Survey Paper - Google TPU

Submitted by:
Ripunjay Narula(19BCE0470)
and Samvit
Swaminathan(19BCE0629)

Understanding the Increasing Demands of Modern Deep Neural Networks

An understanding of Neural Network (NN) progression is important to understand the hardware being compared in this article. The current trend in NN programming is to create "deeper" NNs, called Deep Neural Networks (DNN). DNNs have increased the accuracy of NNs by reducing speech recognition errors, increasing the accuracy of image recognition, and beating human challengers in games such as Go and Jeopardy. This accuracy comes at a cost both when training a model and making predictions, called "inference." These DNNs are trained on larger data sets and may have increasing number of input parameters.

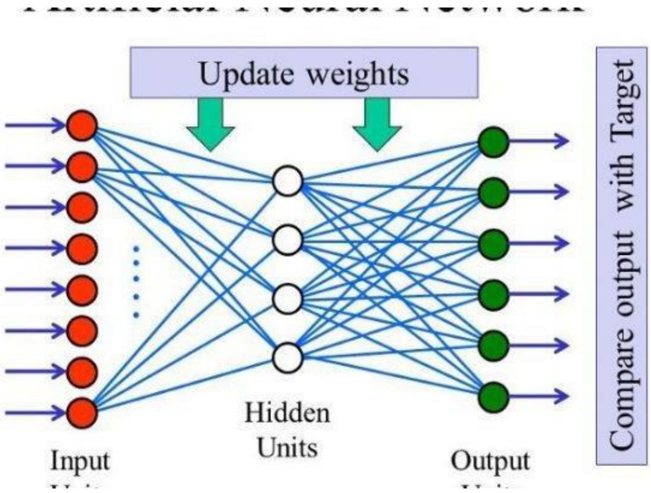
Inference is the term used when a trained NN is making predictions. Many companies use inference at scale so throughput, of course, is important, but also response time is essential. Think of how long your patience lasts when waiting for a response from Siri or Google Assistant when you asked for the nearest Italian restaurant. We will see the demand to reduce response time during inference is a powerful motive in modern NN hardware designs.

Each layer in a NN is a vector of floating point (FP) or integer numbers. Numbers used range from INT4, INT8, FP16, FP32 and FP 64. Larger FP provide greater accuracy, but they make the NN slow and more computationally expensive. Using 8-bits per neuron is becoming more common during certain stages as processing speed is largely reduced due to smaller bit sizes used in computation. Facebook is one example of having done research showing the benefits of using smaller bit size to optimize inference response-time.

The three factors:

1. Deeper Networks
2. Input Size
3. Big Data

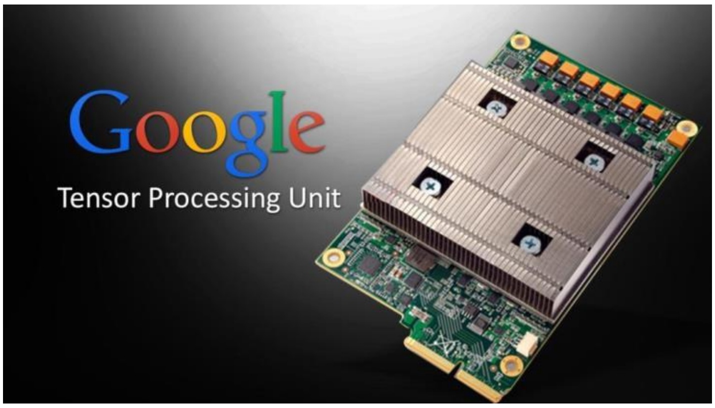
are increasing the amount of compute required for machine learning. Considering this software problem, we can see an opportunity for hardware to step in and increase efficiency.



