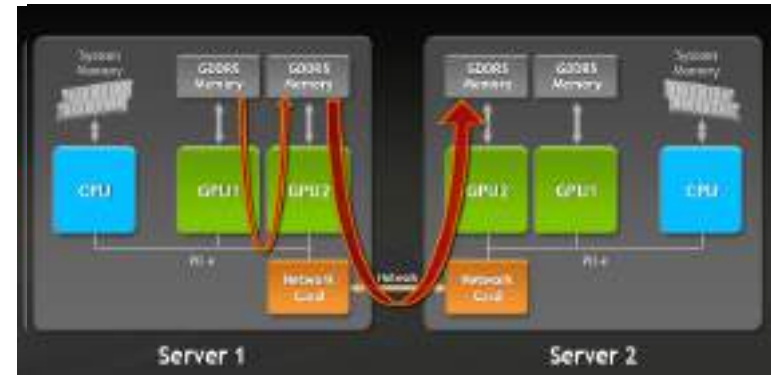
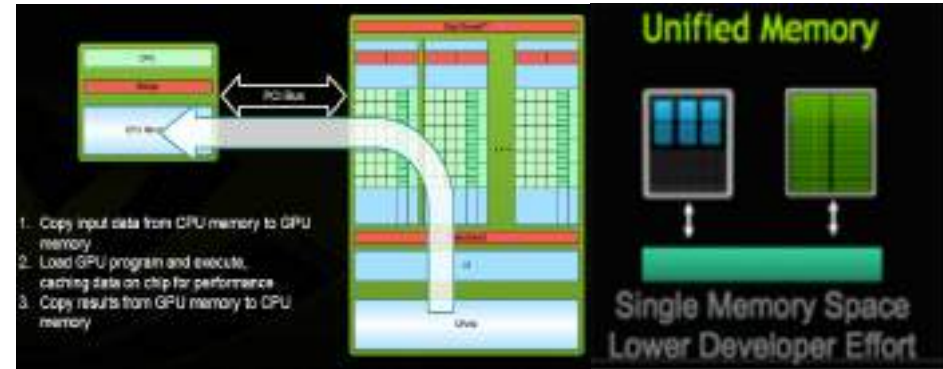
HBM High Bandwidth Memory



Item	DDR3 (x8)	QDDR5 (x32)	4-Hi HBM (x1024)
I/O	8	32	1024
Prefetch (Per IO)	8	8	?
Access Granularity (xIO x Prefetch)	8Byte	32Byte	256Byte
Max. Bandwidth	2GB/s	28GB/s	128-256GB/s

