



SUPERMICRO SERVERS EQUIPPED WITH 3RD GEN AMD EPYC™ PROCESSORS BOOST ORACLE DATABASE 19C PERFORMANCE BY 54%

Supermicro A+ Servers Powered by AMD EPYC Processors Accelerate Oracle Database 19c Performance

Executive Summary

Supermicro has closely partnered with AMD since becoming one of the first server vendors to bring 1st Gen AMD EPYC™ processors with “Zen” microarchitecture to market in our H11 platforms in 2017. In 2019, Supermicro launched its first family of H12 generation AMD processor-powered Supermicro A+ servers with 2nd Gen AMD EPYC processors to deliver a new level of integration and superior performance for modern datacenters. Today, the new 3rd Gen AMD EPYC Processors in the new Supermicro A+ servers deliver up to 19% more instructions per clock than the previous generation¹.

3rd Gen AMD EPYC processors have up to 64 “Zen 3” cores per CPU, introduce new levels of per-core cache memory, and offer PCIe® 4.0 connectivity bandwidth. They also include AMD Infinity Guard², which offers Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP) to help enhance security in

TABLE OF CONTENTS

Executive Summary	1
AMD EPYC Overview	2
AMD EPYC 74F3 Processors Deliver Superior Performance.....	3
Conclusion.....	12
APPENDIX A – Oracle Linux & Configuration	13
APPENDIX B – Oracle Database 19c Configuration	15

SUPERMICRO

Supermicro (Nasdaq: SMCI), the leading innovator in high-performance, high-efficiency server, and storage technology is a premier global provider of advanced server Building Block Solutions® for Enterprise Data Center, Cloud Computing, Artificial Intelligence, and Edge Computing Systems. Supermicro is committed to protecting the environment through its “We Keep IT Green®” initiative and provides customers with the most energy-efficient, environmentally friendly solutions available on the market.



virtualized environments. SEV-SNP expands on the SEV features found in earlier EPYC processors by adding memory integrity capabilities that can help prevent hypervisor-based attacks by creating an isolated execution environment for each virtual machine.

AMD EPYC 7003 Series Processors Deliver Flexibility, Performance, and Security Features

Overview

AMD EPYC 7003 Series Processors retain the proven Multi-Chip Module (MCM) Chiplet Architecture of prior successful AMD EPYC server-class processors with further improvements and upgraded “Zen 3” compute cores.

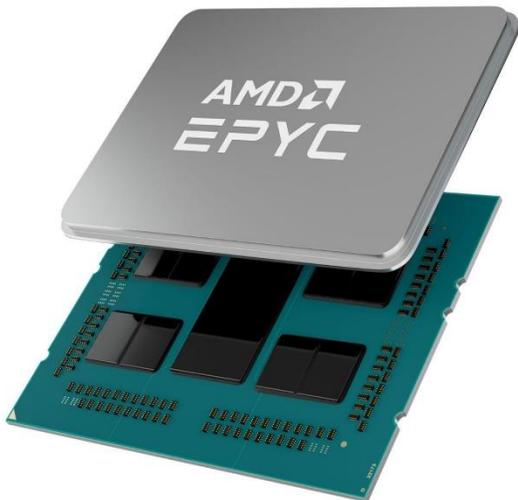


Figure 1 - AMD EPYC 7003 processor

A. “Zen 3” Core Microarchitectural Overview

AMD EPYC 7003 Series Processors are built with new 7nm “Zen 3” compute cores that provide an Instructions per Cycle (IPC) uplift over previous “Zen” generations. Each core supports Simultaneous Multi-threading (SMT) that allows running two simultaneous threads per core when enabled. Every core includes both an optimized 32 KB L1 cache and a private 512 KB Unified (Instruction/Data) L2 cache.

B. Core Complex (CCX) and Core Complex Die (CCD)

Up to eight “Zen 3” compute units share an L3 cache of up to 32 MB in a grouping referred to as a Core Complex (CCX). Enabling SMT on a core means that a CCX may support up to 16 concurrent hardware threads, up to 4 MB of L2 cache, and up to 32 MB L3 cache, which is shareable across all cores within the CCX. AMD EPYC 7003 Series Processors contain a single CCX called a Core Complex Die (CCD) on a single die. See Figure 2.



