Lecture 03 - What is knowing machine architecture good for?

2 Reasons

1. Performance
2. Knowledge over time

Example 1 - NTRU Encryption

Quantum Computers and the “shore” algorithm will break RSA and ECDSA encryption. This is the underlying encryption behind TLS, the transport layer security that enables all of HTTPS.

Quick explanation of how ECDSA works.

Why non-planer encryption is quantum proof.

Google current at 53 bits. Needs to get to 64 to 98 bits to break ECDSA.

NTRU Encryption entered into the NIS competition 4 years ago. NTRU Is a replacement for all of ECDSA that is quantum proof. It is also faster than ECDSA and AES. One of the finalists for the new encryption standard.

Port to multiple architectures.

Example 2 - Faster database

Data growth and IoT is driving this. The world is creating 2.5 Quintilian bytes of data a data. 90% of the data in the world has been created in the past 2 years. This is just the tip of the iceberg. For example Tesla cars have 2.1 billion miles of driving under Auto-Pilot - and they generate 2.8 terabytes of data a day that Tesla would just love to have. Right now they are pulling about 25 megabytes of this back per week. 5G really won’t make much difference in phone calls. It will provide the thru-put necessary for Tesla to get the terabytes of data back from every car.

About $^1/_2$ of time in a database management system (DBMS) is spent sorting data. One way to sort faster is to use faster hardware for the sorting. The other $^1/_2$ is on searching. If you can get more CPU/Cores to do the searching in parallel you can speed it up.

Database is about lots of data - the other slow portion of database is the disks. Replace the disks with memory (SSD) or store the data in main memory (RAM). Redis is an example of a “memory” database - but might be better called a shared / persistent data structure.

What about passing the sorting to the GPU in a dedicated box.

CPU v.s. GPU based computation

Latest AMD system with 64 cores.

Nvidia Kinetica CPU based database: 5100+ cores for GPU cards.

Kinetica also runs on DGX-2 Nvidia - GPUs on non-graphics boards in 4 redundant systems: 81,920 cores.

MapD: GPU-powered SQL data platform uses in-memory storage and also leverages modern SSDs for persistent storage, It boasts microsecond query processing performance in the billions of rows.

Example 3 - Turning FedEx Packages Tracking

FedEx tracks packages - When they put in the system they had “performance” problems with Oracle database and a bunch of very expensive people from Oracle working on it. They hired me to be an outside review of they performance tuning from Oracle.

My conclusion: Given the requirements that the Oracle folks were never going to get the database to perform as needed. They needed a different solution - to split the load into multiple chunks. We changed the hash code and used a computer as a “switch” to direct traffic.

Example 4 - Building / Porting an OS

New architectures will require ports of an operating system. The Unix/Posix system has dominated the world. VxWorks is the 2nd most common system. Both require a “port” for new hardware. The Thrid most common system in the world is Minix - it is an open source Unix-ish system found in the Intel ME engine.

Another alternative is to build a system from the ground up. I have done this once.

Example 5 - Merkle Hash

One of the biggest computer platforms in existence is Bitcoin. Over 12,000 “nodes” running a very simple hashing algorithm. (Ethereum is 2nd largest with 11,000 “nodes”)

Mining Algorithm: Use an infinite loop to:

The Proof-of-Work mining process is:

1. Serialize the data from the block for hashing.
2. Calculate the hash of the data. This is the slow part. In BTC it is done in an ASIC, in Ethereum it is done on GPU. A GPU can run 4 billion hashes a second?
3. See if the first 4 characters of the hash are 0's. - if so we have met the work criteria.
    - Set the block's "Seal" to the hash
    - return
4. Increment the Nonce in the block, and...
5. Back to the top of the loop for another try at finding a seal for this block.

Example 6 - System Vitalization

OS 360 running as a virtual system on OS 370. The post office.

VMWare brought this to the land of PCs.

Docker move this into a "container" system (Not a new idea by the way - BSD "jails" had been doing this for over a decade).

Can you write software that will "mimic" an entire hardware system?

Alternative Architectures

Stack architectures

Ethereum runs on a "stack" system. The "fourth" programming language.

VILW - very long instruction words

Run in parallel multiple parts of a computation by building bigger instructions.

Computation in Memory

Move the "calculation" from the CPU to memory.

Copyright