New Hardware to Increase Server Memory Capacity for In-Memory Computing and Big Data
Memcon 2015

Marc Greenberg, Director, Product Marketing, Synopsys
October 13, 2015
What is Big Data?

• The best description I’ve seen:
  – Nobody knows what it is
  – Everyone is talking about it
  – Therefore, everyone says they are doing it

• Let’s use:
  A database bigger than the one you’re using today
What is In-Memory Computing?

• Traditional compute architectures store the majority of their data in HDD or SSD
  – HDD/SSD has much larger capacity than memory
  – HDD/SSD has much lower cost than memory

• Sounds like the HDD or SSD is a “no brainer.” What’s the catch?
  – Memory has much better IOPs & Bandwidth
  – Memory has much better latency

• In-memory computing stores data in memory local to the CPU instead of in HDD/SSD
So What’s the Killer App?

• Applications where **IOPs** and **latency** are more important than **cost** or **capacity**
  …which means…

• Applications where getting the answer faster is worth money
  – Retail
  – Financial analysis
  – Situational awareness / Security
  – Business intelligence
  – Trend analysis
  – Machine learning
Most Databases are Not in Memory Today

Why not?

• Most databases that run in real-time are fairly simple and use indexing and analytics run over many hours to give real-time results.
Example: Transportation Company

• Example to show the benefit of Big Data and in-memory computing
• Consider the evolution of information in a transportation company
• Goal: Maximize revenue from the vehicles in their fleet
1970 – Bus

- No reservation system
  - Effectively everyone is standby
- Flat Fare
- One Agent (bus driver), one table
- Analytics: Count money in fare box

Query: Is the bus here?

Database: Fare

Analytics: Daily Fare Totals
Mid-70’s – One Aircraft

• Limited seats – everyone standby doesn’t work
  – Must allow for advance purchase
• Flat fare
• Ticket-based reservation system
  – Fixed number of available tickets
  – Present physical ticket at gate
    – Ticket is like cash
  – If flight is sold out, too bad
• One agent, few tables

Query  Agent  Database  Analytics

Do you have a seat on Oct 15th?
1980’s – Several aircraft

- Computer reservation system
- Prices decided long in advance
- Fare classes to try to match passengers to fares
- Ticket-based reservation system within each fare class
  - Tracking availability within each fare class for each flight
- Multiple agent access to database

Query: How much is a seat on Oct 15th?

Agent: Oct 14 - 4 available
  $125 with restrictions
  Oct 15 – 2 available
  $150 no restrictions

Analytics: Daily Unsold Tickets By class

© 2015 Synopsys, Inc.
2000’s – Many aircraft

- Reservation system
- Seasonal adjustments to pricing
- Weekly load based pricing adjustments
- Still fare class based, fewer restrictions
- Internet booking

Adjust fares based on availability and history

Query: How much is a seat on Oct 15th?

Agent:
- Oct 14 - 4 available
- $125 with restrictions
- Oct 15 – 2 available
- $150 no restrictions

Database:
- Current availability
- Past history

Analytics:
2010’s – The start of big data

- End of fare classes & round trips
- Demand based pricing
- Predictive pricing
- Web agent access to database
- Macro-level demand generation

Promotion
Deep search
Balance load

Analytics

Database

Adjust fares based on availability and history and forward prediction

- Current availability
- Past history
- Prediction

Query

Agent

Oct 14 - 4 available
$125 with restrictions
Oct 15 – 2 available
$150 no restrictions

Do you want to fly on the 14th instead?

How much is a seat on Oct 15th?

How about a weekend in Des Moines?
2020’s – In-memory computing

- Faster access to data enables more search in less time
- Intelligent personalized pricing drives
- Must be kept in memory to produce a result from diverse data sources in milliseconds

In-Memory Database

- Availability
- All customer history
- Predicted demand
- Analytics

This Customer History:
- Browser History
- Purchases
- Preferences
- Frequent Flyer Status
- Ability to pay
- Travel Companions
2020’s – In-memory computing

How much is a seat on Oct 15th?