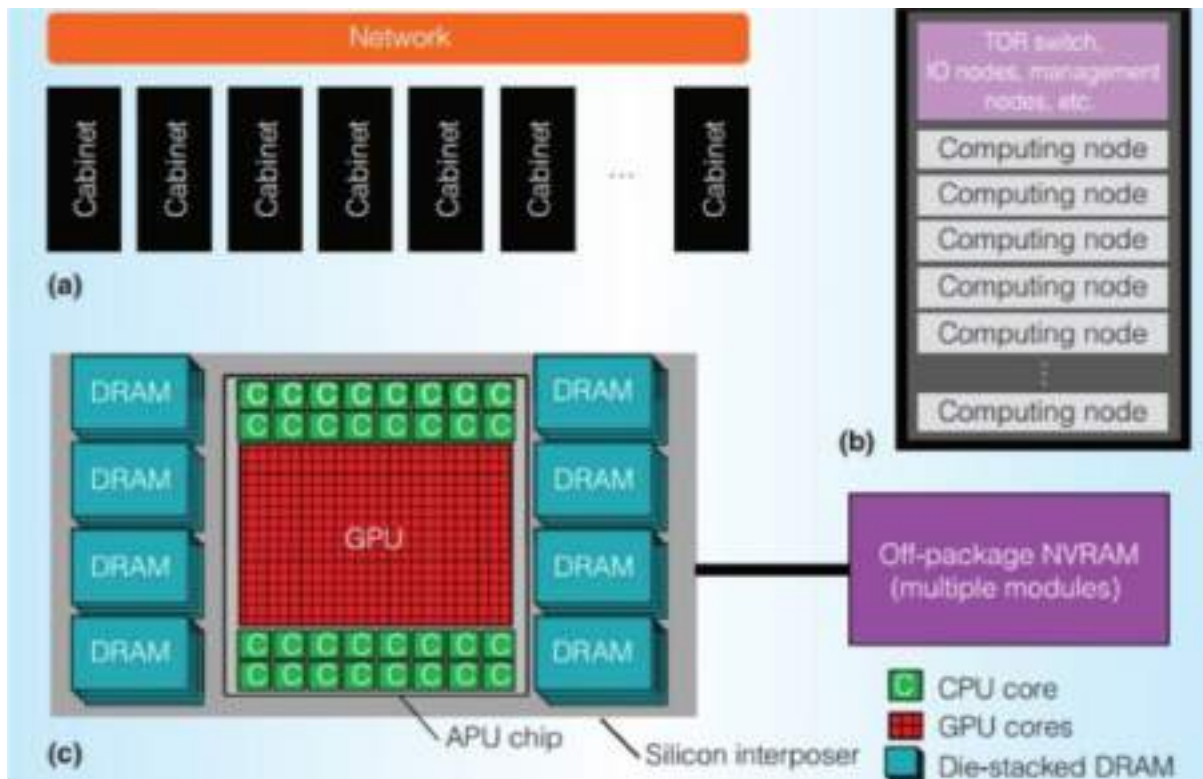


Full GPUDirect



CPU AMD: Adaptar el procesador a la computación

AMD Exascale Heterogeneous Processor: CPU, GPU y memoria HBM2 en una CPU

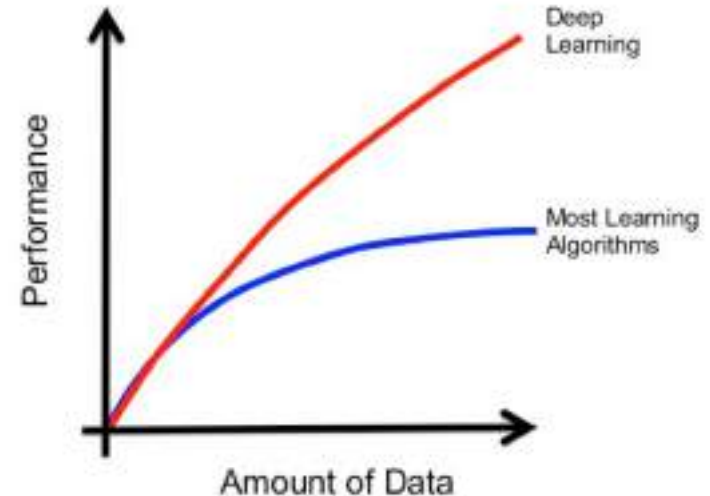


Caso de integración: Deep Learning usando GPUS con TensorFlow y Spark

Deep learning uses general learning algorithms

- The algorithms need to build the layers of an artificial neural network
 - Training data
- Processing this training data requires lots of computation
 - Convolutional NN -> Matrix multiplications

BIG DATA & DEEP LEARNING



Source: <https://towardsdatascience.com/7-practical-deep-learning-tips-97a9f514100e>