Hardware of Google Tensor Processing Unit (TPU) for Machine Learning

Google's Tensor Processing Unit (TPU) is an application specific integrated circuit created by Google to process neural networks. Google announced, in 2016, the TPUs have been in use in their data centres for over a year. Google designed the hardware to work with their open-source software Tensorflow, an application specifically built for working with neural networks. One motivation for creating the Application-Specific Integrated Circuit (ASIC) was to support the growing number of speech translations Google continually processes.

The TPU works by creating a grid of simplified ALUs. The data is sent via a PCIe bus to the grid. As the multiplication and addition of each vector is applied to each layer the result is passed to the next layer creating a pipelining effect throughout the 128x128 matrix of ALUs. Memory requirements are low as the output of one layer of ALUs is the input of the next layer. This also reduces power consumption as memory access is more power expensive than ALU computation.



1. *Implementation and Cost*

The TPU is integrated into already running servers via a PCIe bus. This makes implementing it for Google inexpensive and quickly available for rental on the cloud for between $1.35/hr and $5.22/hr! Multiple TPU pods can work together to speed up training models.

The TPU is a simple solution that does not handle “caches, branch prediction, out-of-order execution, multiprocessing, speculative prefetching, address coalescing, multithreading, context switching, and so forth. Minimalism is a virtue of domain-specific processors.” Google wanted a pluggable solution that could be implemented quickly and with little risk/cost.

2. *Tensorflow Software*

Google’s hardware is optimized for Tensorflow. This is an open-sourced library built for building neural networks and was released in 2015.

3. *TPU hardware specifications*

The TPU instructions are sent via PCIe Gen3 x16 bus into an instruction buffer. The TPU does *not* fetch instructions from the CPU, rather the CPU will send instructions to a buffer. This was designed to “keep the matrix unit busy”.

It also reduces the number of pipeline steps required. The architecture uses a 4-stage pipeline, though the grid of ALUs have their own pipelining during matrix computation. The pipelining available thus varies greatly due to various NN instructions that may require 1000s of clock cycles. Twenty-three percent of stalls occur due to read-after-write (RAW) dependencies.

Even though memory is kept low as ALU passes data to contiguous ALU vectors, memory bandwidth is an optimization problem that many NN (LSTM and MLP) bottleneck on. Google’s NN analytics show peak performance of certain models is not reached because of memory constraints. While detrimental to peak performance adding more memory is a known issue that can be solved by simply adding more memory (cheap) or some larger L2/L3 caches.

The MMU contains 256x256 MACs that perform 8-bit to 64-bit multiplication and addition on signed and unsigned integers. It can increase accuracy of computation by increasing to 16-bit or 32-bit calculations at the cost of speed and number of items in the vectors being processed. It can write 256 8-bit values per clock cycle.

