

# Uso de Aceleradores HW en Aplicaciones ML/AI

Iván González

# Agenda

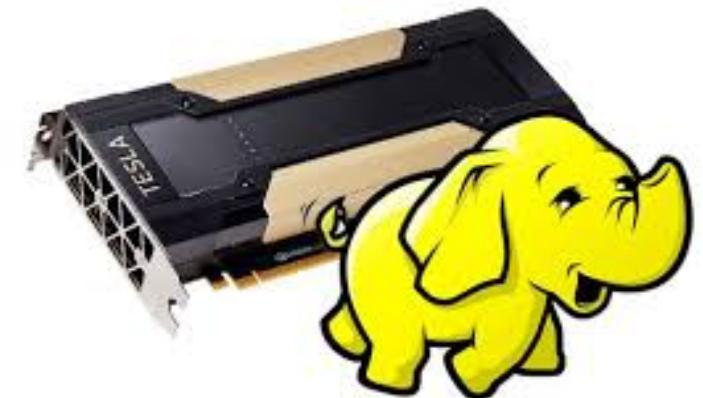
- Infraestructura para acelerar aplicaciones ML/AI
- Graphics Processing Units
  - GPUs en ML/AI
  - GPUs en Análisis de datos
- Field Programmable Gate Arrays
  - FPGAs en ML/AI
  - FPGAs en Análisis de datos
- GPUs y FPGAs en la nube
- Hadoop 3.1.0

# Infraestructura para acelerar aplicaciones ML/AI

- La revolución del BigData y las aplicaciones ML/AI está afectando a los diferentes ecosistemas ya existentes
  - Aparecen nuevas herramientas, librerías ML, etc.
  - Hadoop → Spark
- Al mismo tiempo, la infraestructura HW debe ser capaz de evolucionar
  - De otro modo no hay BigData, ML, AI, etc.
  - Debe cubrir las necesidades de estas nuevas aplicaciones

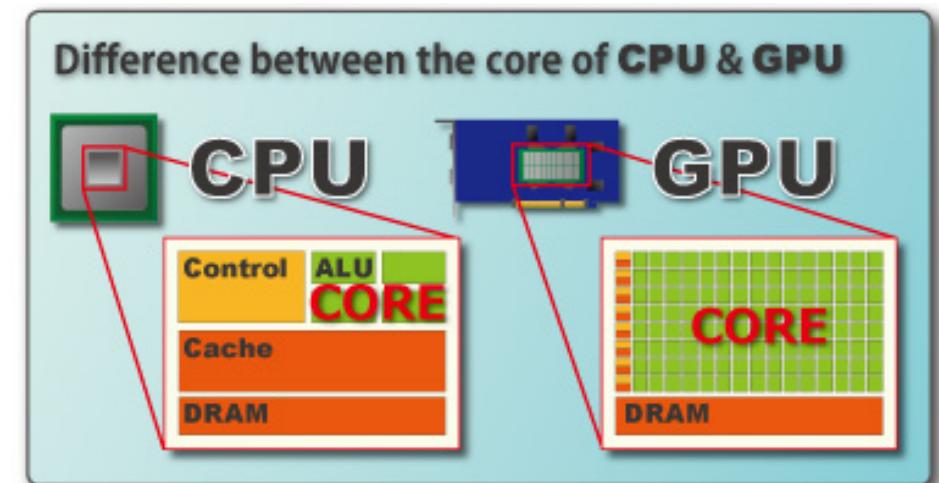
# Infraestructura para acelerar aplicaciones ML/AI

- Se requiere mucha capacidad de computo, almacenamiento, ancho de banda, etc.
- Para resolver las necesidades de computo se empieza a apostar por aceleradores HW
  - Ofrecen muchas UP (unidades de procesamiento)
  - Ofrecen mucho ancho de banda
- Particularmente
  - GPUs
  - FPGAs



