Memory Performance Guidelines
for Dell PowerEdge 12th Generation Servers

Configure your server environment for optimal memory performance

John Beckett
Solutions Performance Analysis
Enterprise Solutions Group
Memory Performance Guidelines for Dell PowerEdge 12th Generation Servers

Contents

Executive summary................................................................................................................................................... 5
Introduction ............................................................................................................................................................... 5
PowerEdge 12th generation memory overview ................................................................................................... 5
  Test methodology ............................................................................................................................................... 10
  Local versus remote memory bandwidth and latency ..................................................................................12
BIOS memory mode selection guidelines for performance ............................................................................ 15
  Advanced ECC compared to Optimizer mode .............................................................................................. 16
  Node Interleave effects ....................................................................................................................................... 17
Memory type comparison ..................................................................................................................................... 19
  RDIMMs versus UDIMMs .................................................................................................................................... 19
  Power comparison: 1.35V RDIMMS versus UDIMMs .................................................................................... 20
  2 DPC RDIMMs (1600MT/s) versus 2 DPC LRDIMMs (1333MT/s) ............................................................... 21
  Quad rank RDIMMs versus LRDIMMs (2 DPC) comparison ......................................................................... 24
  RDIMM voltage effects on system AC power consumption ....................................................................... 27
Memory frequency .................................................................................................................................................. 28
  Memory frequency and virtualization workloads .......................................................................................... 32
  Memory frequency effects on power .............................................................................................................. 33
Balanced versus unbalanced................................................................................................................................. 34
Memory channel population ................................................................................................................................. 37
  UDIMM rank and channel population comparison ..................................................................................... 41
  RDIMM rank and population per channel guidelines ................................................................................. 42
  RDIMM rank and channel population ............................................................................................................ 43
Conclusion ............................................................................................................................................................... 45
References................................................................................................................................................................ 47

Tables

Table 1. DDR3 DIMM types and population per channel ............................................................................. 7
Table 2. PowerEdge memory speeds by type and loading ............................................................................ 7
Table 3. Intel Xeon E5-2600 processors ........................................................................................................ 9

Figures

Figure 1. PowerEdge R720 block diagram with additional memory subsystem details .............................. 8
Figure 2. 2-Socket NUMA node architecture .............................................................................................. 12
Memory Performance Guidelines for Dell PowerEdge 12th Generation Servers

Figure 3. NUMA locality relative to individual cores.................................................................12
Figure 4. Local vs remote NUMA node memory bandwidth (higher is better).........................13
Figure 5. Local vs remote NUMA node memory latency (lower is better)...........................14
Figure 6. Thread count impact on memory bandwidth measurements.................................15
Figure 7. Memory latency impacts of optimizer vs Advanced ECC......................................16
Figure 8. Advanced ECC performance effects compared with memory Optimizer mode.........17
Figure 9. Node Interleave disabled vs enabled.........................................................................18
Figure 10. 1 DPC RDIMM vs 1 DPC UDIMM DR 1333MT/s latency comparison....................19
Figure 11. 1 DPC RDIMM vs 1 DPC UDIMM DR 1333MT/s benchmark/memory bandwidth...20
Figure 12. DDR3L RDIMM vs DDR3L UDIMM system AC power comparison.......................21
Figure 13. 2 DPC dual rank RDIMMs vs 2 DPC LRDIMMs latency comparison.........................22
Figure 14. 2 DPC dual rank RDIMMs vs 2 DPC LRDIMMs integer workload comparison........23
Figure 15. 2 DPC dual rank RDIMMs vs 2 DPC LRDIMMs floating point workload comparison...24
Figure 16. 2 DPC quad rank RDIMMs vs 2 DPC LRDIMMs latency comparison.......................25
Figure 17. 2 DPC quad rank RDIMMs vs 2 DPC LRDIMMs integer workload comparison........26
Figure 18. 2 DPC quad rank RDIMMs vs 2 DPC LRDIMMs floating point workload comparison...27
Figure 19. 1.5V vs. 1.35V RDIMMs (8 x 4GB DR 1 DPC) System AC power comparison..........28
Figure 20. Effects of memory frequency on overall memory bandwidth...............................29
Figure 21. Effects of memory frequency on memory latency (lower is better).........................29
Figure 22. Memory frequency impacts on integer server workloads........................................30
Figure 23. Memory frequency impacts on floating point workloads.......................................31
Figure 24. Memory frequency on the Linpack (HPC) benchmark..........................................32
Figure 25. Memory frequency on virtualization performance using VMmark2.........................33
Figure 26. Memory frequency impacts on overall system AC power.....................................34
Figure 27. Balanced memory configuration (1 DPC, all 4 channels populated).........................35
Figure 28. Unbalanced memory configuration (mixed 1 DPC and 2 DPC).................................35
Figure 29. Balanced vs unbalanced memory configurations and effects on integer workloads....36
Figure 30. Balanced vs unbalanced memory configuration effects on floating point workloads...37
Figure 31. Memory channel population effects on overall memory bandwidth.........................38
Figure 32. Memory channel population effects on integer server workloads..........................39
Figure 33. Memory channel population effects on floating point server workloads..................40
Figure 34. UDIMM memory rank and channel population comparison.................................41
Figure 35. UDIMM channel population effects on server benchmarks and memory bandwidth..42
Figure 36. DIMM per channel population effects on memory latency with RDIMMs...............43
Figure 37. Memory rank and channel population comparison...............................................44
Figure 38. 2 DPC vs. 3 DPC RDIMM latency comparison at a common 1333MT/s...................45
This document is for informational purposes only and may contain typographical errors and technical inaccuracies. The content is provided as is, without express or implied warranties of any kind.

© 2012 Dell Inc. All rights reserved. Dell and its affiliates cannot be responsible for errors or omissions in typography or photography. Dell, the Dell logo, and PowerEdge are trademarks of Dell Inc. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell disclaims proprietary interest in the marks and names of others.

July 2012
Executive summary

Dell PowerEdge 12th generation servers offer unparalleled performance and flexibility of memory configuration. Due to changes in system architecture, memory types offered, and custom configuration options, an understanding of the updated features will help ensure the highest performance possible. The type of memory selected and how the memory is populated in a system will have a major performance impact on a system. Certain BIOS options can also impact the memory performance depending on your system’s operating system environment.

A solid understanding of all the variables and the fundamental elements will ensure optimal memory subsystem performance. This white paper measures the impact of these variables primarily from a performance standpoint with some additional power guidelines, and provides recommendations for optimal memory subsystem layout and performance. To help explain the guidelines and recommendations, this white paper lists various comparisons made between memory types, memory speeds, memory layout options, and two crucial memory-oriented BIOS settings.

Introduction

In order to achieve maximum performance with Dell PowerEdge 12th generation servers using the Intel Xeon processor E5-2600 product family, it is important to understand the impact of memory hardware configuration and relevant BIOS options. Although many general concepts regarding the memory subsystem remain similar to the PowerEdge 11th generation servers using the Intel Xeon processor 5500 and 5600 series, it is important to understand some key differences and crucial concepts in order to achieve optimum memory subsystem performance.

This paper provides guidance on selecting the optimal memory configuration for best performance for a variety of server workloads. The memory subsystem is discussed in detail, and the concepts of local versus remote Non-Uniform Memory Access (NUMA) nodes are described and quantified. The three DDR3 DIMM types available for PowerEdge 12th generation servers are compared to each other. Additionally, two critical BIOS options are also explored: Memory Operating Mode and Node Interleave. Memory frequency and channel population impacts on a variety of industry standard benchmarks are examined. Balanced and unbalanced memory configurations are described and compared, as well as the effect of varying DIMM ranks and population of memory channels.

All measurements noted in this white paper were taken on the Dell PowerEdge R720, but the effects described are identical across PowerEdge x620 and x720 servers (R720xd, R620, M620, and T620). For other PowerEdge 12th generation servers, such as those that use the Intel Xeon processors E5-2400 and E5-4600 product families, there are similarities in terms of memory architecture, but this white paper only addresses the Intel Xeon processor E5-2600 product family.

