The science behind the report:

Make sense of important data faster with AWS EC2 M6i instances

This document describes what we tested, how we tested, and what we found. To learn how these facts translate into real-world benefits, read the report Make sense of important data faster with AWS EC2 M6i instances.

We concluded our hands-on testing on October 4, 2021. During testing, we determined the appropriate hardware and software configurations and applied updates as they became available. The results in this report reflect configurations that we finalized on August 31, 2021 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

To learn more about how we have calculated the wins in this report, go to http://facts.pt/calculating-and-highlighting-wins. Unless we state otherwise, we have followed the rules and principles we outline in that document.

While the TPROC-H workload sets the maximum number of streams for each database size—five simultaneous streams for the 100GB database —we wanted to see how m6i.16xlarge instance handled increased load. We increased the number of simultaneous streams to 8, 16, and 32. As Table 3 shows, the instance with the 3rd Gen Intel Xeon Scalable processor handled the load much better than the instance with the older processor, showing that the m6i.16xlarge instance can offer better scaling as the database load increases.

Table 1: Time required for the m6i.2xlarge and m5.2xlarge instances to complete a HammerDB TPROC-H workload on 10GB Microsoft SQL Server databases. All times are in seconds. Less time is better.

Stream count	m5.2xlarge instance	m6i.2xlarge instance
1 stream	16	12
2 streams	24	18
3 streams	31	25

Table 2: Time required for the m6i.4xlarge and m5.4xlarge instances to complete a HammerDB TPROC-H workload on 30GB Microsoft SQL Server databases. All times are in seconds. Less time is better.

Stream count	m5.4xlarge instance	m6i.4xlarge instance
1 stream	22	16
2 streams	35	27
3 streams	46	36
4 streams	59	46

Table 3: Time required for the m6i.16xlarge and m5.16xlarge instances to complete a HammerDB TPROC-H workload on 100GB Microsoft SQL Server databases. All times are in seconds. Less time is better.

Stream count	m5.16xlarge instance	m6i.16xlarge instance
1 stream	27	25
2 streams	42	37
3 streams	55	52
4 streams	66	61
5 streams	77	68
8 streams	113	96
16 streams	218	182
32 streams	450	368

Instance configuration information

Table 4: Detailed information on the 8vCPU instances we tested.

Instance configuration information	m5.2xlarge instance	m6i.2xlarge instance
General information		
Date testing ended	09/02/21	09/02/21
Cloud service provider (CSP)	AWS	AWS
Region	us-east-1F	us-east-1F
Workload information		
Workload name and version	HammerDB v4.2 TPROC-H	HammerDB v4.2 TPROC-H
Workload or software specific parameters	Up to 3 Virtual Users 8 MAXDOP 10 Scale Factor	Up to 3 Virtual Users 8 MAXDOP 10 Scale Factor
Iterations and result choice	3 test runs, median reported	3 test runs, median reported
Cloud VM or instance details		
Number of VMs	1	1
VM or instance size	m5.2xlarge	m6i.2xlarge
BIOS name and version	Amazon EC2 1.0	Amazon EC2 1.0
vCPU	8	8
Number of cores/threads	4 cores/8 threads	4 cores/8 threads
Memory (GB)	32	32
Underlying processor model	Intel® Xeon® Platinum 8175M	Intel Xeon Platinum 8375C
vNIC information/Underlying NIC speed	Amazon Elastic Network Adapter up to 10 Gigabit	Amazon Elastic Network Adapter up to 12.5 Gigabit
vNIC driver version	2.2.3.0	2.2.3.0
Other instance hardware or parameter details	Set Lock pages in memory; SQL max memory to 29,491 MB; SQL MAXDOP to 8; TPROC-H and TempDB on separate disk from OS	Set Lock pages in memory; SQL max memory to 29,491 MB; SQL MAXDOP to 8; TPROC-H and TempDB on separate disk from OS
Operating system information		
Image or template name and UUID	Microsoft Windows Server 2019	Microsoft Windows Server 2019
Operating system name	Microsoft Windows Server 2019 Datacenter	Microsoft Windows Server 2019 Datacenter
Operating system build number	Build 17763	Build 17763
Kernel version	1809	1809
Changes made from CSP image	Installed Microsoft SQL Server 2019	Installed Microsoft SQL Server 2019