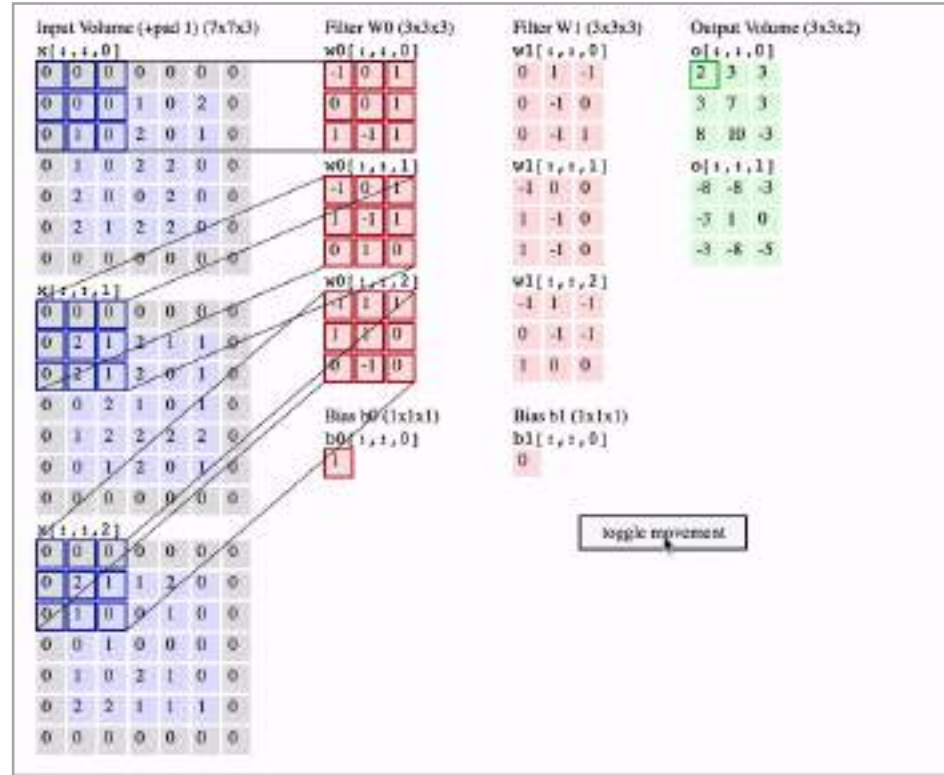
Deep Learning usando GPUs con TensorFlow y Spark

Se crea una red neuronal
Convolutiva

Operación básica:

$$\begin{matrix} a_1 \rightarrow \\ a_2 \rightarrow \end{matrix} \begin{bmatrix} 1 & 7 \\ 2 & 4 \end{bmatrix} \cdot \begin{matrix} b_1 \downarrow \\ b_2 \downarrow \end{matrix} \begin{bmatrix} 3 & 3 \\ 5 & 2 \end{bmatrix} = \begin{bmatrix} a_1 \cdot b_1 & a_1 \cdot b_2 \\ a_2 \cdot b_1 & a_2 \cdot b_2 \end{bmatrix} = \begin{bmatrix} 3 & 21 \\ 10 & 8 \end{bmatrix}$$

$A \quad B \quad C$



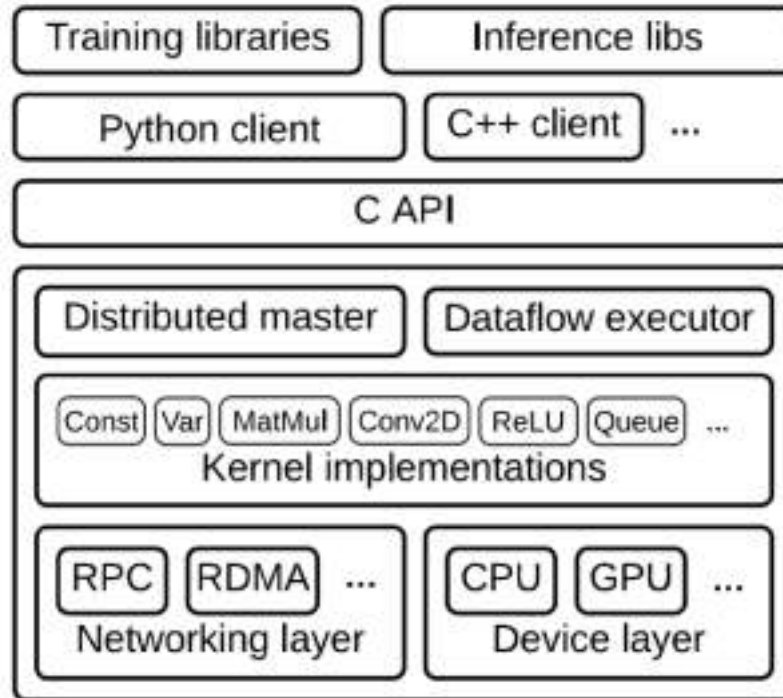
Source: <https://medium.com/@phidaouss/convolutional-neural-networks-cnn-or-convnets-d7c688b0a207>