Figure 2: AMD EPYC 7003 I/O Die

C. I/O Die (Infinity Fabric™ technology)

Each CCD connects to memory, I/O, and each other through the I/O Die (IOD) via a dedicated high-speed or Global Memory Interconnect (GMI) link. The IOD also contains memory channels, PCIe Gen4 lanes, and Infinity Fabric™



links. All dies (or *chipllets*) interconnect with each other via AMD's Infinity Fabric Technology. The fabric clock (FCLK) can now run up to 1600 MHz and thus be coupled with DDR4-3200 Memory DIMMs that are also running at 1600 MHz (MEMCLK) to further reduce memory latency.

NUMA Topology

AMD EPYC 7003 Series Processors use a Non-Uniform Memory Access (NUMA) micro-architecture. Users can optimize this NUMA topology for their specific operating environment and workload by adjusting the BIOS NUMA Nodes Per Socket (NPS) setting.

New Software- Visible Features

The new “Zen 3” core in AMD EPYC 7003 Series Processors includes new and improved Instruction Set Architecture (ISA) features, such as:

- ShadowStack CET (control-flow enforcement) technology.
- AVX2: VAES & VPCLMULQDQ w/256-bit support
- Broadcast TLB invalidation (INVLPG & TLBSYNC Instructions)
- Fast short rep moves
- Predictive Store Forward Disable
- Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP)
- Process control ID
- Memory protection keys for users

The emerging boom in the number of IoT devices is driving a critical need for faster IoT gateway data analytics. Supermicro offers an industry-leading portfolio of AMD EPYC based servers via our Server Building Block Solutions® that empower customers to build application-optimized solutions across a wide range of possible configurations from single-socket mainstream and WIO servers to high-end Ultra server systems and multi-node systems, including BigTwin® and TwinPro® servers.

AMD EPYC 74F3 Processors Deliver Superior Performance on Transactional Processing Workloads with Oracle Database 19c

Supermicro AS -1014S-WTRT Server

This benchmark ran Oracle Database 19c on a high density single-socket Supermicro AS -1014S-WTRT server powered by AMD EPYC 7F72 and AMD EPYC 74F3 processors, as shown in Table 1. The high core density of AMD EPYC processors makes the Supermicro AS-1014S-WTRT an ideal, compact (1U, 25.6”

deep), and cost-effective solution for database processing and enterprise application workloads. Customers may choose AMD EPYC 8, 16, 24, 32, and 64 core, and P variant CPU SKUs that offer unique pricing benefits compared to dual-socket-capable AMD CPUs. The benchmarking process selected a single-socket system because it offers a unique and robust value proposition capable of predictable workload performance scaling and because 3rd Gen AMD EPYC cores can consolidate workloads that previously required dual- or multi-socket systems into a single-socket scale-up server.

Server	Form Factor	System Memory	Drive Bays	Network Controllers
AS-1014S-WTRT	1U Rackmount	8 x DDR4 slots up to 3200MHZ	4 x hot swap 3.5" or 2.5" SATA3 or NVMe drives	Dual Broadcom 10G Base -TLAN ports

Table 1: Benchmarks CPU Model Specifics



Image 1 - AS -1014S-WTRT

AMD EPYC 74F3 Processors Boost Scaling Efficiency by 54%

Oracle Database delivers leading-edge innovations in relational database management systems (RDBMS) for on-premises, cloud, and hybrid workloads with exceptional performance and ease of use. AMD EPYC™ 74F3 processor-based systems deliver high performance for Online Transaction Processing (OLTP) with Oracle Database 19c. Table 2 shows the feature differences between the 3rd Gen EPYC 74F3 and 2nd Gen EPYC 7F72 CPUs.

CPU	Series	Base Frequency	Boost ⁴ Frequency (up to)*	Core Processors	Memory Channels	Maximum Memory/Socket(DDR4-3200GHz)	PCIe® Gen4 Lanes / System
AMD EPYC 74F3	3 rd Gen 7003	3.2 GHz	4.0 GHz	24	8	4 TB	128
AMD EPYC 7F72	2 nd Gen 7002	3.2 GHz	3.7 GHz	24	8	4 TB	128

Table 2: Feature Comparison Between 2nd and 3rd Gen AMD EPYC CPUs

Benchmarking with Oracle Database 19c and AMD EPYC 74F3 Processors

Supermicro engineers tested the relative performance³ of the 3rd Gen AMD EPYC 74F3 processor versus the 2nd Gen AMD EPYC 7F72 processor and obtained the results shown in Figure 3.