# GPUs (Graphics Processing Units)

- Inicialmente desarrolladas para mejorar el rendering gráfico en la industria de los videojuegos.
- Su arquitectura se ha demostrado muy adecuada para ML
  - Fundamentalmente en la parte de entrenamiento de redes neuronales
- Proporcionan un alto grado de paralelismo
  - Las CPUs también pueden hacerlo



# GPUs (Graphics Processing Units)

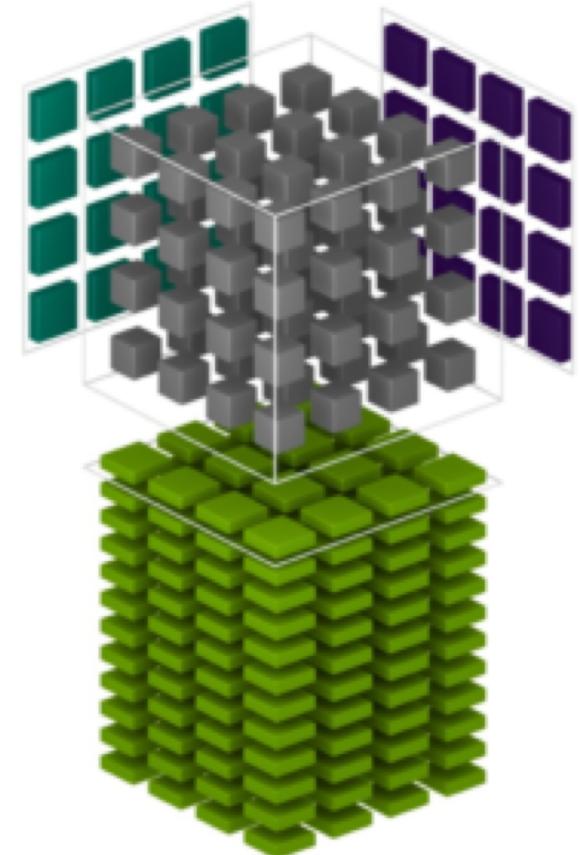
- Existen dos fabricantes AMD y Nvidia
  - Nvidia es la solución dominante gracias a CUDA
- Desde la arquitectura Volta, se incluyen Tensor cores
  - La nueva arquitectura Turing mejora el rendimiento
  - 500 trillones de operaciones tensor por segundo
  - La tecnología NGX para llevar el AI al pipeline gráfico
    - No útil para ML tradicional

GPU	Memory	Memory w/NVLink	Ray Tracing	CUDA Cores	Tensor Cores
Quadro RTX 8000	48GB	96GB	10 Giga rays/sec	4,608	576
Quadro RTX 6000	24GB	48GB	10 Giga rays/sec	4,608	576
Quadro RTX 5000	16GB	32GB	6 Giga rays/sec	3,072	384