Integración: Deep Learning usando GPUs con TensorFlow y Spark

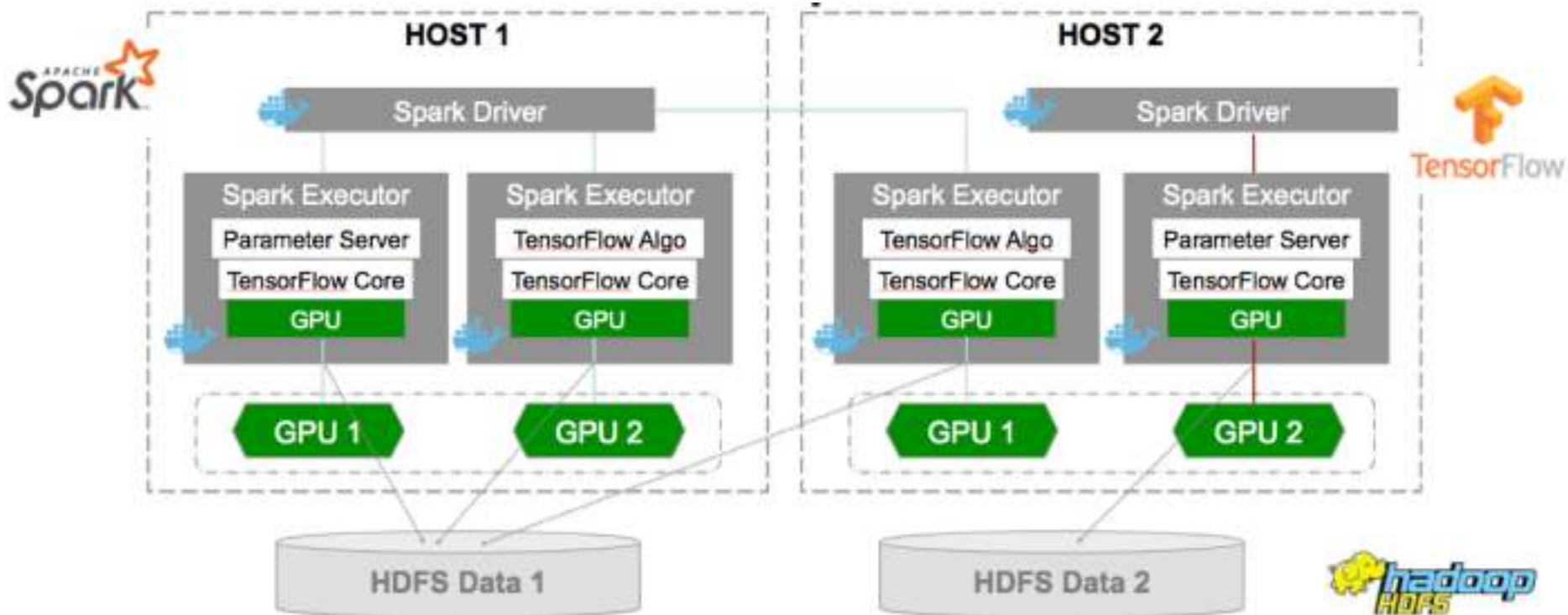
Arquitectura de TensorFlow



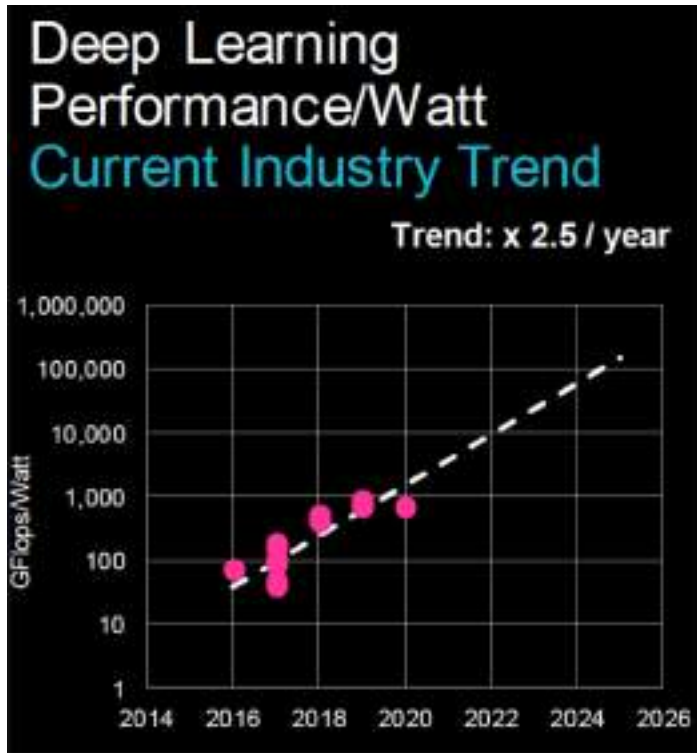
Source: www.tensorflow.org/extend/architecture