Instance configuration information	m5.2xlarge instance	m6i.2xlarge instance
Instance storage (volume type 1)		
Number of volumes	1	1
Volume use in this test	Data drive	Data drive
CSP volume type	AWS-EBS io1	AWS-EBS io1
Volume size (GB)	100	100
IOPS requested	3,000	3,000
Encryption type	Not encrypted	Not encrypted
Instance storage (volume type 2)		
Number of volumes	1	1
Volume use in this test	OS	OS
CSP volume type	EBS-gp2	EBS-gp2
Volume size (GB)	50	50
IOPS requested	100	100
Encryption type	Not encrypted	Not encrypted

Table 5: Detailed information on the 16vCPU instances we tested.

Instance configuration information	m5.4xlarge instance	m6i.4xlarge instance
General information		
Date testing ended	09/03/21	09/03/21
Cloud service provider (CSP)	AWS	AWS
Region	us-east-1F	us-east-1F
Workload information		
Workload name and version	HammerDB v4.2 TPROC-H	HammerDB v4.2 TPROC-H
Workload or software specific parameters	4 Virtual Users 16 MAXDOP 30 Scale Factor	4 Virtual Users 16 MAXDOP 30 Scale Factor
Iterations and result choice	3 test runs, median reported	3 test runs, median reported
Cloud VM or instance details		
Number of VMs	1	1
VM or instance size	m5.4xlarge	m6i.4xlarge
BIOS name and version	Amazon EC2 1.0	Amazon EC2 1.0
vCPU	16	16
Number of cores/threads	8 cores/16 threads	8 cores/16 threads
Memory (GB)	64	64
Underlying processor model	Intel Xeon Platinum 8259CL	Intel Xeon Platinum 8370C
vNIC information/Underlying NIC speed	Amazon Elastic Network Adapter up to 10 Gbps	Amazon Elastic Network Adapter up to 12.5 Gbps
vNIC driver version	2.2.3.0	N/A
Other instance hardware or parameter details	Set Lock pages in memory; SQL max memory to 58,982 MB; SQL MAXDOP to 16; TPROC-H and TempDB on separate disk from OS.	Set Lock pages in memory; SQL max memory to 58,982 MB; SQL MAXDOP to 16; TPROC-H and TempDB on separate disk from OS.
Operating system information		
Image or template name and UUID	Microsoft Windows Server 2019	Microsoft Windows Server 2019
Operating system name	Microsoft Windows Server 2019 Datacenter	Microsoft Windows Server 2019 Datacenter
Operating system build number	Build 17763	Build 17763
Kernel version	1809	1809
Changes made from CSP image	Installed Microsoft SQL Server 2019	Installed Microsoft SQL Server 2019

Instance configuration information	m5.4xlarge instance	m6i.4xlarge instance
Instance storage (volume type 1)		
Number of volumes	1	1
Volume use in this test	Data drive	Data drive
CSP volume type	AWS-EBS io1	AWS-EBS io1
Volume size (GB)	120	120
IOPS requested	3,000	3,000
Encryption type	Not encrypted	Not encrypted
Instance storage (volume type 2)		
Number of volumes	1	1
Volume use in this test	OS	OS
CSP volume type	EBS-gp2	EBS-gp2
Volume size (GB)	50	50
IOPS requested	100	100
Encryption type	Not encrypted	Not encrypted

Table 6: Detailed information on the 64vCPU instances we tested.