PowerEdge 12th generation memory overview

The Intel Xeon processor E5-2600 product family shares some similarities to the previous-generation Intel Xeon processor 5500 and 5600 series, in that the memory controller is integrated into the processor package rather than being on a separate Northbridge. However, the memory controller now can operate at the same maximum non-Turbo processor frequency as the cores, rather than running at a potentially lower frequency. Another new feature of the E5-2600 processors is that the number of
Memory Performance Guidelines for Dell PowerEdge 12th Generation Servers

DDR3 memory controllers and channels per processor has increased from 3 to 4, and the maximum memory frequency has increased to 1600MT/s from 1333MT/s. The number of inter-processor QuickPath Interconnect (QPI) links has doubled from 1 to 2, and the links themselves are faster at up to 8.0GT/s compared to a previous maximum of 6.4GT/s. Another significant architectural change is the integration of PCIe 3.0 controllers into the processors themselves; this means that I/O traffic no longer has to go to an external I/O Hub (IOH) to reach the PCIe slots, and now all slots are “local” to one or another of the two processor sockets.

The PowerEdge x620 and x720 servers that support the Intel Xeon E5-2600 processors support three separate dual inline memory module (DIMM) types: registered DIMM (RDIMM), unbuffered DIMM (UDIMM), and load reduced DIMM (LRDIMM). These DIMM types cannot be mixed in a single system; you must select the type of DIMM appropriate for your workload at the time of purchase.

- **RDIMMs** have their address, control, and clock lines buffered in a register, but data signals are not buffered. RDIMMs are the most commonly used DIMM type, and offer the best mix of frequency, capacity, and rank structure choices. Single rank (SR) and dual rank (DR) RDIMMs have a maximum frequency of 1600MT/s, and a maximum capacity per DIMM of 16GB. 32GB RDIMMs are also available, but only in quad rank (QR), which limits the number populated per memory channel to 2, and limits the frequency to 1066MT/s. The current maximum memory configuration using RDIMMs is 512GB.

- **UDIMMs** do not buffer address, control, clock, and data signals. These ECC UDIMMs can offer some minor latency and power advantages in limited cases, but they are currently limited to 1333MT/s, 1600MT/s, and 4GB capacities. UDIMMs are limited to 2 DIMMs per channel, and a current maximum memory capacity of 64GB.

- **LRDIMMs** have the address, control, clock, and data signals buffered. LRDIMMs offer the largest capacities (32GB per DIMM), but are only available at 1333MT/s and quad rank. In 3 DIMM per channel (DPC) configurations, the frequency is limited to 1066MT/s. Using 32GB LRDIMMs at 3 DIMMs per channel gives a maximum memory capacity of 768GB.

This paper compares and contrasts all three memory types, but the primary focus is on RDIMMs since they are overwhelmingly the most common choice for customers.
Table 1 shows the relationship of DDR3 DIMMs to the limits imposed on population per channel, which is important to understand when calculating total memory capacity.

### Table 1. DDR3 DIMM types and population per channel

<table>
<thead>
<tr>
<th>DIMM type</th>
<th>Ranks</th>
<th>Maximum DIMMs per channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDIMM (single rank or dual rank)</td>
<td>1 or 2</td>
<td>3</td>
</tr>
<tr>
<td>RDIMM (quad rank)</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>UDIMM</td>
<td>1 or 2</td>
<td>2</td>
</tr>
<tr>
<td>LRDIMM</td>
<td>up to 4</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 2 lists the memory speeds by type and by loading.

### Table 2. PowerEdge memory speeds by type and loading

<table>
<thead>
<tr>
<th>DIMM Type</th>
<th>DIMM Ranking</th>
<th>DIMM Rated Voltage, Speed</th>
<th>12G Memory POR (Socket R, 3DPC Platforms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 DPC</td>
</tr>
<tr>
<td>RDIMM</td>
<td>SR / DR</td>
<td>DDR3 (1.5V), 1600</td>
<td>1600</td>
</tr>
<tr>
<td>RDIMM</td>
<td>SR / DR</td>
<td>DDR3L (1.35V/1.5V), 1333</td>
<td>1333</td>
</tr>
<tr>
<td>UDIMM</td>
<td>SR / DR</td>
<td>DDR3L (1.35V/1.5V), 1333</td>
<td>1333</td>
</tr>
<tr>
<td>RDIMM</td>
<td>QR</td>
<td>DDR3L (1.35V/1.5V), 1066/1333</td>
<td>1066</td>
</tr>
<tr>
<td>LRDIMM</td>
<td>QR</td>
<td>DDR3L (1.35V/1.5V), 1333</td>
<td>1333</td>
</tr>
</tbody>
</table>

* Green boxes are default settings.
All Dell PowerEdge servers that use the Intel Xeon E5-2600 processors support up to 3 DIMMs per channel. This differs from some of the previous generation of 2-socket Intel Xeon-based servers that were limited to 2 DIMMs per channel. Populating different numbers of DIMMs per channel can affect the frequency of the memory and cause secondary latency impacts. As mentioned before, the number of memory channels per processor has increased from 3 to 4. The number of channels populated with memory can drastically alter the overall memory bandwidth of a system, and may negatively impact workloads if improperly configured.

Figure 1 shows the block diagram of the R720. Some details of the PCIe interconnects will differ for other Dell systems that support Intel Xeon E5-2600. This figure illustrates the number of memory channels per processor, as well as the number of DIMMs per channel and the dual inter-processor QPI links.

**Figure 1.** PowerEdge R720 block diagram with additional memory subsystem details
The processor model chosen from the Intel Xeon processor E5-2600 product family can have an impact on the memory subsystem since the processor determines the maximum memory frequency. The advanced and frequency optimized processors, and one of the low-power processors support a maximum memory frequency of 1600MT/s. All of the standard and one of the low-power processors support a maximum memory frequency of 1333MT/s. The basic processors, with lower frequency and fewer cores, support a maximum memory frequency of 1066MT/s.

Memory bandwidth is also influenced by choice of processor model. Memory bandwidth is governed not only by the maximum memory frequency supported, but also from the processor frequency and number of cores per processor. With this generation of Intel Xeon processors, the memory controllers run a maximum of the highest non-Turbo core speed. Be aware that larger cache sizes provided by the higher core count processor models can have a beneficial impact on memory subsystem performance as well. Table 2 shows the impact the Intel Xeon E5-2600 processors have on memory frequency and bandwidth.

<table>
<thead>
<tr>
<th>Processor class</th>
<th>Maximum memory frequency</th>
<th>Processor model</th>
<th>Processor characteristics</th>
<th>Estimated bandwidth (GB/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced</td>
<td>1600MT/s memory</td>
<td>E5-2690</td>
<td>2.9GHz, 8C</td>
<td>80.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2680</td>
<td>2.7GHz, 8C</td>
<td>79.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2670</td>
<td>2.6GHz, 8C</td>
<td>79.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2665</td>
<td>2.4GHz, 8C</td>
<td>78.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2660</td>
<td>2.2GHz, 8C</td>
<td>77.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2650</td>
<td>2.0GHz, 8C</td>
<td>74.9</td>
</tr>
<tr>
<td>Standard</td>
<td>1333MT/s memory</td>
<td>E5-2640</td>
<td>2.5GHz, 6C</td>
<td>71.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2630</td>
<td>2.3GHz, 6C</td>
<td>70.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2620</td>
<td>2.0GHz, 6C</td>
<td>69.3</td>
</tr>
<tr>
<td>Basic</td>
<td>1066MT/s memory</td>
<td>E5-2609</td>
<td>2.4GHz, 4C</td>
<td>54.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2603</td>
<td>1.8GHz, 4C</td>
<td>48.0</td>
</tr>
<tr>
<td>Frequency</td>
<td>1600MT/s memory</td>
<td>E5-2667</td>
<td>2.9GHz, 6C</td>
<td>82.6</td>
</tr>
<tr>
<td>optimized</td>
<td></td>
<td>E5-2643</td>
<td>3.3GHz, 4C</td>
<td>75.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2637</td>
<td>3.0GHz, 2C</td>
<td>43.1</td>
</tr>
<tr>
<td>Low power</td>
<td>1600MT/s 1333MT/s</td>
<td>E5-2650L</td>
<td>1.8GHz, 8C</td>
<td>69.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5-2630L</td>
<td>2.0GHz, 6C</td>
<td>63.6</td>
</tr>
</tbody>
</table>

PowerEdge 12th generation servers are very modular by nature, and this new generation of servers offers great freedom in terms of how you populate a system’s memory upon purchase. You can now choose the exact quantity and type of DIMMs for your system instead of a limited set of memory configurations. This new memory ordering method offers you great flexibility, but if the memory type and population are not properly weighed at time of purchase, there may be negative performance side effects.
The focus of this white paper is to showcase the highest performing DIMM types in several configurations, and to show that the certain BIOS options can have a large impact on the overall performance of a server.