Deep Learning with TensorFlow and Spark



Deep Learning: GPU and FPGA



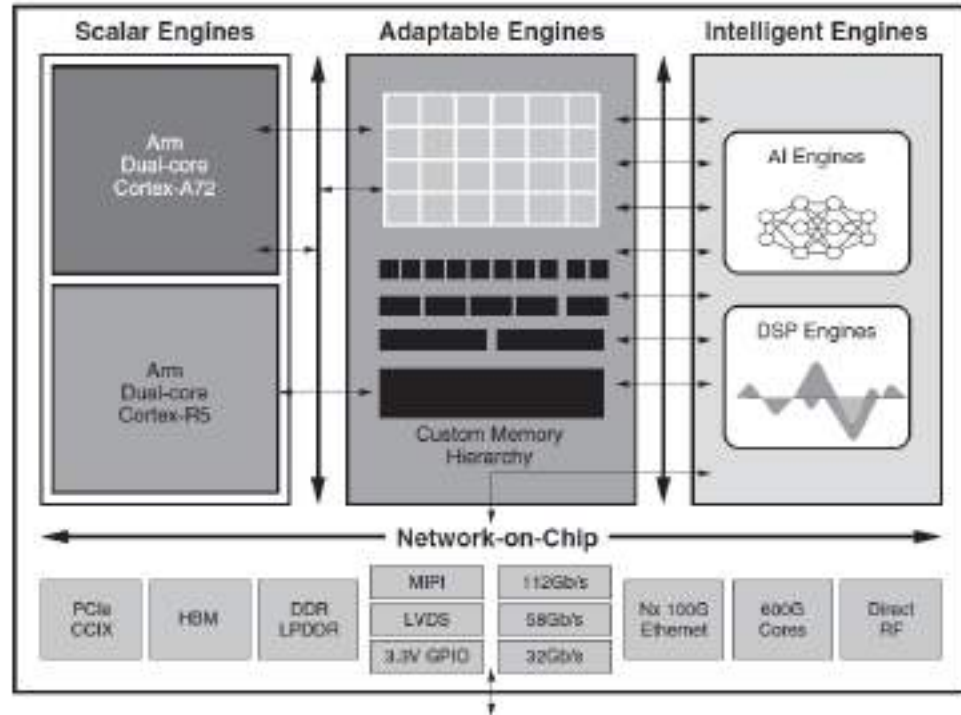
Los aceleradores actuales y los que se van utilizar a medio plaza para deep Learning están basados en tecnología CMOS

- Evolución de GPUs.
- Nuevas posibilidades con FPGAs
- Diseño de ASICs específicos.