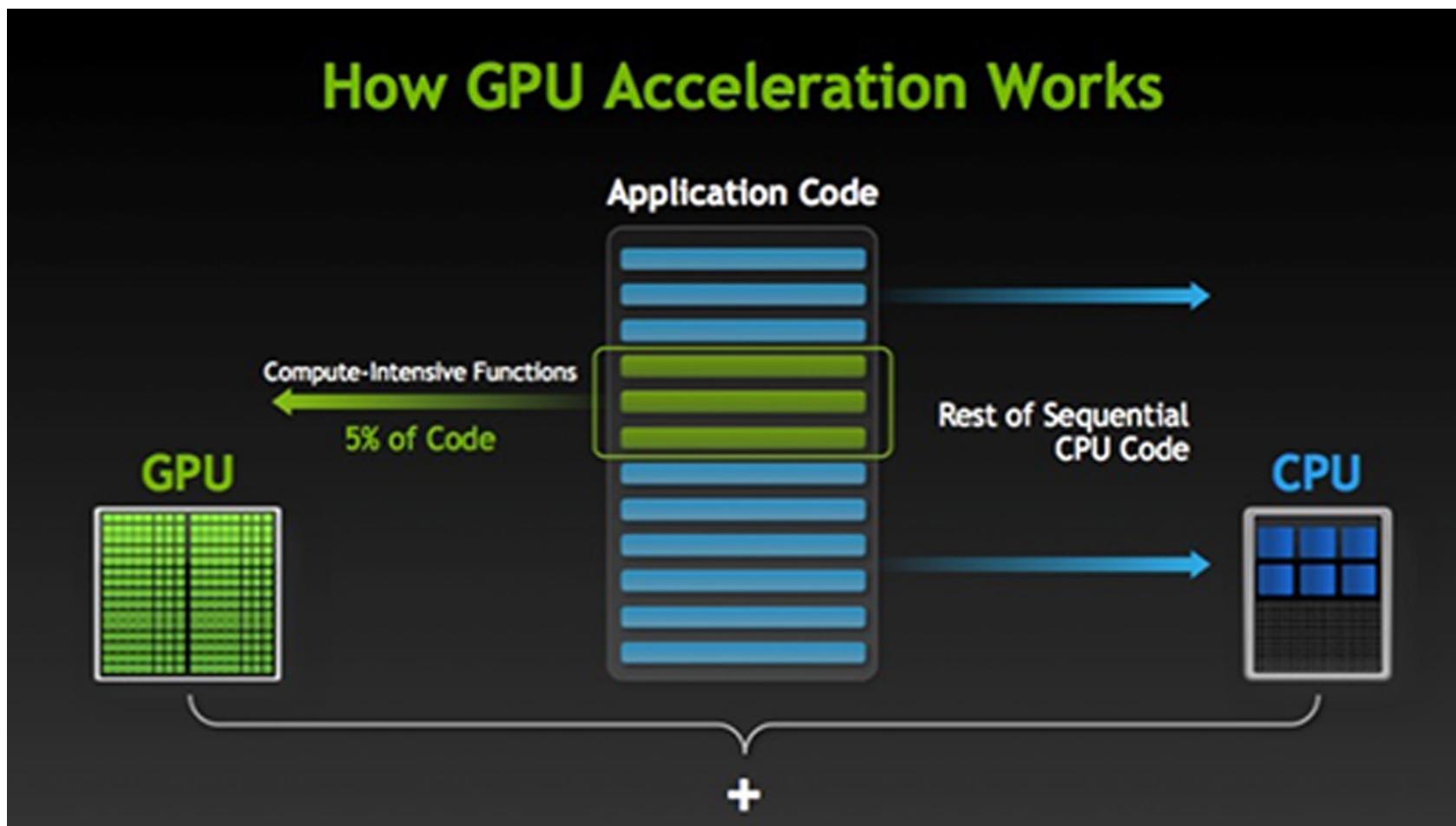
# GPUs (Graphics Processing Units)

## ➤ Tensor Core

- Permiten acelerar las operaciones en grandes matrices de datos
  - La función principal en AI
- Realizan la operación FP16 de multiplicación y acumulación ( $x += y * x$ ) en 1 ciclo de reloj GPU
  - Multiplica dos matrices FP16 4x4 y suma el producto de la multiplicación (FP32 4x4) al acumulador (FP32 4x4)
  - Precisión mixta



# GPUs (Graphics Processing Units)

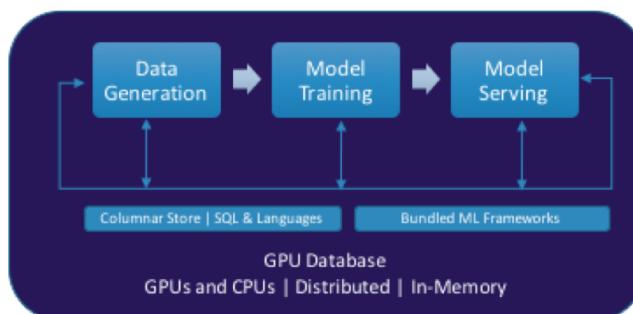


# GPUs en ML

- Nvidia incluye una librería ML/DL en CUDA: cuDNN
- La mayoría de frameworks ML con soporte para GPUs emplean cuDNN:
  - Caffe2, Chainer, Keras, Matlab, MxNet, PyTorch y TensorFflow.
- Para programar tus propios modelos tendrías que usar CUDA, pero la mejor opción es OpenCL
  - Soporta también AMD
  - Menos desarrollada que CUDA