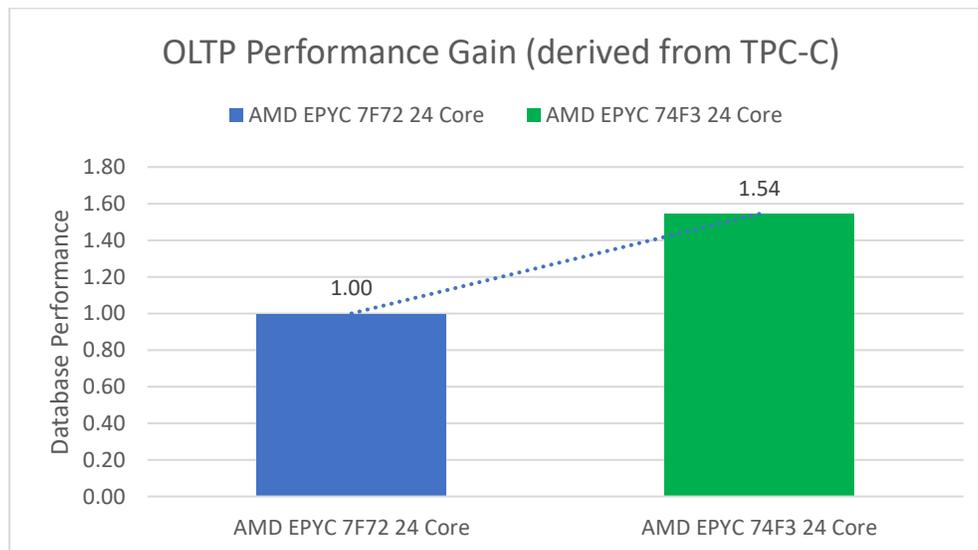


Figure 3. The relative performance of the 3rd Gen AMD EPYC 74F3 processor vs. the 2nd Gen AMD EPYC 7F72 processor

Benchmarking with Oracle Database 19c

Both of the CPUs shown in Table 2 above are optimized for higher frequencies and excel at single-threaded workloads that prefer limited core counts.

The OLTP workloads used for this benchmark are derived from the TPC-C Benchmark; they model an order fulfillment system where the database receives requests for data, adds new data, and makes multiple changes to that data from a large number of users. The results show increased transaction throughput from the 3rd Gen AMD EPYC 74F3 processor compared to the 2nd Gen AMD EPYC processor. Customers can right-size compute power to their application needs, thereby helping lower the total cost of ownership (TCO). The OLTP workload is derived from the TPC-C Benchmark, and as such, is not comparable to published TPC-C Benchmark results, as the OLTP workload results do not comply with the TPC-C Benchmark.

System Configuration and Setup

The hardware configuration for these benchmarks consisted of:

- One Supermicro AS -1014S-WTRT server is equipped with a 24-core 3rd Gen AMD EPYC 74F3 CPU (see Table 2, above) and 512GB RAM.
- One Supermicro AS -1014S-WTRT server is equipped with a 24-core 2nd GEN AMD EPYC 7F72 CPU (see Table 2, above) and 512GB RAM.



- One Supermicro SSE-C3632SR 40GB data switch.



- One Supermicro SSC-X3349T 10GB management switch.



- One Supermicro 2024US-TRT server for the benchmarking/Hammer DB.



Supermicro system architects executed these benchmarks using the industry standard HammerDB Benchmark Tool. Both systems used an identical HammerDB 3.2 configuration to execute the benchmarks under similar conditions to obtain similar results. Please see the Appendix for basic HammerDB configuration, benchmark database schema, and execution scripts.