$150, or, if you buy now and travel on the 14th you can travel for $125 and we’ll give you a voucher for $50 off a flight to Chicago to use by Nov 15th

In-Memory Database

Competitor pricing is still at $150 for Oct 15th
Predict we can sell seats on Oct 15th for $200 later
User has never paid more than $150 for this flight
Might not sell out on Oct 14th
→ Push Oct 14th flight
User has 3 social media friends in Chicago
Took weekend trip to Chicago 1 year ago
Very high availability on weekends to Chicago
→ Make special offer on trip to Chicago
User sometimes travels this route with Betty
Betty is a frequent flyer with free upgrades
Betty hasn’t searched for any flights today
→ Send Betty a special offer for flight on Oct 14th with free companion upgrade
How do we enable in-memory computing?

• Move data from HDD/SSD on to the memory bus
• Debatable: move data from Flash into DRAM
• Modify the application to take advantage of latency and IOPs

• Multiple possible techniques used individually or together:
  – Add memory channels
  – Add DIMMs
  – Memory Buffers and Data Buffers (RDIMM & LRDIMM)
  – 3D Stacking of DRAM
  – Larger DRAM dies
  – Serial Bus Extenders
  – NV-DIMM and future storage-class memory
HDD/SSD vs In-Memory Database

**Application**
- Random Read

**Kernel**
- File System
- Driver

**Hardware**
- PCIe/SATA Controller
- Disk Controller
- Flash Controller

**Storage**
- HDD
- SSD
- NVDIMM-F
- DRAM

Latency:
- HDD: ~10,000,000ns
- SSD: ~1,000,000ns
- NVDIMM-F: ~90,000ns
- DRAM: ~10ns

Capacity:
- HDD: TB’s
- SSD: 100’s GB
- NVDIMM-F: 100’s GB
- DRAM: 10’s GB
In-Memory Databases Require Structural Changes

• New or modified databases emerge to take advantage of in-memory
  – Apache Spark:
    – “Run programs up to 100x faster than Hadoop MapReduce in memory”
  – IBM DB2 with BLU:
    – “35x to 73x faster analytics, with some queries running more than 1400x faster”
  – Microsoft In-Memory Option:
    – “SQL Server’s in-memory technology enables you to experience up to 30x improvement in transactional performance”
  – Oracle In-Memory option:
    – “…even greater performance is possible when applications take full advantage of the potential of IMDB [In-Memory Database] technology.”
  – SAP Hana:
    – “SAP HANA lets you take in massive amounts of data and lets business users get vital information out -- and act on it -- in real time.”

Sources: Manufacturers’ own websites
Adding More DRAM – Existing Techniques

• Add more DRAM channels
  – Also adds DRAM bandwidth
  – Current servers already at 4 channels per CPU
    – Roadmap to 6 channels
  – Limited by PCB area
  – Limited by CPU ballout
  – Limited by silicon area & beachfront

• 4-channel Server CPU example:
  – 1804 pin package
  – 639 signal & VREF pins associated with DDR
Adding More DRAM – Existing Techniques

Unbuffered DIMM

- Add more DRAM Ranks with unbuffered DIMM
  - Does not add DRAM bandwidth
  - Number of ranks limited by load on Address and DQ (data) bus
  - Adding DIMMs reduces bus speed
    - Typical DDR4 system limited to 2 slots, 2 ranks/slot at DDR4-2133

Typical max system: 32GB/channel using 2 Dual-Rank uDIMM of 8Gb X8 DDR4 Devices at DDR4-2133
Adding More DRAM – Existing Techniques

Registered DIMM

- Add more DRAM Ranks with Registered DIMM
  - Does not add DRAM bandwidth
  - Number of ranks limited by load on DQ (data) bus
    - Address bus buffered on each DIMM – can use X4 DRAM for 2x density per rank
    - Requires one RDIMM memory buffer chip per DIMM
  - Adding DIMMs reduces bus speed
    - Typical DDR4 system limited to 3 slots, 2 ranks/slot, 1DIMM at DDR4-2133
    - Typical DDR4 system limited to 3 slots, 2 ranks/slot, 3DIMM at DDR4-1600

Typical max system: 96GB/channel using 3 Dual-Rank RDIMM of 8Gb X4 DDR4 Devices at DDR4-1600
Adding More DRAM – Existing Techniques