For more information on Dell PowerEdge 12th generation server memory technology, see the Memory for Dell PowerEdge 12th Generation Servers technical overview on Dell.com.

Test methodology

For the purposes of this white paper, we characterized the memory subsystem in two primary ways: memory bandwidth and memory latency. Memory bandwidth represents the rate at which memory can be read from or written to by a processor. Memory latency is the time it takes to initiate a 64-byte message transfer. Both metrics are important for evaluating memory subsystem performance. High memory bandwidth minimizes the time needed to transfer large amounts of data; therefore higher memory bandwidth is better. For the latency metric, measured in nanoseconds, lower is better. Both memory bandwidth and latency measurements were conducted on Novell SuSE Linux Enterprise Server (SLES) 11 SP2.

Memory bandwidth was measured using a version of the common STREAM benchmark, developed by John McAlpin of the University of Virginia. The version used for this test was compiled with OpenMP (parallel) support and optimized for the new generation of Intel Xeon processors. When testing overall system memory bandwidth, optimal results are achieved by running the STREAM benchmark with one thread per physical core. Although STREAM uses four separate vector kernels during the course of measurement (Add, Scale, Copy, and Triad), the Triad value is used for the purposes of this white paper. STREAM results are returned in values of MB/sec.

Latency measurements were taken using the lat_mem_rd subtest, part of the LMBENCH benchmark. Lat_mem_rd measures memory read latency for varying memory sizes and strides. The results are reported in nanoseconds per load. The entire memory hierarchy is measured, including onboard cache latency and size, external cache latency and size, and main memory latency. For the purposes of this white paper, the focus is main memory latency. Both local and remote NUMA node main memory latency is explored with lat_mem_rd, localizing the process and forcing memory access to local or remote using the numactl tool in Linux. The last 20 lines of output from main memory measurements of lat_mem_rd are averaged, and each test executed for three iterations. The median average result for each test type was selected to report latency results for this white paper. In addition, the concept of “loaded latency” is explored for some configurations. This is intended to represent average memory latencies when the NUMA node is already under heavy load. The workload chosen to provide the heavy memory load is STREAM, and n-1 real cores on a single socket were used with separate single threaded STREAM processes to load the local NUMA node targeted for latency experiments. The single free core on the physical processor under test was tasked to run the latency measurement, measuring memory latency to the local NUMA node. As with other latency measurements, the last 20 lines of output were averaged to produce the final loaded latency value.

As important as these memory subsystem specific characterizations are, the overall server impact must also be measured. We measured multiple types of workloads, as each workload has a different...
impact on memory bandwidth and latency. These workloads are all industry standard benchmarks, and represent a variety of server workloads.

**SPEC CPU2006** `int_rate_base` and `fp_rate_base` (hereafter referred to as `SPECint_rate_base2006` and `SPECfp_rate_base2006`) were chosen, as they are the leading industry standard CPU-centric benchmarks that use all available processor cores and threads. Each benchmark is comprised of multiple sub-benchmarks, each tailored to model different common computing tasks or scientific workloads, broken out into either integer or floating point. Both `int_rate` and `fp_rate` are throughput-based benchmarks. The “base” portion of the benchmark name indicates that a standardized set of compiler options were used to create the benchmark binaries. The `SPECfp_rate_base2006` benchmark in particular can be used as somewhat of a proxy for scientific workloads, and tends to be more sensitive to memory frequency and bandwidth differences than `SPECint_rate_base2006`. All CPU2006 measurements were conducted with the Novell SLES11 SP2 operating system.

**SPECjbb2005** evaluates the performance of server side Java by emulating a three-tier client/server system (with emphasis on the middle tier). The benchmark exercises the implementations of the JVM (Java Virtual Machine), JIT (Just-In-Time) compiler, garbage collection, threads and some aspects of the operating system. All logical processors on the system are exercised, and although considerable memory is allocated for the benchmark in the form of Java heap, this benchmark is generally less sensitive to memory frequency and bandwidth differences. **SPECjbb2005** was run on the Microsoft Windows Server 2008 Enterprise Edition R2 SP1 operating system.

**SPECpower_ssj2008** is the first industry standard performance per watt benchmark. Like **SPECjbb2005**, it is a Java-based benchmark with some workload similarities to JBB2005. However, the workload is run in different load levels, from 100% down to “active idle” in 10% increments. Active Idle is defined as a state where the system is ready to perform work, but there are no work requests. Performance and power data is collected for each workload interval using a separate benchmark controller system, an accepted power analyzer, and a temperature monitor. The controller system logs all power and temperature data, and communicates with the System Under Test (SUT) to control workload levels and pull result files at the end of the benchmark run. **SPECpower_ssj2008** was run on the Microsoft Windows Server 2008 Enterprise x64 Edition R2 SP1 operating system.

**Linpack** is a common high-performance computing (HPC) benchmark. The benchmark is constructed to be a system for solving a dense system of linear equations. The version run for this study is considered “standalone Linpack”, which is meant to run on a single system. Other implementations of Linpack, such as High Performance Linpack, are meant to run on a cluster of systems. Standalone Linpack was obtained from the Intel MKL (Math Kernel Library), and the version used to produce the data used in this study was optimized for the E5 series processors. Standalone Linpack measurements were conducted on the Novell SLES11 SP2 Operating System.

**VMmark 2** is one of the industry leading virtualization benchmarks, and was the first standardized methodology for comparing virtualized platforms. VMmark 2 consists of a collection of application-level workloads separated into multiple virtualized OS instances, as well as running infrastructure-intensive workloads such as storage vMotion and dynamic VM (Virtual Machine) relocation. The benchmark metric is based on the concept of “tiles”; each tile represents a set of application VMs focusing on three primary types of workloads. These workloads types include: Microsoft Exchange (email), DVD Store (an OLTP workload), and Olio (a Web 2.0 workload). The benchmark measures the scalability of the server platform (the number of tiles a given system can run concurrently), individual
application performance, and quality of service metrics. VMmark2 is normally run on pairs of identically configured servers with the VMs distributed between them, and both systems connected to shared storage. VMmark 2 measurements were conducted on VMware ESX 4.1 Update 2.

Local versus remote memory bandwidth and latency

Starting with the Intel Xeon processor 5500 series, NUMA systems have grown to dominate the Intel x86 server market. The NUMA architecture revolves around the concept that subsets of memory are divided into “local” and “remote” nodes for systems with multiple physical processors. Because modern processors are much faster than memory, a growing amount of time is spent waiting on memory to feed the processors. Since local memory is faster than remote memory in multi-socket systems, ensuring local memory is accessed rather than remote memory can only be accomplished with a NUMA implementation. The relative locality of a NUMA node depends on which core in a specific processor socket is attempting memory access. Accesses to the local NUMA node normally are lower latency and higher bandwidth than memory accesses to remote NUMA node(s).

For the last three generations of Intel Xeon processors, each processor socket has its own NUMA node associated with it. Figure 2 illustrates a 2-socket NUMA layout. In this example, CPU 0 and CPU 1 represent the physical processor package, not individual cores.

Figure 2. 2-Socket NUMA node architecture

Figure 3 illustrates NUMA node locality (NUMA node is considered local). In this example, for core 0 in physical processor CPU 0, the memory immediately adjacent to that processor socket is considered the local NUMA node. For core 1, which is part of physical processor CPU 1, the NUMA node considered local is the one hanging off of CPU 1. Each physical processor can have up to eight physical cores with the Intel Xeon E5-2600 processor, and up to 16 total logical processors (with Hyper-Threading enabled) per socket. For the purposes of illustration, only the first physical core on each processor socket is shown.

Figure 3. NUMA locality relative to individual cores
In Figure 4, we see that in terms of memory bandwidth, there is an approximately 50% decline going to a remote NUMA node compared to the local NUMA node. For the purposes of this illustration, cores initiating the STREAM test were on socket 0 only, and the NUMA node target was toggled between local and remote to measure the difference.