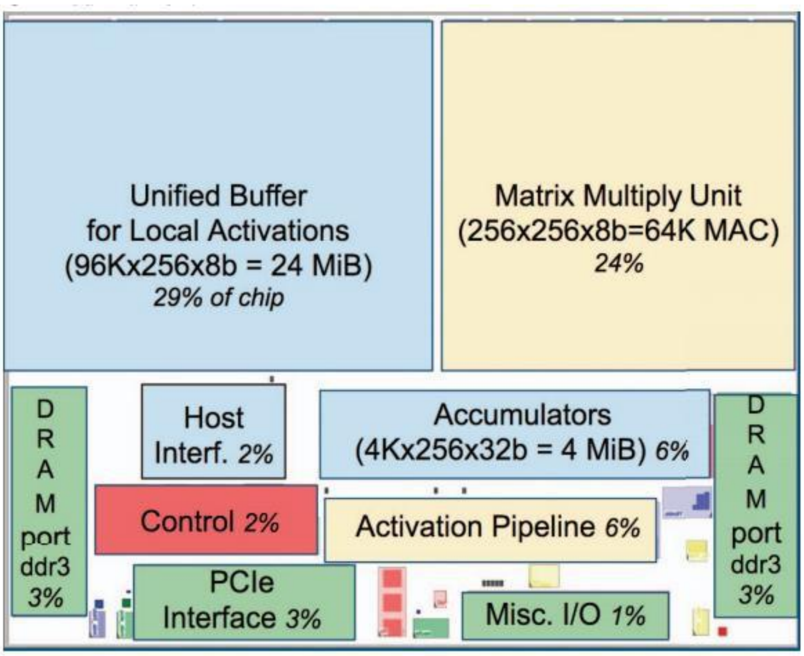


Figure 2. Floorplan of TPU die. The shading follows Figure 1. The light (blue) datapath is 67%, the medium (green) I/O is 10%, and the dark (red) control is just 2% of the die. Control is much larger (and much harder to design) in a CPU or GPU.

4. TFLOPS Performance

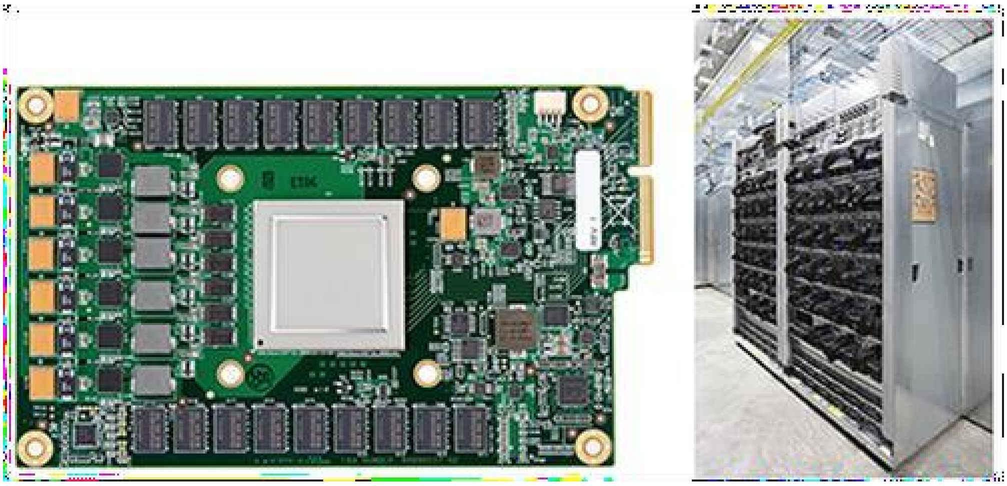
Google claims the TPU is 15–30 times faster at inference than the NVIDIA K80 GPU. The TPU has 25 times as many multiplier accumulators (MACs) and 3.5 times as much on-chip memory as the K80 GPU. They also claim the energy performance is 30–80x more efficient than "contemporary CPUs and GPUs. Google's TPU reaches 180 TFLOPS when combined via four 45 TFLOPS chips.

5. Power

Power optimizations are achieved in part by reducing the reads and writes on the buffer and memory. The TPU idle power required is high compared to GPU and CPU. At 10% capacity the wattage used is 88% the power it uses at 100% capacity.

6. TPU Pods

Each board can be connected together to form "'multi-petaflop' ML supercomputers that [they] call 'TPU Pods''. Google claims machine learning (ML) models can be trained overnight, rather than waiting days or weeks to train a business-critical model. Google claims training ResNet-50[A] and Transformer ML models drop from the about a day to under 30 minutes on a full TPU pod.