Load Reduced DIMM

• Add more DRAM Ranks with Load Reduced DIMM
  – Does not add DRAM bandwidth
  – Number of ranks limited by load on DQ (data) bus
    – Address and data buses buffered on each DIMM
    – 10-chip LRDIMM chipset per DIMM adds cost & power
  – Adding DIMMs reduces bus speed
    – Typical DDR4 system limited to 3 slots, 4 ranks/slot, 2DIMM at DDR4-2133
    – Typical DDR4 system limited to 3 slots, 4 ranks/slot, 3DIMM at DDR4-1600

Typical max system: 192GB/channel using 3 Quad-Rank LRDIMM of 8Gb X4 DDR4 Devices at DDR4-1600
New Techniques for adding DRAM: 3DS

• DDR DRAM dies that are 3D stacked (TSV)
  – Master die (at bottom) controls 2-8 dies
  – Adds Chip ID signals (CID) to identify which DRAM in the stack is being addressed
• Benefits
  – 2x-8x capacity of single dies
  – Presents a single load to the CPU PHY
  – Potential small improvement in inter-die timing compared to inter-rank timing
  – Potentially less power than L/RDIMM
• Alternatives
  – Dual Die Package (DDP)
Larger DRAM Dies

• Present DDR4 dies are mostly 4Gb and 8Gb
  – 16Gb parts in product catalogs are often two 8Gb parts in dual-die packages
• Significant manufacturing challenges in creating 16Gb dies in sub-20nm technology:
  – Write Recovery time (tWR) increase
  – Variable Retention time (VRT) effects
  – Refresh Cycle Time (tRFC) increase
    – Already at 350ns for 8Gb DDR4 device
• Future solution to address these problems under discussion at JEDEC
Serial Bus Extenders

- Proprietary methods of adding more memory by aggregating many DIMMs on to a high-speed serial channel
Non-Volatile Memory

• One problem with storing database in DRAM: What happens if the power goes away?
• Energy-backed DDR DIMM and suspend to disk
  – Takes up room in chassis
  – Limited storage time on supercaps
• Non-Volatile DIMM (NVDIMM)
  – NVDIMM–N – replacement for energy-backed DIMM – suspend to flash
  – NVDIMM-F – Behaves like a mounted drive – Flash storage on the memory bus but technically not in memory. CPU interacts with RAM buffer. Block access
  – NVDIMM–P – hybrid may use DRAM or other means with Flash to give DRAM-like latency with NAND-like capacity
  – NVDIMM - ? - New storage-class memory devices with DRAM-like latency (~order of magnitude) and Flash-like capacity (~order of magnitude)
### Using NV-DIMM to increase capacity

<table>
<thead>
<tr>
<th>Homogeneous system of NVDIMM-N</th>
<th>CPU</th>
<th>NVDIMM-N</th>
<th>NVDIMM-N</th>
<th>NVDIMM-N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Similar latency and performance as RDIMM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple power driver</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does not increase memory capacity vs RDIMM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Heterogeneous system of RDIMM and NVDIMM-F</th>
<th>CPU</th>
<th>RDIMM</th>
<th>RDIMM</th>
<th>NVDIMM-F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accesses to Flash Rank incur flash latency</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requires flash driver for flash rank</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Increases memory capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Homogeneous system of NVDIMM-P or -?</th>
<th>CPU</th>
<th>NVDIMM-P</th>
<th>NVDIMM-P</th>
<th>NVDIMM-P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latency dependent on partitioning of data</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requires flash driver</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Increases memory capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
An example map through the maze

Each arrow adds capacity. Arrow labels are subjective.
Conclusions

• In-memory computing is valuable when getting the answer faster is worth money
• In-memory computing requires large amounts of memory
• Many methods exist for adding memory to the system
  – Add Memory channels, DIMMs, Memory Buffers and Data Buffers (RDIMM & LRDIMM), 3D Stacking of DRAM, Larger DRAM dies, Serial Bus Extenders, NV-DIMM and future storage-class memory

• Synopsys offers a range of solutions for Enterprise-class memory:
  – DDR Controller IP, DDR PHY IP up to 3200MT/s, Verification IP, Architectural models, Prototyping kits, and services for signal integrity and subsystem creation