Example STREAM command to measure local NUMA node memory bandwidth (assuming 8 threads for STREAM, one per physical core on socket 0):

```
numactl --m 0 --cpunodebind=0 stream
```

Example STREAM command to measure remote NUMA node memory bandwidth (assuming 8 threads for STREAM, one per physical core on socket 0):

```
numactl --m 1 --cpunodebind=0 stream
```

Figure 4. Local vs. remote NUMA node memory bandwidth (higher is better)
In Figure 5, the relationship between local and remote NUMA node access is examined from the perspective of memory latency. In this case, remote memory latency is over 50% higher (worse) than local memory latency.

Example `lat_mem_rd` command to measure local NUMA node memory latency:

```
numactl -m 0 --physcpubind=0 lat_mem_rd (size in MB) (stride size)
```

Example `lat_mem_rd` command to measure remote NUMA node memory latency:

```
numactl -m 1 --physcpubind=0 lat_mem_rd (size in MB) (stride size)
```

**Figure 5.** Local vs. remote NUMA node memory latency (lower is better)

Figure 6 explores the relationship between the number of cores exercised with the STREAM benchmark and the relative disparity in achieved memory bandwidth from local to remote memory. As demonstrated, the differences between one, two, and four threads (each thread bound to one unique real core) used per socket drives the measured memory bandwidth up with thread count. After four threads, the slope gradient is significantly reduced. As a result, the relative disparity between local and remote memory bandwidth is stabilized.
BIOS memory mode selection guidelines for performance

Dell PowerEdge 12\textsuperscript{th} generation servers offer a range of memory modes that can be chosen in BIOS. The choices include Advanced ECC, Optimizer, Memory Mirroring, and Memory Sparing. All of these modes except for the Optimizer mode are offered for additional RAS (reliability, availability, and serviceability) features, with the Optimizer mode providing the highest memory subsystem performance characteristics.

The Advanced ECC mode (otherwise known as lockstep mode) uses two memory channels and “ties” them together to emulate a 128-bit data bus DIMM. This is primarily used to achieve Single Device Data Correction (SDDC) for DIMMs based on x8 DRAM technology. SDDC is supported with x4 based DIMMs in every memory mode.

The memory Optimized mode (or Independent Channel mode) allows memory channels to run independently of each other; for example, one can be idle, one can be performing a write operation, and the other can be preparing for a read operation. Memory may be installed in one, two, three, or four channels. To fully realize the performance benefit of the memory optimized mode, all four channels per physical processor should be populated.

Memory Mirroring mode allows memory to be “mirrored” from one channel into a paired channel. This provides 100% memory redundancy, but cuts the available memory to the operating system in half in addition to halving the available memory controllers and channels. This will have a direct impact on memory bandwidth, which should see less than half the memory bandwidth as compared to memory Optimized mode. This will have seriously negative performance implications, and should only be considered for applications that require mission critical levels of memory RAS.
Memory Rank Sparing mode reserves a memory rank as a spare as long as there are at least 2 DIMMs or 4 ranks available in a memory channel. If a memory rank experiences a high number of correctable errors above a certain threshold, the memory controller will swap the failing rank with the spare rank. The spare rank is invisible to the operating system, thus overall memory available to the operating system is decreased.

Advanced ECC compared to Optimizer mode

Depending on how the server was ordered, the server can be running in Advanced ECC or Optimizer Memory Operating modes. It is important to understand the effects of Advanced ECC mode on common server workloads, as the impact may be greater than expected.

Figure 7 illustrates that local and remote latency are not affected by turning on the Advanced ECC mode. However, under the loaded latency test, we see the true impact of Advanced ECC mode. Here, the latencies are over 225% that of Optimizer mode, suggesting that the reduction in memory bandwidth illustrated in Figure 8 is increasing queuing times and driving up the loaded latency measurement.

![Figure 7. Memory latency impacts of optimizer vs. Advanced ECC](image-url)
Examining the chart in Figure 8, we can see that the memory bandwidth measured in Advanced ECC mode is only 55% of that for Optimizer mode. This decline in memory bandwidth translates into other server workload impacts as well. Additionally, we can see that SPEC_int_rate_base2006 with Advanced ECC mode achieving only 87% of the performance seen with Optimizer mode. For the more memory bandwidth-sensitive SPECfp_rate_base2006 benchmark, we see Advanced ECC achieving only 71% of the performance seen with Optimizer mode. With the Java-based SPECjbb2005 benchmark, the impacts are considerably less pronounced. For SPECjbb2005, Advanced ECC mode achieves a full 97% of the performance of Optimizer mode, suggesting that Advanced ECC mode may not be particularly impactful for workloads that share many similarities with this benchmark.

Figure 8. Advanced ECC performance effects compared with memory Optimizer mode

![Advanced ECC performance effects compared with memory Optimizer mode](image)

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Optimizer</th>
<th>Advanced ECC</th>
</tr>
</thead>
<tbody>
<tr>
<td>STREAM</td>
<td>1.00</td>
<td>0.55</td>
</tr>
<tr>
<td>SPECjbb2005</td>
<td>1.00</td>
<td>0.97</td>
</tr>
<tr>
<td>SPECint_rate_base2006</td>
<td>1.00</td>
<td>0.87</td>
</tr>
<tr>
<td>SPECfp_rate_base2006</td>
<td>1.00</td>
<td>0.71</td>
</tr>
</tbody>
</table>

**Node Interleave effects**

Another option in the memory subsection of the BIOS is Node Interleave. This option is disabled by default, which means that NUMA mode is enabled. Conversely, enabling Node Interleave means the memory is interleaved between memory nodes, and there is no NUMA presentation to the operating system. Most modern operating systems have been NUMA-aware for many years, and schedulers have been optimized to ensure memory allocations are localized to the correct (closest) NUMA node. However, these schedulers are not perfect, and to achieve maximum performance with a given workload (especially for benchmarks), manual intervention may be required to ‘pin’ workload processes and threads to specific cores, taking care to ensure memory is always allocated to the local NUMA node for any given logical processor.

For some applications, where the memory required is larger than the memory available in a single memory node (such as very large databases), the memory allocation must necessarily occur on the remote node as well. It is in cases such as this, or for other applications that can’t be easily localized to a single socket/NUMA node combination, where enabling Node Interleave could have positive effects. Enabling Node Interleave was most common several years ago when NUMA was a relatively recent arrival to x86 servers and operating systems were not as NUMA aware. However, this situation has
radically improved, and today the number of customers that need to enable Node Interleave is diminishing.

In Figure 9, we see that by enabling Node Interleave, the following effects become apparent. Memory bandwidth as measured with the STREAM tool decreases by almost 25%. The effects on SPECint_rate_base2006 and SPECfp_rate_base2006 are less pronounced; for SPECint_rate_base2006, the decline in performance as compared to Node Interleave disabled is just over 7%, and an 8.6% decline in SPECfp_rate_base2006.

Figure 9. Node Interleave disabled vs. enabled
Memory type comparison

RDIMMs versus UDIMMs

In Figure 10, we see that RDIMMs and UDIMMs in the same basic memory configuration (8 x 4GB dual Rank DIMM population, 32GB total memory population), and clocked to a common 1333MT/s memory frequency show very similar performance characteristics. Both types score equivalently for SPECint_base_rate2006 and SPECfp_base_rate2006, as well as achieving the same memory bandwidth. Memory latency is the only real differentiation here, and UDIMMs are within 1-2\% of the latency figures measured with RDIMMs.