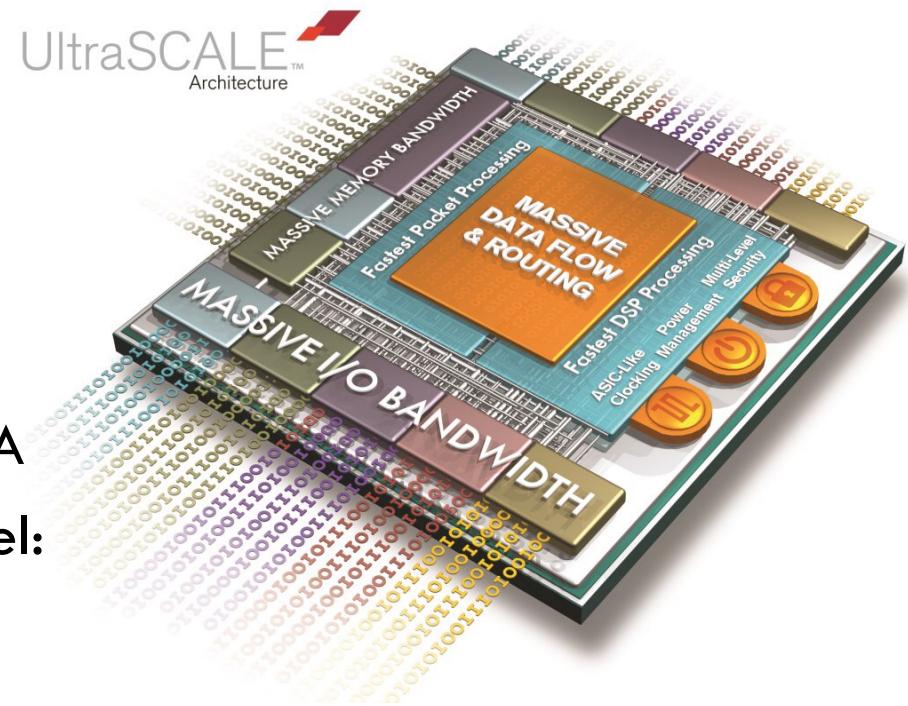
# GPUs en Análisis de Datos

- **Bases de datos con GPUs**
  - Emplean GPUs para realizar operaciones en la base de datos
  - Generalmente se ejecutan en la nube por su alto coste
    - Si se usan decenas de GPUs
  - Las GPUs ofrecen el rendimiento necesario para unificar y operar todo el pipeline de la base de datos, incluidas capacidades analíticas o ML
  - OmniSci, Sqream, Kinetica, BlazingDB, BrytlytDB, MapD, PG-Strom, y SQream.



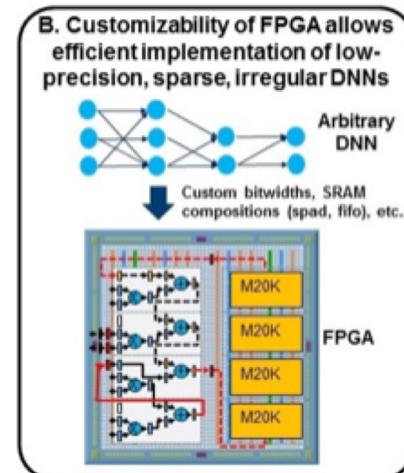
# FPGAs (Field Programmable Gate Arrays)

- Chips reconfigurables
  - Xilinx y Altera
- Su uso es limitado debido a las herramientas de desarrollo
  - Se considera “dark art”
  - Sería necesario un “CUDA” para FPGA
  - Hay algunas herramientas de alto nivel:
    - Compiladores de C
    - Toolkits específicas