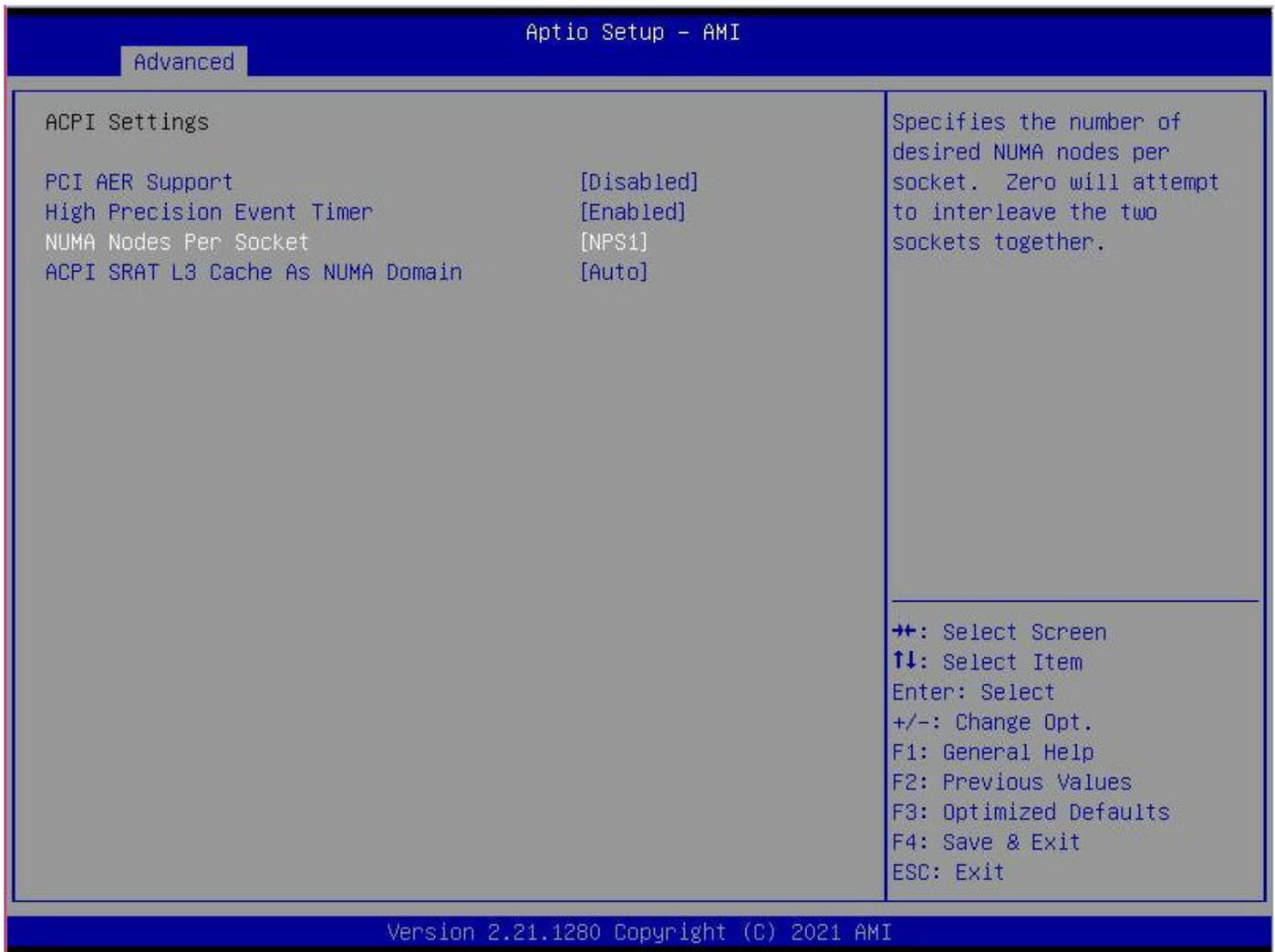
An Oracle Restart instance was installed and configured. An Oracle Restart instance is the most basic form of Oracle Real Application Clusters (RAC) with only one system with Oracle Grid Infrastructure and Oracle ASM file system.

- Oracle Restart instance
- Oracle Grid Infrastructure, release 19.3.0.0.0
- Oracle Database Enterprise Edition, release 19.3.0.0.0
- Oracle ASM File System
- Oracle Linux 8.3 (5.4.17-2102.201.3.el8uek.x86_64)
- 4 Datacenter Class 1.8 TB U2 PCI NVME SSD Disks (2 for Data + 2 Redo Logs)

Firmware/BIOS configuration

The test configuration was designed for maximum memory channel and interleaving efficiency on all 8 DIMMs, as shown in Figure 1: Firmware / Bios configuration.

NUMA Nodes Per Socket: NPS1 (because this is a single-socket system)



SMT Control: Auto (enabled), which allows AMD Simultaneous Multithreading to optimize resource utilization by executing multiple independent execution threads per CPU core.