Figure 10. 1 DPC RDIMM vs. 1 DPC UDIMM DR 1333MT/s latency comparison

Figure 11 demonstrates the innate performance characteristics of RDIMMs versus UDIMMs. The fact that RDIMMs were downclocked to 1333MT/s for this comparison means that RDIMMs at 1600MT/s will score higher for SPECint_rate_base2006, SPECfp_rate_base2006 and STREAM as of Q2 2012. As higher frequency UDIMMs are adopted in the future, this delta should be reduced. For more information on frequency effects on RDIMM performance, see the Memory frequency section in this white paper.
Power comparison: 1.35V RDIMMs versus UDIMMs

Figure 12 illustrates a power comparison (system level power at AC) of RDIMMs versus UDIMMs at a common frequency, memory size, population, and voltage using the SPECpower_ssj2008 benchmark. This chart represents the relative system AC power savings at different load levels of the benchmark by using 1.35V UDIMMs compared to a baseline of 1.35V RDIMMs of the same number, size, and speed. The memory configuration for this test was 8 x 4GB dual rank DIMMs, running at 1333MT/s and 1.35V for both DIMM types. You can see that the power saved using UDIMMs rises from less than 1% at the 100% load level to just over 3% power savings at the 10% load level, and dropping down to less than 1% at active idle. Using UDIMMs, the power saved per DIMM, even at the 3% peak delta, is less than half a watt per DIMM. These power savings are relative to system configuration, and as more power consuming devices such as higher TDP processors, additional I/O expansion devices, or more hard drives are added, the proportion of memory power consumption to overall system power consumption will fall.
In order to achieve higher memory size, a common configuration of LRDIMMs is a 512GB total memory configuration, using a 2 DPC population of 16 x 32GB LRDIMMs. Figure 13 compares latency characteristics of 2 DPC RDIMMs in a 128GB total memory configuration against the 2 DPC LRDIMM configuration running at 1333MT/s. Our measurements demonstrated that LRDIMMs show 4% higher memory latency than RDIMMs, and 12% higher latency for remote NUMA node access. The loaded latency comparison shows LRDIMMs with 43% higher latency than RDIMMs. This is due to the fact that the LRDIMM memory buffer adds some latency in each direction, which impacts memory bandwidth in scenarios where there are mixed reads and writes. Lower memory frequency will add additional latencies.
The latency penalties of LRDIMMs are evident in other workload comparisons against similarly populated RDIMMs in addition to the expected disparities caused by slower memory frequency. For the SPECint_rate_base2006 benchmark, 2 DPC 1333MT/s LRDIMMs showed a decline in performance of almost 6% as compared to 2 DPC 1600MT/s RDIMMs. As we have seen in other comparisons, some integer workloads appear to be completely insensitive to memory latency or frequency effects. This is exemplified by 400.perlbench, which showed no performance delta using LRDIMMs.

Several other workloads showed moderate declines using LRDIMMs. Among these are 462.libquantum and 401.bzip2, both of which experienced declines of approximately 2.8% and 5.2%, respectively. Other workloads showed a more substantial negative impact using LRDIMMs. These workloads include 429.mcf and 471.omnetpp, both of which suffered an approximately 14% decline in performance compared to RDIMMs. These impacts suggest that although the lower frequency of LRDIMMs is playing an important role in the overall performance decline, the additional latencies of this DIMM type can be just as important, if not more, with many workloads.
For scientific workloads such as those contained in the SPECfp_rate_base2006 benchmark, the impact of using LRDIMMs is even more apparent. The overall SPECfp_rate_base2006 benchmark suffered an approximately 13.5% decline in performance compared with 2 DPC RDIMMs at 1600MT/s. However, some workloads such as 435.gromacs saw no appreciable impact in performance with LRDIMMs. Other workloads such as 436.cactusADM, 481.wrf, 437.leslie3D and 410.bwaves all saw a decline in performance with LRDIMMS of between 15 and 23%. Memory bandwidth also suffers an impact with LRDIMMs compared to RDIMMs, with a decline of 21% measured.
In considering larger memory configurations where the individual DIMM size is of great importance, there are two primary choices: quad rank RDIMMs and LRDIMMs. Both DIMM types come in 32GB sizes, and both are capable of 1333MT/s operation. However, quad rank RDIMMs are limited to 1333MT/s memory frequency at 1 DPC, and 2 DPC causes the memory frequency to drop to 1066MT/s. For LRDIMMs, 1333MT/s operation is achievable in both 1 DPC and 2 DPC memory configurations, dropping to 1066MT/s for 3 DPC.

In Figure 16, a common 512GB memory configuration is compared using 2 DPC LRDIMMs at 1333MT/s and 2 DPC quad rank RDIMMs at 1066MT/s with respect to memory latency. The results of our latency testing indicate that for local and remote memory latency, quad rank RDIMMs are slightly better in terms of memory latency than LRDIMMs in a 2 DPC configuration. However, when loaded latency is examined, LRDIMMs show a clear advantage. In this test, 2 DPC quad rank RDIMMs were 21% higher (worse) for loaded latency than 2 DPC LRDIMMs, suggesting although the lower memory frequency of the quad rank RDIMMs is playing a role in higher loaded latency measurements. However, the 21% increase in loaded latency for quad rank RDIMMs is less than the 31% expected based on frequency alone (see the Memory frequency section), suggesting that at a common memory frequency, quad rank RDIMMs would be superior.
Examining the Integer workload performance of these two DIMM types in a 2 DPC configuration, it is clear that 2 DPC LRDIMMs are marginally better in most cases than the quad rank RDIMMs, as illustrated in Figure 17. For the overall SPECint_rate_base2006 score, 2 DPC quad rank RDIMMs were 2% worse than 2 DPC LRDIMMs. However, as we have seen in other comparisons, the individual workload choice can see somewhat greater effects than the overall SPECint_rate_base2006 score might otherwise suggest.

Some workloads, such as 458.sjeng, saw no score change between 2 DPC LRDIMMs and 2 DPC quad rank RDIMMs. Other workloads saw more moderate effects of DIMM type selection. Among these were 403.gcc, 473.astar, 483.xalancbmk, and 429.mcf. All of these workloads showed a decline in performance of between 3% and 5% using quad rank RDIMMs at 1066MT/s, as compared to 2 DPC LRDIMMs with a memory frequency of 1333MT/s.
Figure 17. 2 DPC quad rank RDIMMs vs. 2 DPC LRDIMMs integer workload comparison

Examine the floating point workload performance of these two DIMM types in a 2 DPC configuration, it is noticeable that 2 DPC LRDIMMs are superior in all workloads to 2 DPC quad rank RDIMMs. Much of this can be attributed to the increased memory frequency and memory bandwidth provided by the LRDIMM configuration, as most of the workloads that make up the SPECfp_base_rate2006 benchmark are markedly more influenced by memory frequency and bandwidth than integer workloads. Figure 18 illustrates the difference between the performance of 2 DPC LRDIMMs and 2 DPC quad rank RDIMMs.

Workloads such as 444.namd show very little difference between LRDIMMs and quad rank RDIMMs— a scant 1% delta. However, other workloads, such as 482.sphinx3, 437.leslie3d, 433.milc, and 481.wrf, show a decline in performance of between approximately 9% and 11% using quad rank RDIMMs as compared with LRDIMMs.

Memory bandwidth also shows a similar decline of approximately 12% with 2 DPC quad rank RDIMMs running at 1066MT/s as compared with 2 DPC LRDIMMs running at 1333MT/s. Comparing this memory bandwidth decline to that seen with dual rank RDIMM memory bandwidth dropping from 1333MT/s to 1066MT/s (approximately 18%), it is apparent quad rank DIMMs are likely to show greater memory bandwidth than LRDIMMs at the same memory frequency.
Figure 18. 2 DPC quad rank RDIMMs vs. 2 DPC LRDIMMs floating point workload comparison

RDIMM voltage effects on system AC power consumption

Another element of DIMM type and frequency selection is the power impact of using DDR3L 1.35V DIMMs. These DDR3L 1.35V DIMMs come in both RDIMM and UDIMM selections, and have the capability of running some memory frequencies at 1.35V, compared to 1.5V of standard RDIMMs or UDIMMs. At the time of writing, DIMMs were not capable of running 1600MT/s at 1.35V, but DDR3L are capable of running 1333MT/s and 1066MT/s at this reduced voltage. In Figure 19, we compare the same set of 8 x 4GB DDR3L 1.35V 1333MT/s RDIMMs at both 1.5V and 1.35V with the SPECpower_ssj2008 benchmark to examine the power deltas of memory voltage. This benchmark runs at stepped workload intervals, from 100% down to active idle in 10% increments. We chose a smaller subset of these workload levels to illustrate the declining System AC power savings of voltage as the system is progressively less tasked with work. From the 100% to 50% load levels, we see the rising impact of lower voltage DIMMs from 2.29% at 100% to just below 3% System power at the 50% load level. This delta then starts a shallow decline from the 50% load level all the way to the 10% load level, where less than 2.59% power savings was measured. At active idle, the smallest delta was measured at 1.48%. Keep in mind that the power savings per DIMM is are relative to overall system power consumption, and the maximum power saved per DIMM at 1.35V is less than one watt. However, in larger memory configurations additional savings can be had, especially if the system is kept tasked at all beyond active idle. Over the course of a server’s lifetime, seemingly small power differences can add up to considerable power savings.
Memory frequency

Dell PowerEdge 12th generation servers offer a choice of DIMMs in several different memory frequencies. For the purpose of this white paper, frequency impacts on various server workloads were populated with 16 x 8GB dual rank RDIMMs factory operating at 1600MT/s. For testing lower frequency impacts on the workloads tested, the memory was downclocked from the factory 1600MT/s to 1333MT/s and 1066MT/s. This memory population of 16 x 8GB RDIMMs was chosen, as it provides optimal performance for the benchmarks tested. For the case of VMmark2, the memory population was 16 x 16GB dual rank 1600MT/s RDIMMs due to the high memory usage of this benchmark.