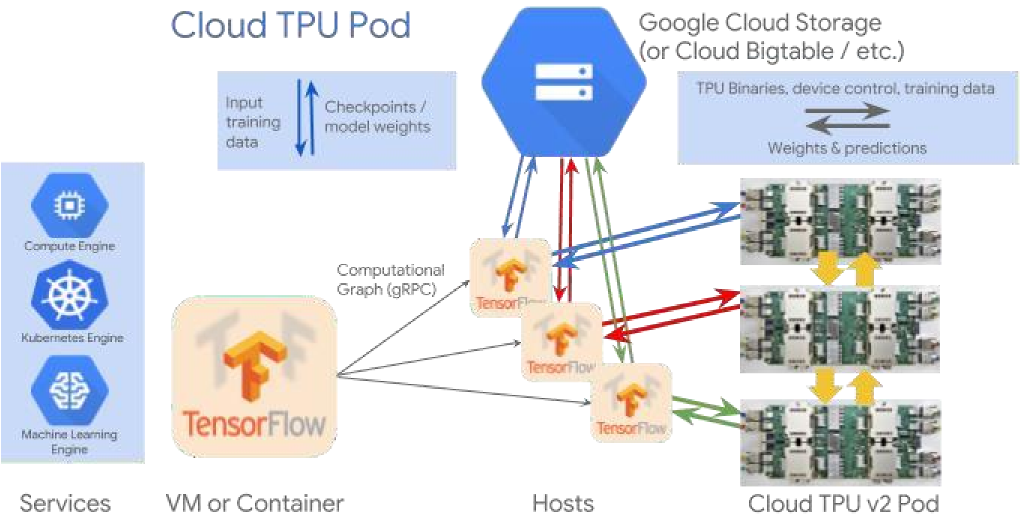


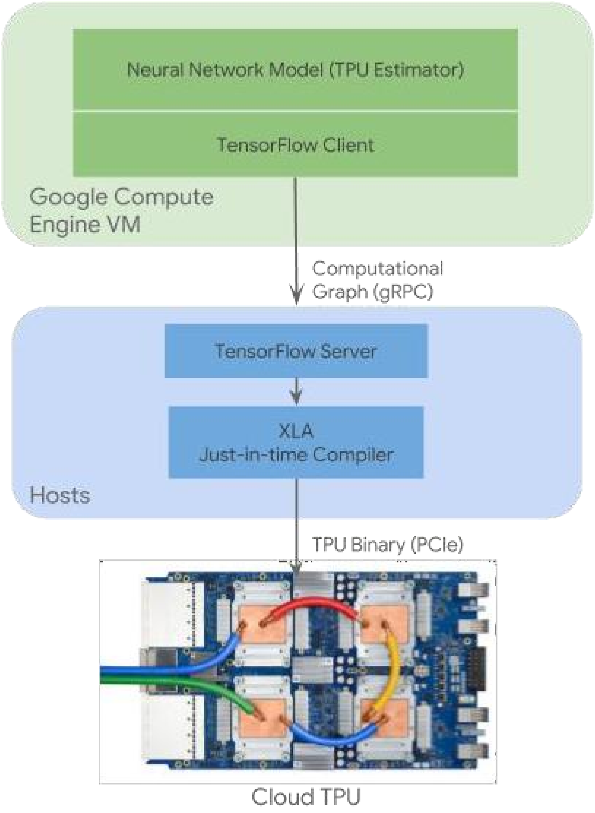
Google's first Tensor Processing Unit (TPU) on a printed circuit board (left);

TPUs deployed in a Google datacenter (right)

By evaluating a custom ASIC---called a Tensor Processing Unit (TPU)--
-deployed in datacentres since 2015 that accelerates the inference

phase of neural networks (NN). The heart of the TPU is a 65,536 8-bit MAC matrix multiply unit that offers a peak throughput of 92 TeraOps/second (TOPS) and a large (28 MiB) software-managed on-chip memory. The TPU's deterministic execution model is a better match to the 99th-percentile response-time requirement of our NN applications than are the time-varying optimizations of CPUs and GPUs (caches, out-of-order execution, multithreading, multiprocessing, prefetching etc.) that help average throughput more than guaranteed latency. The lack of such features helps explain why, despite having myriad MACs and a big memory, the TPU is relatively small and low power. Their workload was written in the high-level TensorFlow framework, uses production NN applications (MLPs, CNNs, and LSTMs) that represent 95% of our datacenters' NN inference demand. Despite low utilization for some applications, the TPU is on average about 15X - 30X faster than its contemporary GPU or CPU, with TOPS/Watt about 30X - 80X higher. Moreover, using the GPU's GDDR5 memory in the TPU would triple achieved TOPS and raise TOPS/Watt to nearly 70X the GPU and 200X the CPU.