The screenshot shows the 'Advanced' menu of the Aptio Setup - AMI BIOS. The 'CPU Configuration' section is expanded, showing various settings. The 'SMT Control' setting is highlighted in blue and set to '[Auto]'. Other settings include Core Performance Boost, Global C-state Control, Local APIC Mode, CCD Control, Core control, L1 Stream HW Prefetcher, L2 Stream HW Prefetcher, SMEE, SEV ASID Count, and SEV-ES ASID Space Limit Control, all set to '[Auto]'. The 'SVM Mode' is set to '[Enabled]'. Below the settings, there is a 'CPU1 Information' section. On the right side of the screen, there is a help text block explaining the navigation keys: '+' and '-' for selecting screens, up and down arrows for selecting items, Enter for selecting, +/- for changing options, F1 for general help, F2 for previous values, F3 for optimized defaults, F4 for save and exit, and ESC for exit. At the bottom of the screen, the version '2.21.1280' and copyright '© 2021 AMI' are displayed.

Setting	Value
CPU Configuration	
SMT Control	[Auto]
Core Performance Boost	[Auto]
Global C-state Control	[Auto]
Local APIC Mode	[Auto]
CCD Control	[Auto]
Core control	[Auto]
L1 Stream HW Prefetcher	[Auto]
L2 Stream HW Prefetcher	[Auto]
SMEE	[Auto]
SEV ASID Count	[Auto]
SEV-ES ASID Space Limit Control	[Auto]
SVM Mode	[Enabled]

Can be used to disable symmetric multithreading. To re-enable SMT, a POWER CYCLE is needed after selecting the 'Enable' option. Select 'Auto' base on BIOS PCD (PcdAmdSmtMode) default setting. WARNING - S3 is NOT SUPPORTED on systems where SMT is disabled.

++: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit

Version 2.21.1280 Copyright (C) 2021 AMI

Determinism Slider: Performance (to boost performance while also increasing power consumption)

The screenshot shows the 'Advanced' menu of the Aptio Setup - AMI BIOS. The 'Determinism Slider' is set to 'Performance'. The 'cTDP Control' is set to 'Auto'. The 'IDMMU' is set to 'Auto'. The 'Package Power Limit Control' is set to 'Auto'. The 'APBDIS' is set to 'Auto'. The 'DF Cstates' is set to 'Auto'. The 'Preferred IO' is set to 'Auto'. The 'Memory Configuration' and 'CPU1 Memory Information' are also visible.

Setting	Value
North Bridge Configuration	
Determinism Control	[Manual]
Determinism Slider	[Performance]
cTDP Control	[Auto]
IDMMU	[Auto]
Package Power Limit Control	[Auto]
APBDIS	[Auto]
DF Cstates	[Auto]
Preferred IO	[Auto]

▶ Memory Configuration
▶ CPU1 Memory Information

Auto = Use default performance determinism settings
Power
Performance

++: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit

Version 2.21.1280 Copyright (C) 2021 AMI

Memory Interleaving: Auto (all DIMM slots must be populated)

The screenshot shows the 'Advanced' menu of the Aptio Setup - AMI BIOS. The settings listed are:

Setting	Value	Description
Memory Clock	[3200 MHz]	Allows for disabling memory interleaving. Note that NUMA nodes per socket will be honored regardless of this setting.
Memory interleaving	[Auto]	
Memory interleaving size	[Auto]	
Chipselect Interleaving	[Auto]	
BankGroupSwap	[Auto]	
DRAM Scrub Time	[Auto]	
TSME	[Auto]	
DDR Power Down Enable	[Auto]	

Navigation instructions:

- ++: Select Screen
- ↑↓: Select Item
- Enter: Select
- +/-: Change Opt.
- F1: General Help
- F2: Previous Values
- F3: Optimized Defaults
- F4: Save & Exit
- ESC: Exit

Version 2.21.1280 Copyright (C) 2021 AMI

The following screenshot shows the memory slots



Competitive TCO

TCO is a crucial consideration when architecting a database solution. Oracle Processor Licensing is available for Oracle Database Enterprise Edition and is calculated by multiplying each processor's total number of cores by the appropriate licensing factor in the [Oracle Processor Core Factor Table](#). Oracle also charges an additional support fee. This license model allows customers to reduce license costs by selecting low-core CPUs with high frequencies, such as the AMD EPYC 74F3.