Figure 20 illustrates the memory frequency effects on memory bandwidth as measured with STREAM. We see an over 13% decline in memory bandwidth dropping down to 1333MT/s from 1600MT/s. Comparing 1066MT/s with 1600MT/s, the decline is expectedly even more pronounced at almost 29%.
For local and remote memory latency, our testing showed that 1600MT/s and 1333MT/s memory frequencies are essentially equivalent. At 1333MT/s, measurements showed approximately 5% worse latency for remote NUMA node access. However, the 1066MT/s memory frequency was measured at approximately 2% higher memory latency for local node access and 10% higher for remote. For loaded latencies, the effects were even more pronounced. We saw that 1333MT/s memory was measured 26% higher than 1600MT/s, and 1066MT/s memory measured 65% higher with loaded latency, as compared to memory running at 1600MT/s.
Examining SPECint_rate_base2006, the impacts to performance are obvious but not pronounced. Dropping the memory frequency from 1600MT/s to 1333MT/s resulted in a decline in just over 2%. However, the impact from dropping from 1600MT/s to 1066MT/s was more evident with a decline of approximately 6.8%.

One of the workloads that make up SPECint_rate_base2006 is 429.mcf. This workload, which is a combinatorial optimization workload modeling single-depot vehicle scheduling in a public mass transportation scenario, is considerably more affected by memory frequency than the overall SPECint_rate_base2006 score. We see a decline in performance of 5.7% moving from 1600MT/s to 1333MT/s. Dropping to 1066MT/s from 1600MT/s, we see a decline of 10.7%.

Some workloads, such as 400.Perlbench (PERL Programming Language), appear to be extremely insensitive to memory frequency. This workload saw no statistically significant decline in score at either 1333MT/s or 1066MT/s. Other workloads appear somewhat more impacted by memory frequency. Among these are 471.omnetpp (Discrete Event Simulation) and 473.Astar (Path-Finding Algorithm), both of which saw a decline in performance of between 3% and 5% at 1333MT/s from 1600MT/s. Reducing the memory frequency to 1066MT/s, these workloads suffered declines ranging from 6% to 10%.

When we examined SPECjbb2005, we noticed that this workload appeared to be relatively insensitive to memory frequency, especially when compared to most floating point and integer workloads that make up CPU2006. This benchmark is intended for evaluating servers running typical Java business applications. We saw that performance declined by less than 1% at 1333MT/s as compared to 1600MT/s, and by approximately 4% when dropping the memory all the way to 1066MT/s.

Figure 22. Memory frequency impacts on integer server workloads
For SPECfp_rate_base2006, the effects of memory frequency are more pronounced because most of the workloads use larger data sets and have a higher sensitivity to memory bandwidth constraints. For the overall SPECfp_base_rate 2006 score, dropping the memory frequency from 1600MT/s to 1333MT/s resulted in a decline of 6.8%. Dropping the frequency even further to 1066MT/s, we see a decline of 16.8% as compared to 1600MT/s performance. Examining some of the individual workloads shows even larger impacts. For the 410.bwaves workload, which is a computational fluid dynamics blast wave simulator, the impacts are considerable. Going from 1600MT/s memory to 1333MT/s memory, we see a decline in performance of approximately 12.5%. Dropping to 1066MT/s is even more pronounced, dropping the score by 27.3% as compared to 1600MT/s memory. The workload 433.milc, which is a Quantum Chromodynamics four-dimensional lattice gauge theory simulation, showed very similar declines to 410.bwaves.

A floating point workload that showed a modest impact from memory frequency changes was 465.tonto (Quantum Chemistry). We saw an approximately 5.9% decline in performance when memory frequency was reduced to 1333MT/s, and approximately 14.5% when dropped to 1066MT/s. Not all floating point workloads are equally sensitive to memory frequency as we discovered during our testing. The 454.calculix workload (Structural Mechanics) saw very little overall impact when changing memory frequency. There was no change in performance going from 1600MT/s memory to 1333MT/s, and an only 0.9% decline when the frequency was reduced to 1066MT/s.

Figure 23. Memory frequency impacts on floating point workloads
For standalone Linpack, a common HPC workload, the effects of memory frequency are even less pronounced than the other benchmarks tested. Dropping the memory frequency to 1333MT/s gave us 99.92% of the score that 1600MT/s provided. Going to 1066MT/s memory was more noticeable, providing 99.56% of the score that was measured with 1600MT/s memory. Most of this relative insensitivity to memory frequency can be traced to the behavior of the workload, which is optimized for the Xeon E5-2600 series of processors, and run almost entirely out of processor cache. Large amounts of memory are recruited at the beginning of the test (51GB used), but very little memory activity appears to take place outside the processor caches during runtime.

**Figure 24. Memory frequency on the Linpack (HPC) benchmark**

![Memory Frequency Comparison](image)

**Memory frequency and virtualization workloads**

For virtualized workloads such as VMmark2, we concluded after testing that the amount of memory is more critical to overall benchmark performance than memory frequency. There was almost no difference detected between 1600MT/s and 1333MT/s memory frequencies, and the system was able to support 10 tiles. When we dropped the memory to 1066MT/s, we saw a decline in the number of tiles the system could support. The tile count sustained at 1066MT/s was 8 tiles, and we saw an overall decline in performance by 11%. See Figure 25.
Memory frequency effects on power

Another impact to consider of memory frequency is the power utilization. We devised a series of experiments to try to illustrate these impacts. In Figure 26, the power savings of 1333MT/s/1.35V and 1066MT/s/1.35V is compared to a baseline of 1600MT/s/1.5V. All configuration parameters were held the same between these tests except the memory configuration. For the 1600MT/s/1.5V baseline, 8 x factory-rated 4GB dual rank 1600MT/s RDIMMs were run at default frequency and voltage. For the other two data points, 8 x factory-rated 4GB dual rank 1333MT/s LV RDIMMs were run at 1333MT/s/1.35V and then downclocked to 1066MT/s/1.35V.

The workload chosen for this power comparison was SPECpower_ssj2008. The power deltas illustrated in Figure 26 are in comparison to 1600MT/s 1.5V RDIMMs and were measured in overall System AC power. We can see that the power savings of 1333MT/s low voltage DIMMs is approximately 3.5% at the 50% load level, and holds steady until the load level declines to below 10% (active idle). For the case of 1066MT/s low voltage memory, the power savings are maximized at the 50% and decline all the way to active idle.
Balanced versus unbalanced

You must also consider the effects of memory population from the standpoint of achieving an even DIMM population count across all memory channels. A balanced memory configuration for Dell PowerEdge 12th generation servers means that 8, 16, or 24 DIMMs are populated in 2-socket configurations, and where the DIMMs are evenly distributed across all 8 total memory channels. This means that for any of the three balanced memory configurations, the DIMMs would be populated in 1 DPC, 2 DPC, or 3 DPC. In our testing, balanced memory configurations are always superior to the alternative.

To change the memory channel population away from even distribution so that some memory channels have more DIMMs than others, or to populate channels with a mix of DIMM capacities is considered an unbalanced memory configuration. Because not all memory channels are equivalently loaded in some unbalanced memory configurations, the BIOS is unable to achieve optimal memory channel interleaving. This will have a negative impact to memory bandwidth and overall server workload performance.

