CMP SCI 635: Modern Computer Architecture

Understanding and Designing New Server Architectures for Emerging Warehouse-Computing Environments

Name: Kunjal Panchal Date: 5th Dec, 2019 Student ID: 32126469

Paper: 23 – Data Centers [Lim08]

Strengths:

1. To overcome the challenges in scaling the physical capacity of datacentres, the efforts in this paper optimized and enabled the memory disaggregation. Microsoft has complemented this work by exploring a knob for dynamic provisioning of memory and storage resources within a server using polymorphic emerging memory technologies, taking one step further in solving the datacentre resource provisioning problem.

2. Although the 3D-stacked DRAM has various advantages, its power consumption is rapidly growing with the capacity increase. DRAM memory consumes 20% to 40% of the overall system power. To tackle the DRAM refresh power overhead associated with the larger size DRAM, phase change memory (PCM) can be explored as an alternative to the DRAM.

Weaknesses:

1. Bottlenecks in the memory system prevent server processors from getting high performance on server applications. As server workloads operate on a large volume of data, they produce active memory working sets that dwarf the capacity-limited on-chip caches of server processors and reside in the off-chip memory; hence, these applications frequently miss the data in the on-chip caches and access the long-latency memory to retrieve it.

2. This paper has identified significant over-provisioning in conventional server processors in the context of datacentres and warehouse computing, both in core capabilities and in the memory hierarchy. It focused on request-oriented workloads such as web search. For this type of server workloads, they find that large instruction footprints limit system performance, which does not happen in High Performance Computing.

Questions/Assertions:

1. This paper tackles the characterizing resource consumption and performance breakdowns inside modern cloud data centres. This is conducted at larger scale and have made valuable data publicly available to the research community. As microservices have gained interest recently, what are the architectural implications of Function-as-a-Service applications and platforms?

2. Are there any resources that talk about moving from a mostly modelled warehouse environment to more of a raw unstructured environment like a data lake? For instance, what data is best left in former, and what other data is better served in later?

3. Generally, do we use a model which puts our data in a data lake, allowing various tools to dip into it, and write back results? Or do we have the data locked down in a data warehouse instead, and either request extracts from that to allow you to process the data (particularly if you want to use Python or R, rather than just SQL), or possibly push the data processing into the warehouse itself should it have that capability?

4. Data warehouse architecture has seen a huge shift towards cloud-based warehouses and away from traditional on-site warehouses. Chief reasons being scale, cost and performance.

5. Data warehouse appliance should be able to run most queries very fast. This is what many organizations experience when they migrate their data warehouses, implemented with a classic database server that they had been optimizing for years, to a data warehouse appliance. Most of them experienced a performance improvement without any tuning and optimization. What's the advantage? Less time (maybe no time at all) has to be spent on tuning and optimization of the server, the database server, and the database design. The value of this should not be underestimated. A classic database server requires a lot of optimization and tuning before an acceptable performance is achieved.