Google is offering more powerful hardware for its machine learning cloud customers as it launches its next generation of Tensor Processing Unit (TPU) chips through its Google Compute Engine.

Officially called Cloud TPUs, these chips can work with Intel's Skylake processors, as well as Nvidia's GPUs. The new generation of TPUs offer 180 teraflops of floating-point performance, and Google has designed these chips so that they can be stacked in what the company calls a TPU pod, which houses 64 TPUs and can provide up to 11.5 petaflops of compute power.

Taken together, these TPUs are designed to accelerate machine learning, while giving customers access to the power of the Google's cloud, which can lower the barrier to entry for many businesses trying to build machine learning and artificial intelligence (AI) applications.

- **A TPU pod**

AlphaGo is powered by TPU, which is built on a 28nm process, runs at 700MHz and consumes 40W when running. Because TPU is needed to deploy to Google’s existing servers as fast as possible, the company chose to package the processor as an external accelerator card that fits into a SATA hard disk slot for drop-in installation. The TPU is connected to its host via a PCIe Gen3 x16 bus that provides 12.5GB/s of effective bandwidth. The secret of TPU’s outstanding performance is its dedication to neural network inference. The quantization choices, CISC instruction set, matrix processor and minimal design all became possible when it was decided to focus on neural network inference. Google also announced that its second-generation TPUs were coming to Google Cloud to accelerate a wide range of machine learning workloads, including both training and inference.

Cloud TPU is designed to run cutting-edge machine learning models with AI services on Google Cloud. And its custom high-speed network offers over **100 petaflops** of performance in a single pod — enough computational power to transform your business or create the next research breakthrough.

What Did Google Announce?

Google announced a new ASIC that will accelerate its internal machine learning algorithms, as well as provide a compelling platform for AI practitioners to use the Google Cloud for their research, development, and production AI work. The 2nd generation TPOU chip delivers 45 Trillion Floating Point Operations Per Second (presumably 16-bit TFLOPS) for Machine Learning, roughly twice that which is available today from NVIDIA P100 (20 TFLOPS) or Advanced Micro Devices' upcoming Vega GPU (25 TFLOPS), however it will be surpasses by NVIDIA's new Volta chip described below. The "Cloud TPU" is packaged on a 4-chip module complete with a fabric to interconnect these powerful processors, allowing very high levels of scaling. This scaling capability is important, because training a neural network can take advantage of an almost limitless supply of accelerators.