Oracle Database Standard Edition licensing uses a per-socket license model that requires an individual license for each socket. A processor is counted as equivalent to a socket; however, in the case of multi-chip modules, each chip in the multi-chip module is counted as one occupied socket. The single-socket Supermicro AS-1014S-WTRT is, therefore, an ideal cost-optimized solution for small to mid-size Oracle Database deployments.

Conclusion

The single-socket Supermicro AS-1014S-WTRT is an ideal cost-optimized solution for small to mid-size Oracle Database deployments. Figure 3 illustrates the increased generational performance between 2nd Gen and 3rd Gen AMD EPYC processors, which directly results from the improved frequency, IPC, and larger 32MB L3 cache available to any single core found in the 3rd Gen AMD EPYC Series processors.

Supermicro observed this generational performance improvement by comparing the 3rd Gen AMD EPYC 74F3 CPU to the 2nd Gen AMD EPYC 7F72 CPU. As a result, we believe that 3rd Gen AMD EPYC 74F3



processors show exceptional per-core and per-node performance across all load levels and offer compelling single-socket and dual-socket solutions to customers.

APPENDIX A – Oracle Linux & Configuration

Oracle Linux Installation & Configuration

Benchmark system OS version: Oracle Linux 8.3

Kernel version at the time the benchmarks were run: 5.4.17-2102.201.3.el8uek.x86_64

The benchmarks used the “Full” ISO image from the Oracle Linux Installation Media webpage located at:

<https://yum.oracle.com/oracle-linux-isos.html>

https://yum.oracle.com/ISOS/OracleLinux/OL8/u3/x86_64/OracleLinux-R8-U3-x86_64-dvd.iso

The Oracle Linux OS configuration was prepared by installing the “oracle-database-preinstall-19c” package, which checks and configures the required OS packages/parameters as needed. The installation was completed per the recommendations made by this tool. Please see the OS parameter changes made to “/etc/sysctl.conf” in the commands excerpt.

The Oracle ASM filesystem was selected for use with Oracle RAC standalone system (Oracle Restart instance), and the ASMLib driver was therefore installed and configured. ASM disks were on SSDs with single partitions.

ORAchk is a tool that checks many different Oracle software components. Scheduling regular ORAchk runs can help the database administrator see changes in the Oracle RAC configuration. We installed ORAchk to make a final check of the OS configuration, per [Autonomous Health Framework \(AHF\) - Including TFA and ORAchk/EXAchk \(Doc ID 2550798.1\)](#) (valid Oracle Support account required).

Per the ORACHK results:

- Set Linux Disk I/O Scheduler to Deadline.
- Set baLinux Transparent HugePages to Disable.

Configuring HugePages for SGA is especially important for buffer cache performance. Set this to an appropriate value in proportion to your RAM, as described in APPENDIX B - Oracle Database 19c Configuration. The System Global Area (SGA) is a group of shared memory structures called SGA components that contain data and control information for one Oracle Database. All server and background processes share the SGA.

OS parameters for Oracle Setup (/etc/sysctl.conf)

```
[root@master2 ~]# cat /etc/sysctl.conf
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
# For more information, see sysctl.conf(5) and sysctl.d(5).

# oracle-database-preinstall-19c setting for fs.file-max is 6815744
fs.file-max = 6815744
# oracle-database-preinstall-19c setting for kernel.sem is '250 32000 100 128'
kernel.sem = 250 32000 100 128
# oracle-database-preinstall-19c setting for kernel.shmmni is 4096
kernel.shmmni = 4096
# oracle-database-preinstall-19c setting for kernel.shmall is 1073741824 on x86_64
kernel.shmall = 1073741824
# oracle-database-preinstall-19c setting for kernel.shmmax is 4398046511104 on x86_64
kernel.shmmax = 4398046511104
# oracle-database-preinstall-19c setting for kernel.panic_on_oops is 1 per Orabug 19212317
kernel.panic_on_oops = 1
# oracle-database-preinstall-19c setting for net.core.rmem_default is 262144
net.core.rmem_default = 262144
# oracle-database-preinstall-19c setting for net.core.rmem_max is 4194304
net.core.rmem_max = 4194304
# oracle-database-preinstall-19c setting for net.core.wmem_default is 262144
net.core.wmem_default = 262144
# oracle-database-preinstall-19c setting for net.core.wmem_max is 1048576
net.core.wmem_max = 1048576
# oracle-database-preinstall-19c setting for net.ipv4.conf.all.rp_filter is 2
net.ipv4.conf.all.rp_filter = 2
# oracle-database-preinstall-19c setting for net.ipv4.conf.default.rp_filter is 2
net.ipv4.conf.default.rp_filter = 2
# oracle-database-preinstall-19c setting for fs.aio-max-nr is 1048576
fs.aio-max-nr = 1048576

# oracle-database-preinstall-19c setting for net.ipv4.ip_local_port_range is 9000 65500
net.ipv4.ip_local_port_range = 9000 65500
vm.nr_hugepages=225280
```