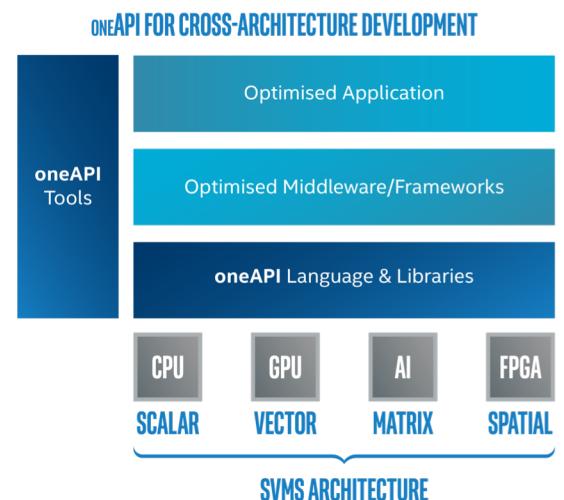
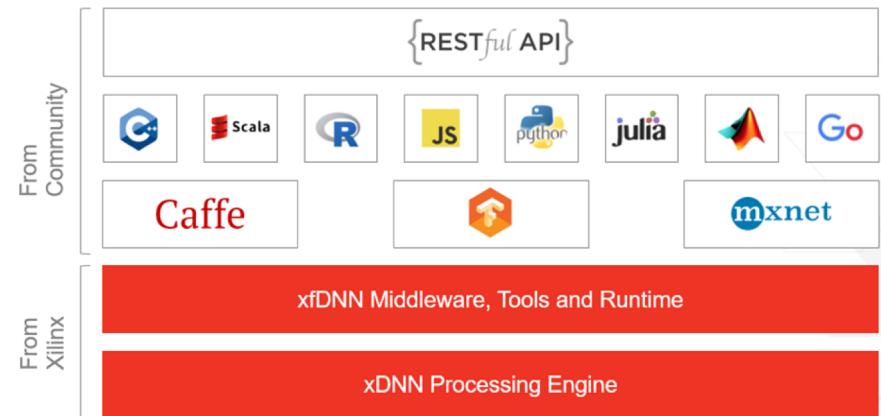
# FPGAs (Field Programmable Gate Array)

- Intel ha mostrado interés en esta tecnología comprando Altera
  - Tecnología de interconexión
  - Se entiende como maniobra para competir contra Nvidia
    - Las soluciones multicore como Xeon Phi, etc. no parecen cumplir expectativas
  - Algunos resultados publicados muestran rendimientos superior a GPUs
    - Optimizando aritmética
    - Baja precisión es la clave
    - En DL puede tener sentido



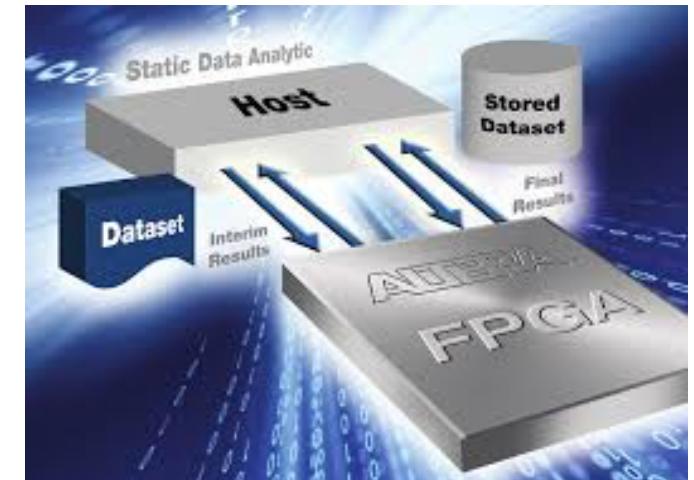
# FPGAs en ML

- No existen muchas soluciones todavía
  - Desarrollos “beta”
  - Trabajos académicos
- Xilinx Machine Learning Suite
  - Soportado por varios frameworks
- Intel oneAPI
  - Un API para “programarlos a todos”