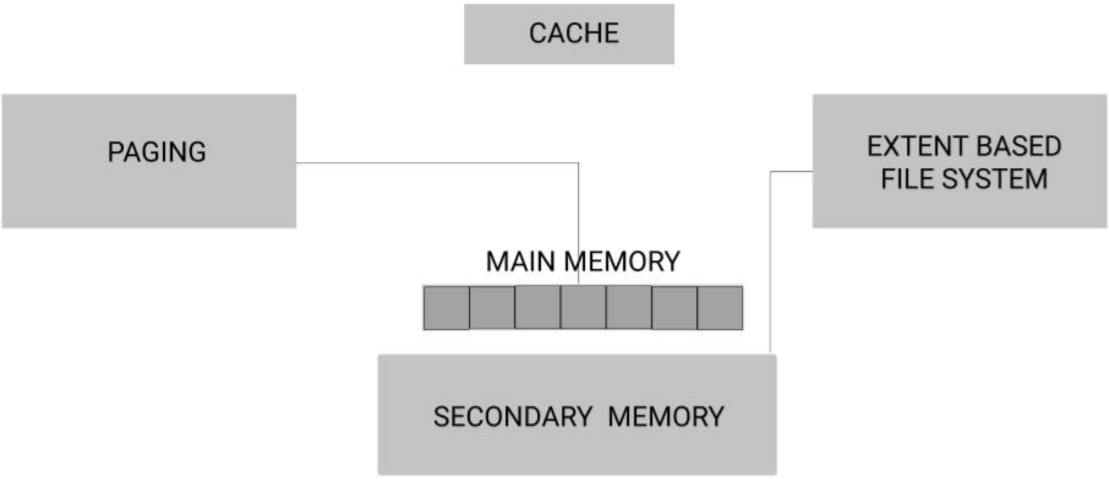
The 4 chip Cloud TPU board forms the building block node for interconnecting 1000s of TPUs in a cluster for research and cloud services. There were no visible signs of active cooling, and the company did not disclose power consumption details. (Source: Google). The 4-chip Cloud TPU, therefore delivers 180 TFLOPS, and were shown in a "TPU Pod" with 32 interconnected boards, delivering 11.5 TeraFlops of peak performance—effectively a large supercomputer in a single rack.

Google also announced the TensorFlow Research Cloud, a 1,000-TPU (4,000 Cloud TPU Chip) supercomputer delivering 180 PetaFlops (one thousand trillion, or one quadrillion, presumably 16-bit FLOPS) of compute power, available free to qualified research teams. While this is similar but significantly larger in concept to the Saturn V Supercomputer from NVIDIA, the Google Supercomputer is designed to support only Google's own open-source TensorFlow Machine Learning framework and ecosystem, while Saturn V is available for all types of software.

Problem and its Solution:

Abstract:

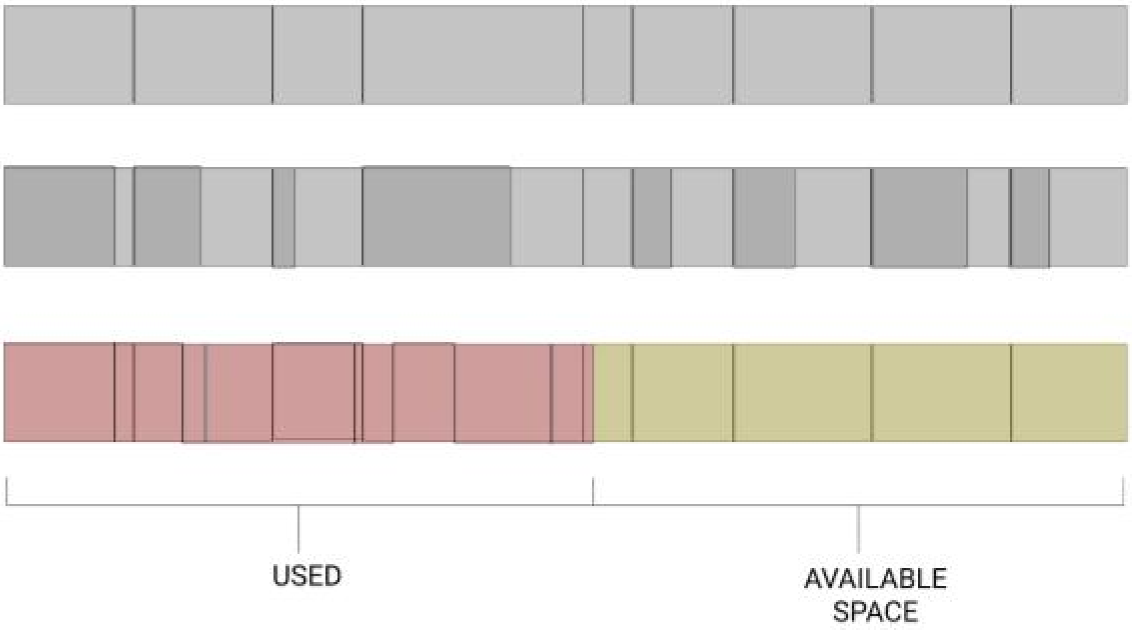
The memory architecture for the System should be immediate and secure. So here I modified allocation of process in paging concept to remove external fragmentation and I used extent based file system to extract the information quickly.