APPENDIX B - Oracle Database 19c Configuration

The Oracle Init.ora parameters used to run this benchmark with 512 GB memory were set as follows:

```
MILATPCC.__data_transfer_cache_size=0
MILATPCC.__db_cache_size=280G
MILATPCC.__large_pool_size=2684M
MILATPCC.__pga_aggregate_target=60G
MILATPCC.__sga_target=380G
MILATPCC.__shared_io_pool_size=0
MILATPCC.__shared_pool_size=50G
MILATPCC.__strams_pool_size=0
MILATPCC.__unified_pga_pool_size=0
MILATPCC.__inmemory_ext_roarea=0
MILATPCC.__inmemory_ext_rwarea=0
MILATPCC.__java_pool_size=1G
*.audit_file_dest='/u01/app/oracle/admin/MILATPCC/adump'
*.audit_trail='db'
*.compatible='19.0.0'
*.control_files='+DATAZ/MILATPCC/CONTROLFILE/Current.258.1073386059','+REDOZ/MILANTPCC/CONTROLFILE/Current.258.1073386059'
*.db_block_size=8192
*.db_create_file_dest='+DATA'
*.db_recovery_file_dest='+REDO'
*.db_recovery_file_dest_size=8256m
*.db_domain='supermicro.com'
*.db_name='MILATPCC'
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=MILATPCCXDB)'
*.processes=2048
*.remote_login_passwordfile='EXCLUSIVE'
*.open_cursors=300
*.undo_tablespace='UNDOTBS1'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.undo_retention=10
*.undo_management='AUTO'
*._undo_autotune=false
*.log_buffer=1G
*._rollback_segment_count=5000
*.parallel_min_servers=0
*.parallel_max_servers=0
*.parallel_servers_target=1792
*.db_writer_processes=8
*.sga_target=160G
*.pga_aggregate_target=110G
*.db_16k_cache_size=0G
*.disk_asynch_io=TRUE
*.inmemory_size=48g
*.lock_sga=TRUE
*.result_cache_max_size=32G
*.db_block_checking='FALSE'
*.db_block_checksum='FALSE'
*.DB_CACHE_ADVICE='ON'
*.db_file_multiblock_read_count=16
*.dml_locks=16384
*.fast_start_mttr_target=0
*.filesystemio_options='setall'
*.transactions_per_rollback_segment=1
*.log_checkpoint_interval=262144
*.log_checkpoint_timeout=900
*.log_checkpoints_to_alert=TRUE
*.plssql_code_type='NATIVE'
*.plssql_optimize_level=2
*.query_rewrite_enabled='FALSE'
*.replication_dependency_tracking=FALSE
*.statistics_level='TYPICAL'
*.timed_statistics='TRUE'
*.trace_enabled='FALSE'
```

HammerDB Installation and Configuration

Successful TNSping to the database system is required before testing HammerDB. Finish installing and configuring the Oracle client before installing HammerDB. These benchmarks used HammerDV v3.1, which can be found at <https://github.com/TPC-Council/HammerDB/releases>. Be sure to create a separate user called “hammerdb1” to run the HammerDB client.