Arquitecturas para BigData: Coprocesadores FPGA

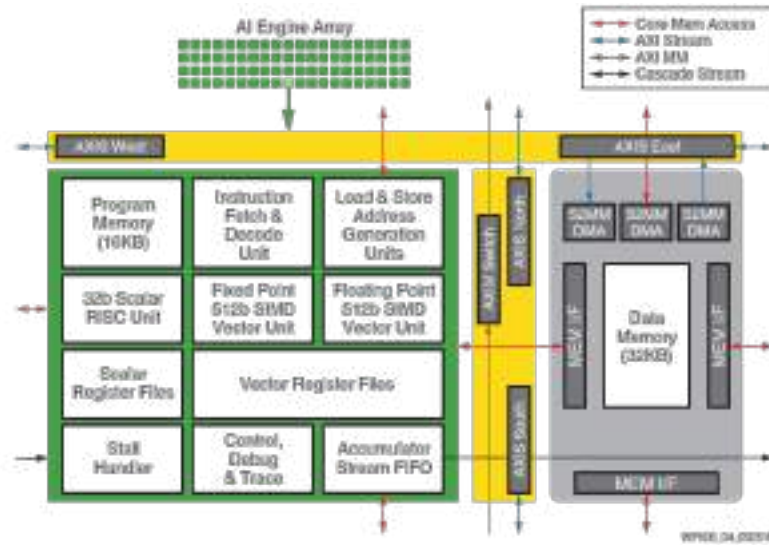
Hardware Reconfigurable
Capaz de integrar todas las
necesidades de computación:

Adaptive
Compute
Acceleration
Platform (ACAP)



FPGA Xilinx versal: Processing System

- Dual-core ARM A72 with 2x single-threaded performance of previous generation A53's
- Dual-core ARM R5 for real-time and deterministic processing
- Adaptable resources (FPGA)
- AI Engine



Coprocesadores FPGA



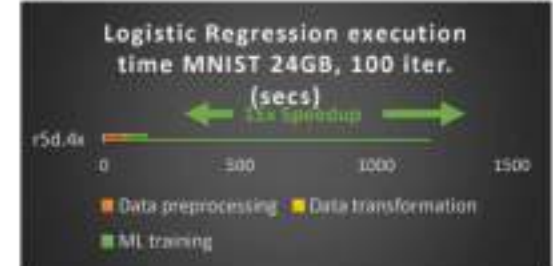
InAccel Coral:

- Program against your FPGAs like it's a single pool of accelerators
- “automated deployment, scaling, and management of FPGAs”

- > Up to **15x** speedup for **LR ML** (7.5x overall)
- > Up to **14x** speedup for **Kmeans ML** (6.2x overall)
- > Spark- GPU* (3.8x – 5.7x)

- > **F1.4x**
 - ⇒ 16 cores + 2 FPGAs (InAccel)
- > **R5d.4x**
 - ⇒ 16 cores

*[Spark-GPU: An Accelerated In-Memory Data Processing Engine on Clusters]



Coprocesadores FPGA

The main differentiating factor is that **can be reconfigured** as opposed to the other chips:

- It allows for specifying hardware description language (HDL) that can be in turn configured in a way that matches the requirements of specific tasks or applications.
- It is known to consume less power and offer better performance.
- It can also offer a cost-effective option for prototype. It is much more flexible and is, therefore, a good choice for applications that involve customer-centric applications

Advantages

- It is highly flexible and is suited for rapidly growing and changing AI applications. For instance, with neural networks improving, it provides an architecture to undergo changes
- It shows better performance and consumption ratio
- Offers high accuracy
- FPGA shows efficiency in parallel processing. Overall it has significantly higher computer capability
- FPGAs offer lower latency than GPUs

Disadvantages

- Difficult to program and Development time is more
- Performance may not be up to the mark sometimes
- Not good for floating-point operation.

Infraestructura para BigData: Repaso de conceptos

Debe entender y ser capaz de responder :