Figure 27 shows a typical balanced memory configuration with 1 DPC. Other balanced memory configurations would be with 2 DPC or 3 DPC, where all channels are filled with the same number of DIMMs across both sockets.
Figure 27. Balanced memory configuration (1 DPC, all 4 channels populated)

Figure 28 shows a typical unbalanced memory configuration where two of the memory channels on each processor are populated at 2 DPC, and the other two memory channels are populated at 1 DPC.

Figure 28. Unbalanced memory configuration (mixed 1 DPC and 2 DPC)

Another type of unbalanced memory configuration that was not measured for this white paper mixes memory ranks within the same memory channel. For example, each memory channel would have a mix of single rank and dual rank memory. Although not measured, it is expected that there will be moderately negative impacts to memory performance due to the inability of the memory controller to interleave evenly between different DIMM ranks.

In addition to the standard balanced and unbalanced configurations, there is another concept known as a “near balanced” memory configuration. This memory configuration has all memory channels equally populated with DIMMs of the same rank structure, but each channel contains a mix of DIMM capacities. An example would be one 4GB and one 8GB RDIMM populated per channel. This memory configuration has none of the drawbacks of classic unbalanced configuration, and typically scores approximately the same as a classic balanced configuration.

In the course of our testing, we compared several server workloads and memory bandwidth using a balanced memory configuration at 1 DPC compared to an unbalanced configuration which populated 2 DPC for the first and second memory channels and 1 DPC for the third and fourth channels. For
memory bandwidth measured by STREAM, the impact was felt sharply as performance declined by over 45% by moving to an unbalanced configuration.

For SPECint_rate_base2006, the effects were more muted. Performance declined with the unbalanced configuration by over 8% compared with a balanced memory configuration. Different effects were noticed depending on the individual workload. Some workloads, such as 400.perlbench (a Perl Programming Language workload) saw no deleterious effects whatsoever running in an unbalanced configuration. Other workloads, such as 403.gcc (a C Compiler workload) and 473.astar (a Path Finding algorithm) saw a decline in performance with the unbalanced memory configuration of between 13% and 15%. Certain workloads, among them 471.omnetpp (a Discrete Event Simulation) and 483.xalancbmk (an XML processing test), saw a decline of approximately 21% in an unbalanced memory configuration.

Figure 29. Balanced vs. unbalanced memory configurations and effects on integer workloads

For SPECfp_rate_base2006, the effects were similar to the integer workloads. Performance again declined with the unbalanced configuration by over 8% compared with a balanced memory configuration. There were a range of effects observed in the individual workloads ranging from a negligible effect to impacts of up to 20%.

Some workloads, such as 416.gamess (Quantum Chemical Computations), saw less than 1% decline going to unbalanced mode. This is within the run-to-run variation of the benchmark, which is not statistically significant. Other workloads seemed to see some deleterious effects of unbalanced mode, such as 465.tonto and 470.lbm. These two workloads saw an approximately 10% to 11% decline in
performance in an unbalanced configuration. The third tier of workloads was the most sensitive to unbalanced memory configurations. Workloads that made up this tier include 482.sphinx3 (Speech Recognition) and 450.soplex (Linear Programming Optimization). These two workloads saw a decline in performance of between 19% and 20% with an unbalanced memory configuration.

These results, especially the fact that the overall SPECfp_rate_base2006 score declined by the same amount as SPECint_rate_base2006, suggests that workload effects are highly variable. Many workloads that might otherwise be negatively impacted to a greater extent by the fact that unbalanced memory bandwidth is only 54% of balanced memory bandwidth nonetheless are being buoyed by workload memory access patterns.

Figure 30. Balanced vs. unbalanced memory configuration effects on floating point workloads

---

Memory channel population

The Intel Xeon E5-2600 processors support four memory channels per physical processor. Each memory channel is operated by its own memory controller. When run in memory Optimizer (independent channel) mode, we have seen that the system performs better than in the other RAS-centric memory operating modes. For the purposes of this white paper, we wanted to chart the effects of memory channel population and the effects on a select few benchmarks. By populating less than the full 4 memory channels per processor results in lower memory bandwidth and a potential loss in performance of many server applications that are sensitive to memory subsystem changes. This test
was accomplished using pairs of 8GB dual rank 1600MT/s RDIMMs in each memory channel, in order to provide sufficient memory for some of the workloads we tried. The effects should be identical between 1 and 2 DIMMs per channel.

As Figure 31 indicates, the overall system memory bandwidth declines as memory channels are depopulated. This decline occurs in an almost linear fashion as fewer memory channels are contributing to overall bandwidth. This decline in memory bandwidth will have direct performance impacts on standard server workloads, especially scientific workloads that tend to be the most memory bandwidth sensitive. Anything less than 4 memory channels populated indicates an unbalanced memory configuration.

![Figure 31. Memory channel population effects on overall memory bandwidth](image)

We also examined the effects of memory channel population on a number of integer server workloads, focusing on two primary benchmarks: SPECint_rate_base2006 and SPECjbb2005. Several interesting effects were noted with individual workloads within SPECint_rate_base2006, so those are included as well. Refer to Figure 32 for the integer workload measurement data.

As memory channels are depopulated, the effects to the overall score for SPECint_rate_2006 are not severe until only a single memory channel is left per socket. We saw that the overall SPECint_rate_base2006 score decline by almost 2% dropping to 3 channels, and 7% when dropping to only 2 channels. However, the impact was much greater when only a single memory channel was populated. In this configuration, the decline was approximately 24.5% as compared to the balanced 4 channel memory configuration.

There were some workloads such as 400.perlibench (a test of the PERL programming language) that saw almost no effects from memory channel depopulation, all the way to 1 channel where the effect was still less than a 1% decline from 4 channel population. This suggests that this workload is extremely insensitive to memory bandwidth constriction, and may be running almost entirely out of processor cache (speculation). The 456.hmmer workload (a gene sequence searching test) is similarly
insensitive to bandwidth constriction, at least until 2 and 1 channel populations. For 2 channel population, the decline is less than 2%, and for 1 channel population the decline is just under 9%.

Other workloads were more impacted by channel depopulation. Workloads such as 403.gcc (C Compiler) and 473.astar (Path-Finding algorithm), dropping from 4 to 3 channels, saw a subtle decline of under 4%. With just 2 memory channels populated, the decline was approximately 13% as compared to a full 4 channel population. With 1 channel populated, these workloads experienced a decline of over 30%. 403.gcc in particular was harder hit by channel population, seeing a decline of 37%.

Several integer workloads in particular appeared to be the most sensitive to memory channel depopulation. Two of these workloads were selected for illustration purposes. The 429.mcf and 483.xalancbmk (XML Processor) workloads, dropping from 4 memory channels to 3 saw a decline in performance of approximately 4-5%. With a 2 channel memory population, we saw a negative impact of between 13% and 16.8% respectively. When only a single memory channel was populated, the effects were to almost halve the performance. Both workloads saw an approximately 45% decline in performance as compared to a 4 channel memory population.

For SPECjbb2005, we did not measure a single memory channel population due to difficulties with memory requirements while keeping other elements of our measurement consistent. We were able to chart the effects of 4, 3, and 2 memory channels populated with this benchmark. We discovered that again, SPECjbb2005 is not highly sensitive to memory bandwidth constraints, at least with the memory configurations we were able to test. The decline from 4 memory channels to 3 for this benchmark was slightly less than 1%. Dropping to 2 memory channels saw an impact of just under 10%.

Figure 32. Memory channel population effects on integer server workloads

In addition to the integer workloads we used to characterize memory channel population, we also used SPECfp_rate_base2006 to showcase workloads that we anticipated would be more sensitive to memory configurations and were memory bandwidth limited. For the overall SPECfp_rate_base2006
score, we saw that the 3 channel population caused a 9% decline in performance. For 2 and 1 channel memory channel population, the relative declines compared to 4 channel population were 26.8 and 53.9%, respectively.

Some floating point tests do not seem to be sensitive at all to memory channel population. A case in point is 444.namd (a Molecular Dynamics simulation) where declines are barely noticeable, even at 1 channel population.

Some SPECfp_rate_base2006 workloads, such as 447.dealII (Finite Element Analysis) and 465.tonto (Quantum Chemistry), show moderate impact with memory channel depopulation. Neither workload displays significant impact at 3 channels, but at 2 channels performance declines approximately 10%. At 1 channel population, performance declines by approximately 30%.