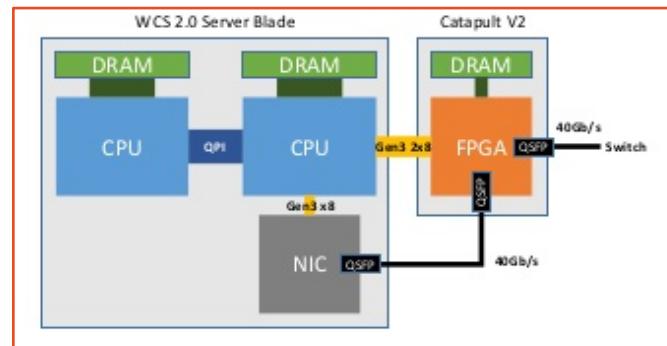
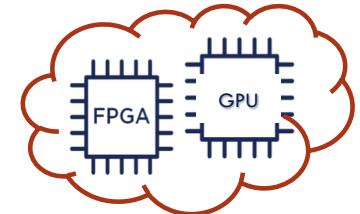
# FPGAs en Análisis de datos

- **Bases de datos con FPGAs**
  - Acelerar las operaciones SQL
    - Netezza (ahora de IBM) fue uno de los precursores
  - Swarm64 (startup) añade una capa de aceleración que mejora el rendimiento x12 en PostgreSQL, MarAIDB y MySQL
    - Indican soporte para Oracle o SQL Server
  - rENAIC ofrece algo similar para Cassandra



# GPUs y FPGAs en la nube

- GPUs disponibles en AWS, Azure y Google Cloud
  - La mayoría usa GPUs de Nvidia
- FPGAs disponibles en AWS (EC2 F1)
  - FPGAs de Xilinx
  - No incluye servicios con soporte para ML
- Azure
  - FPGAs de Intel (Altera)
  - Incluye modelos pre-entrenados
    - “Soft” DNN processor (DPU)
    - <https://docs.microsoft.com/en-us/azure/machine-learning/how-to-deploy-fpga-web-service>
- Google Cloud ofrece sus TPU (TensorFlow Processing Units)
  - Biblioteca de modelos optimizados



- The architecture justifies the economics
  1. Can act as a local compute accelerator
  2. Can act as a network/storage accelerator
  3. Can act as a remote compute accelerator

# Hadoop 3.1.0

- Añade soporte para GPUs y FPGAs en Yarn
  - Como recursos computaciones independientes
  - Solo soporte para el “IntelFPGAOpenCLPlugin”
  - Solo soporte para GPUs de Nvidia
- Los node managers deben instalarse con los drivers correspondientes y configurados adecuadamente
- Configuración de Yarn para GPUs
  - <https://hadoop.apache.org/docs/r3.1.0/hadoop-yarn/hadoop-yarn-site/UsingGpus.html>
- Configuración de Yarn para FPGAs
  - <https://hadoop.apache.org/docs/r3.1.0/hadoop-yarn/hadoop-yarn-site/UsingFPGA.html>

# Uso de GPUs en Google Colaboratory (Colab)

- Permite ejecutar y programar en Python en tu navegador con las siguientes ventajas:
  - No requiere configuración
  - Da acceso gratuito a GPUs
  - Permite compartir contenido fácilmente

The screenshot shows the Google Colaboratory interface with the following details:

- Title Bar:** Ejemplo\_programar\_GPU
- Left Sidebar (Índice):**
  - Check HW
    - Install the environment
    - Run a program (using Google overpowered GPUs)
  - Simple mode, everything on the same file
  - Running code in multiple source files
  - Run other programs (even not using the GPU)
- Code Editor:** Contains CUDA C code for matrix multiplication and memory copy.
- Output Area:** Shows the command `incc cuda/\*.cu -o cuda/program` and its output, which includes the running kernel and a long list of numerical results.