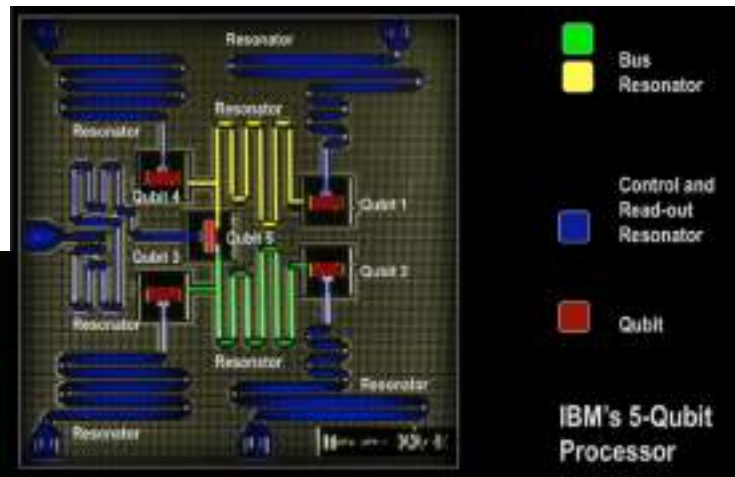
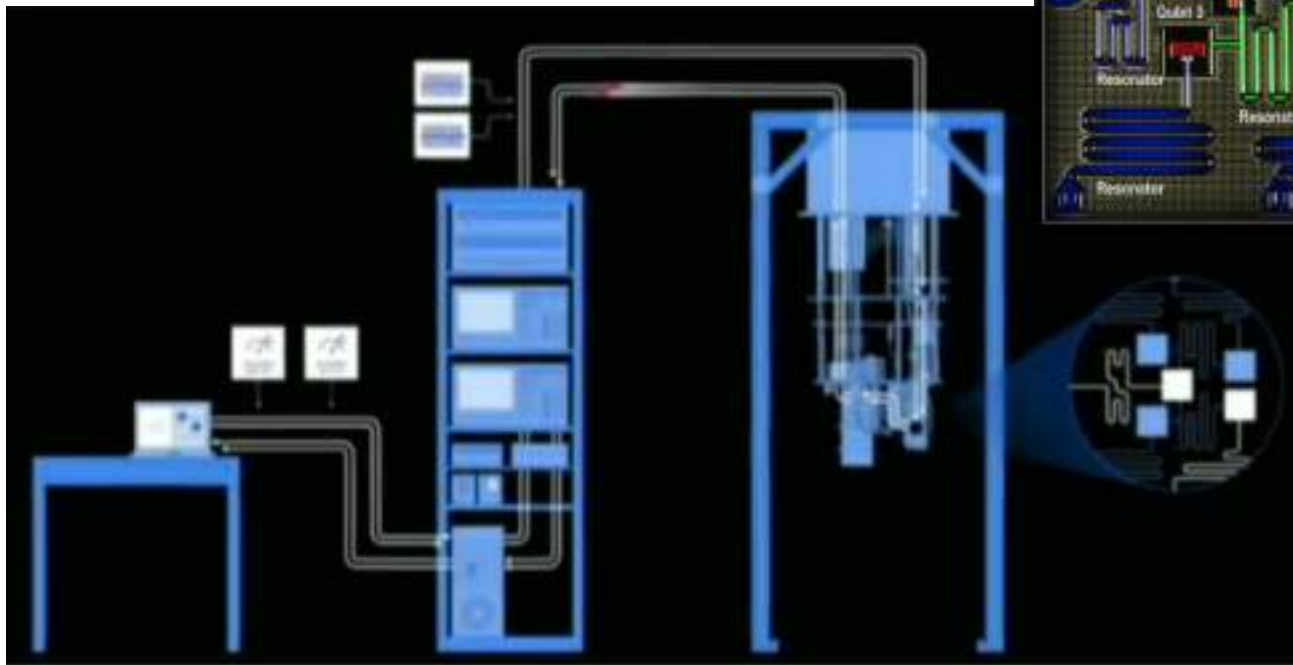
- ¿Qué es?
- ¿Qué se mejora?
- ¿Cuándo tiene sentido usarlo y que implica?

Para los siguientes conceptos:

- ✓ GPUs vs CPU
- ✓ Modelo de programación GPU
- ✓ FPGA
- ✓ Casos de optimización con GPU (TensorFlow)

Computador cuántico de IBM

Join the IBM Q Experience Community
<https://quantumexperience.ng.bluemix.net>



Referencias

1. ACCELERATING APACHE SPARK MACHINE LEARNING WITH CLEAR LINUX® OS FOR INTEL ARCHITECTURE® AND INTEL SOFTWARE OPTIMIZATIONS.<https://01.org/blogs/2018/apache-spark-clear-linux/>

2.- Architectural Impact on Performance of In-memory Data Analytics: Apache Spark Case Study

3.- Ref: Running Apache Spark on a High-Performance Cluster Using RDMA and NVMe Flash por Patrick Stuedi, IBM Research