Implementation:

In paging, instead of using FIFO I have modified it to reduce external fragmentation and apparently more number of process can be allocated to the memory
At first will be placing all the process in the main memory through first in first out.
Then we will be shifting all the process to one side and making other side available.
This will reduce the external fragmentation while allocating process in main

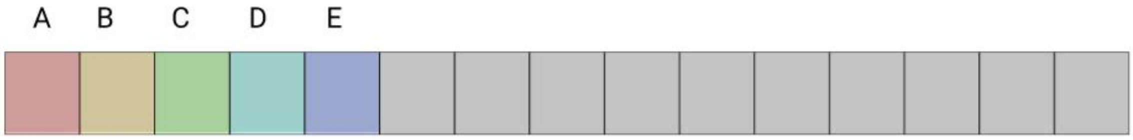
memory.

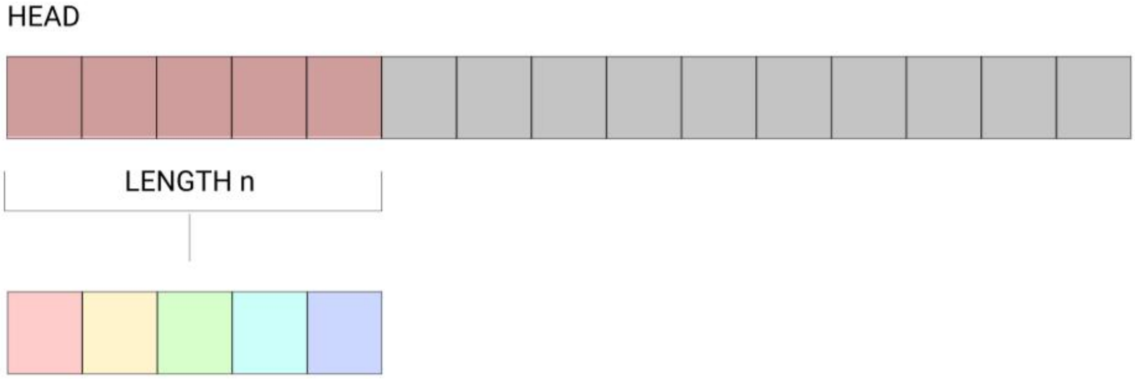


In extent based file system, we will be extracting files as bulk amount Instead of using pointers concept to retrieve files from main memory. It will reduce the time consumption of to retrieve information from main memory.

This can help us saving time when we want to extract continuous data like array.

It will first collect the address of head and then it will retrieve the information of next address by adding file size to it.





Memory management is all about allocation and optimization of finite physical resources.
Our own aim is to reduce the computational time and increasing the available space in main memory to execute any type of process easily.
We can use SSD to boot the applications fast.
We can increase the amount of space for SSD than HDD.
If we make all application programs and some required system applications to be in SSD. It will make system even faster. The more space in SSD the more process executes faster.

Conclusions

Google is attempting to build a dominant position in Artificial Intelligence, from optimized search services, to Android capabilities, to autonomous vehicles. Having complete control of the required technology stack enables it to optimize its technology while lowering its CapEx compared to buying technology from the outside. The synergies of controlling the Cloud TPU and TensorFlow should give the company a strategic competitive advantage, while accelerating the underlying science of AI and the industry at large. The impact on NVIDIA may be real, but relatively contained, at least for now. The longer-term impact on the GPU for AI, which has built NVIDIA’s impressive growth engine, remains to be seen.