Step 1: Preparing the Benchmark Database

Tablespace Creation: Use the database file destination values.

```
SQL> show parameter file_dest
```

NAME	TYPE	VALUE
db_create_file_dest	string	+DATA
db_recovery_file_dest	string	+REDO

Step 2: Create the benchmark database using the following SQL scripts:

```
CREATE BIGFILE TABLESPACE TPCC_800 DATAFILE SIZE 300G AUTOEXTEND ON NEXT 10G  
BLOCKSIZE 8K  
EXTENT MANAGEMENT LOCAL AUTOALLOCATE  
SEGMENT SPACE MANAGEMENT AUTO;
```

```
CREATE BIGFILE TABLESPACE TPCC_800_OL DATAFILE SIZE 500G AUTOEXTEND ON NEXT 10G  
BLOCKSIZE 8K  
EXTENT MANAGEMENT LOCAL AUTOALLOCATE  
SEGMENT SPACE MANAGEMENT AUTO;
```

Step 3: Create a temporary tablespace with a minimum file size of 30G:

Create Temporary Tablespace TPCCTEMP TEMPFILE '+DATAZ' size 30G Autoextend On Next 10M Maxsize Unlimited Extent Management Local Uniform Size 10M;

Step 4: Create Users and Grants

Create User tpcc800 Identified by tpcc800 default Tablespace Tocc_800 Temporary Tablespace TPCCTEMP; Create Temporary Tablespace TPCCTEMP TEMPFILE '+DATAZ' size 30G Autoextend on next 10M Maxsize Unlimited Extent Management Local Uniform Size 10M;

```
grant create session to tpcc800;
grant connect,resource to tpcc800;
```

```
grant dba to tpcc800;
GRANT CONNECT, RESOURCE, DBA to tpcc800;
GRANT UNLIMITED TABLESPACE TO tpcc800;
```

Step 5: Create the database schema using the HammerDB client.

The HammerDB CLI commands allow you to create the schema by logging in to the HammerDB client, changing to the HammerDB directory, and then executing the hammerdbcli command:

```
cd /hammerdb1/HammerDB-3.1
./hammerdbc1
```

Execute the following commands to set the run parameters:

```
dbset db ora
dbset bm TPC-C
diset connection system_password Amarit27
diset connection instance MILATPCC
diset tpcc tpcc_user tpcc800
diset tpcc tpcc_pass tpcc800
diset tpcc tpcc_def_temp TPCCTEMP
diset tpcc count_ware 256
diset tpcc num_vu 20
diset tpcc tpcc_def_tab tpcc_800
diset tpcc tpcc_ol_tab tpcc_800_ol
diset tpcc partition true
diset tpcc hash_clusters true
print dict
```

Execute the following command to create the schema and load benchmark data into the database:

```
buildschema
```

Prepare the system for running new benchmarks by destroying virtual users after building the schema and after every benchmark run: We used the following script to change the count_ware (Number of warehouses) and num_vu (Number of virtual users) to 0:

```
dbset db ora
diset tpcc ora_driver timed
dbset bm TPC-C
print bm
diset connection system_password Amarit27
diset connection instance milatpcc
diset tpcc count_ware 256
diset tpcc tpcc_def_tab tpcc_800
diset tpcc tpcc_ol_tab tpcc_800_ol
diset tpcc checkpoint true
diset tpcc rampup 3
diset tpcc duration 10
loadscript
vuset vu 300
vuset showoutput 1
vuset logtotemp 1
vuset unique 1
vucreate
print dict
print vuconf
print script
vurun
```

Footnotes

1. Based on AMD internal testing as of 02/1/2021, average performance improvement at ISO-frequency on an AMD EPYC™ 72F3 (8C/8T, 3.7GHz) compared to an AMD EPYC™ 7F32 (8C/8T, 3.7GHz), per-core, single thread, using a select set of workloads including SPECrate®2017_int_base, SPECrate®2017_fp_base, and representative server workloads. SPEC® and SPECrate® are registered trademarks of Standard Performance Evaluation Corporation. Learn more at [spec.org](https://www.hammerdb.com/docs/ch03s02.html). See also <https://www.hammerdb.com/docs/ch03s02.html>.
2. AMD Infinity Guard features vary by EPYC™ Processor generations. Infinity Guard security features must be enabled by server OEMs and/or Cloud Service Providers to operate. Check with your OEM or provider to confirm support of these features. Learn more about Infinity Guard at <https://www.amd.com/en/technologies/infinity-guard>.

- 
3. Performance was measured with Oracle Database 19c on Oracle Linux 8.3 using HammerDB version 3.1.
 4. Max boost for AMD EPYC processors is the maximum frequency achievable by any single core on the processor under normal operating conditions for server systems.

AMD, the AMD logo, EPYC, Infinity Fabric, and combinations thereof are trademarks of Advanced Micro Devices, Inc.