Certain floating point workloads appear to be highly affected by memory channel depopulation. Among these are 436.cactusADM (Computational Electromagnetics), 433.milc (Quantum Chromodynamics), and 410.bwaves (Fluid Dynamics blast wave simulation). At 3 channel population, these workloads suffer a decline of between 15% and 20%. At 2 channel memory population, all three workloads suffer an approximately 45% decline in performance. At a single memory channel populated, these three workloads experience a decline of approximately 75%. This behavior suggests workloads of this type are extremely sensitive to memory bandwidth reduction.

Figure 33. Memory channel population effects on floating point server workloads
UDIMM rank and channel population comparison

UDIMMs come in two rank structures: single rank and dual rank. Single rank UDIMMs typically constitute the lowest capacity DIMMs available, and are often the least costly of all memory options. Comparing single rank UDIMMS in 1 DPC and 2 DPC populations against dual rank UDIMMs, some conclusions can be drawn. Single rank UDIMMs in either 1 DPC or 2 DPC populations are generally capable of lower memory bandwidth than dual rank UDIMMs, and tend to have higher local and remote memory latencies.

For dual rank UDIMMs, some differences are seen between 1 DPC and 2 DPC memory populations. Dual rank UDIMMs in a 1 DPC configuration tend to have the best memory bandwidth of all available UDIMM rank and DIMM per channel combinations, but have local memory latency about 4% higher than 2 DPC dual rank UDIMM configurations. With 2 DPC dual rank UDIMMs, memory bandwidth is slightly constrained, measuring approximately 3% less than 1 DPC dual rank UDIMM configurations. However, this memory configuration is optimal for local and remote memory latency when compared to the other available UDIMM rank and DIMM per channel combinations.

In the case of dual rank UDIMMs, 1 DPC and 2 DPC memory configurations show more differences than dual rank RDIMMs in terms of memory bandwidth and standard server benchmarks. When memory bandwidth is compared between the 2 dual rank UDIMM channel populations, we see that the 1 DPC population measures only 97% of the memory bandwidth as measured by STREAM than the 2 DPC UDIMM population. For SPECint_rate_base2006, both UDIMM channel populations perform equivalently. However, there is a minor decline of 1% noted when SPECfp_rate_base2006 is compared between the 2 channel populations in the favor of the 1 DPC UDIMM configuration. Refer to Figure 35.
Figure 35. UDIMM channel population effects on server benchmarks and memory bandwidth

RDIMM rank and population per channel guidelines

In populating memory channels with various numbers of different RDIMMs, we see a variety of effects that are worth exploring. For RDIMMs, the effects are extremely minimal as illustrated in Figure 36. For all measures of latency, 1 DPC and 2 DPC dual rank RDIMM configurations are within 1% of each other. 2 DPC dual rank RDIMMs may have an almost imperceptible latency advantage over 1 DPC RDIMMs. The memory bandwidth difference between 1 DPC and 2 DPC configurations is also not significant.
Figure 36. DIMM per channel population effects on memory latency with RDIMMs

RDIMM rank and channel population

RDIMMs come in several rank structures, the most common of which are single rank and dual rank. As we saw in the last comparison, the latency differences between 1 DPC and 2 DPC dual rank RDIMMs are equivalent or within 1%. Memory bandwidth is also equivalent. Figure 37 illustrates the differences between dual rank and single rank RDIMMs in 1 DPC and 2 DPC configurations.

When we consider single rank RDIMMs, which typically make up the smallest size DIMM offerings, some differences appear. Although latencies for 1 DPC and 2 DPC single rank RDIMMs are equivalent to 1 DPC dual rank RDIMMs, memory bandwidth paints a different picture. For 1 DPC single rank RDIMMs, memory bandwidth is 11% lower than that seen with dual rank RDIMMs in either 1 DPC or 2 DPC configurations. For 2 DPC single rank RDIMMs, the reduction in memory bandwidth is lessened. In this configuration, single rank RDIMMs show a decline of only 3% in comparison with dual rank RDIMMs. Much of these differences can be attributed to rank and channel interleaving behaviors that differ with single rank and dual rank RDIMMs. In general, dual rank RDIMMs are preferable to single rank RDIMMs, but the single rank RDIMMs are only problematic for memory bandwidth in 1 DPC configurations.
Since RDIMMs are currently limited to 1333MT/s speeds in 3 DPC populations, memory in this configuration cannot match the raw performance of 1600MT/s RDIMMs in a 1 DPC or 2 DPC configuration. However at the same memory frequency, memory bandwidth is equivalent between 2 DPC and 3 DPC.

In Figure 38, all three measures of memory latency are within statistical equivalence between 2 DPC RDIMMs at 1333MT/s and 3 DPC RDIMMs at 1333MT/s.
Conclusion

Dell PowerEdge 12th generation servers offer great performance potential and customization options, but keep some elements in mind when configuring your system to fit a particular role.

First and foremost, customers concerned about extracting all available performance out of the memory subsystem should strive to populate all memory channels evenly with the same rank, size, and speed of DIMMs. For situations that demand mixing DIMM sizes, all memory should be chosen based on identical rank structure, and all memory channels should be populated with an identical mix of DIMM types.

Next, choose the appropriate processor to support the expected memory frequency. For example, if your workload requires a memory speed of 1600MT/s, select a processor model that supports this speed. Additionally, higher frequency processors with higher core counts do influence memory bandwidth, so this something else to consider when planning for memory-intensive workloads.

For most system configurations, 1 DPC and 2 DPC memory configurations are more optimal than 3 DPC, due to the fact that memory frequency will be reduced when all slots in a given memory channel are populated. Only when memory capacity is of utmost importance should 3 DPC memory configurations be evaluated. For these larger memory configurations, LRDIMMs hold some advantages over quad rank RDIMMs and should be considered when sizing 512GB memory configurations at 2 DPC.

Some BIOS options can play a major role in memory performance. For most performance-centric customers, Optimizer memory operating mode is suggested over Advanced ECC, due to the significant performance penalties implicit in running the memory in this RAS mode. Other RAS-centric memory operating modes impact performance even more, and should not be considered if you intend to obtain maximum memory subsystem performance. In addition, Node Interleave mode should
always be left disabled to allow normal NUMA operation except in very specialized cases where an application memory size is too large to fit inside one NUMA node.

Use RDIMMs over UDIMMs in most cases due to the greater selection, higher frequency, and greater capacity capabilities of RDIMMs. For these two DIMM types, dual rank is generally preferable to single rank DIMMs. However, UDIMMs may be viable in cases where power considerations are on equal or greater footing than simple memory performance. For cases where power savings is of interest, DDR3L DIMMs should be used at a frequency that permits 1.35V memory operation.

Finally, memory frequency is a major driver for overall memory subsystem performance, and 1600MT/s memory is preferred over 1333MT/s for workloads that are sensitive to memory bandwidth limitations such as scientific and HPC workloads. However, even among general computing workloads there are some that exhibit higher sensitivity than others to memory frequency. We recommend that you evaluate your workload to determine whether the uplift in performance provided by higher frequency memory is worth the tradeoff in price and the requirement to use Advanced E5-2600 processor models.
References

1. Memory for Dell PowerEdge 12th Generation Servers

2. Dell PowerEdge 12G servers

3. Dell PowerEdge R720 Technical Guide

4. Intel E5-series Xeon processor details
   http://ark.intel.com/products/family/59138/Intel-Xeon-Processor-E5-Family/server

5. Intel Turbo Boost technology
   http://www.intel.com/go/turbo

6. Configuring Low-Latency Environments on Dell PowerEdge 12th Generation Servers
   http://en.community.dell.com/techcenter/b/techcenter/archive/2012/05/07/configuring-low-latency-environments-on-dell-poweredge-12th-generation-servers.aspx

7. Memory Selection Guidelines for HPC and 11G PowerEdge Servers

8. SPEC CPU2006 Benchmark
   http://www.spec.org/cpu2006

9. SPEC JBB2005 Benchmark
   http://www.spec.org/jbb2005

10. The STREAM Memory Bandwidth Benchmark
    http://www.cs.virginia.edu/stream

11. The LMbench benchmark (lat_mem_rd)
    http://www.bitmover.com/lmbench

12. The VMmark2 Virtualization Benchmark
    http://www.vmmark.com