Instance configuration information	m5.16xlarge instance	m6i.16xlarge instance
General information		
Date testing ended	09/30/21	09/30/21
Cloud service provider (CSP)	AWS	AWS
Region	us-east-1F	us-east-1F
Workload information		
Workload name and version	HammerDB v4.2 TPROC-H	HammerDB v4.2 TPROC-H
Workload or software specific parameters	Up to 64 Virtual Users 64 MAXDOP 100 Scale Factor	Up to 64 Virtual Users 64 MAXDOP 100 Scale Factor
Iterations and result choice	3 test runs, median reported	3 test runs, median reported
Cloud VM or instance details		
Number of VMs	1	1
VM or instance size	m5.16xlarge	m6i.16xlarge
BIOS name and version	Amazon EC2 1.0	Amazon EC2 1.0
vCPU	64	64
Number of cores/threads	32 cores/64 threads	32 cores/64 threads
Memory (GB)	256	256
Underlying processor model	Intel Xeon Platinum 8259CL	Intel Xeon Platinum 8375C
vNIC information/Underlying NIC speed	Amazon Elastic Network Adapter up to 20 Gbps	Amazon Elastic Network Adapter up to 25 Gbps
vNIC driver version	2.2.3.0	2.2.3.0
Other instance hardware or parameter details	Set Lock pages in memory; SQL max memory to 235,929.6MB; SQL MAXDOP to 64; TPROC-H and TempDB on separate disk from OS	Set Lock pages in memory; SQL max memory to 235,929.6MB; SQL MAXDOP to 64; TPROC-H and TempDB on separate disk from OS
Operating system information		
Image or template name and UUID	Microsoft Windows Server 2019	Microsoft Windows Server 2019
Operating system name	Microsoft Windows Server 2019 Datacenter	Microsoft Windows Server 2019 Datacenter
Operating system build number	Build 17763	Build 17763
Kernel version	1809	1809
Changes made from CSP image	Installed Microsoft SQL Server 2019	Installed Microsoft SQL Server 2019

Instance configuration information	m5.16xlarge instance	m6i.16xlarge instance
Instance storage (volume type 1)		
Number of volumes	1	1
Volume use in this test	Data drive	Data drive
CSP volume type	AWS-EBS io1	AWS-EBS io1
Volume size (GB)	150	150
IOPS requested	6,000	6,000
Encryption type	Not encrypted	Not encrypted
Instance storage (volume type 2)		
Number of volumes	1	1
Volume use in this test	OS	OS
CSP volume type	EBS-gp2	EBS-gp2
Volume size (GB)	50	50
IOPS requested	100	100
Encryption type	Not encrypted	Not encrypted

How we tested

Testing overview

We tested AWS instances featuring older Intel processors and compared their data analytics performance to instances featuring 3rd Gen Intel Xeon Scalable (Ice Lake) versions. We ran a TPROC-H workload on Microsoft SQL Server to show the time savings for analytic databases that customers can expect to see using the newer instance series vs. the older.

Creating the Windows Server 2019 baseline image

Creating the baseline image instance

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Launch instance, and from the drop-down menu, select Launch instance.
4. In the search window, type `Windows Server` and press Enter.
5. On the Quick Start tab, next to Microsoft Windows Server 2019 Base, click Select.
6. On the Choose Instance Type tab, select t2.micro, and click Next: Configure Instance Details.
7. On the Configure Instance tab, set the following:
   - Number of instances: 1
   - Purchasing option: Leave unchecked
   - Network: Default VPC.
   - Subnet: Choose the region you are working in.
   - Auto-assign Public IP: Enable.
   - Placement Group: Leave unchecked.
   - Capacity Reservation: Open
   - Domain join directory: No Directory
   - IAM role: None
   - Shutdown behavior: Stop
8. Click Next: Add Storage.
9. On the Add Storage tab, set the following:
   - Size: 50GB
   - Volume Type: gp2
   - Delete on Termination: Checked
   - Encryption: Not Encrypted
10. Click Next: Add Tags.
11. On the Add Tags tab, add any tags you wish to use. Click Next: Configure Security Group.
12. On the Configure Security Group tab, leave defaults, and click Review and Launch.
13. On the Review Tab, click Launch.
14. Choose the appropriate option for the key pair, and click Launch Instances.

Configuring Windows Server 2019

1. Open Server Manager, and click Local Server.
2. Disable IE Enhanced Security Configuration.
3. Change the time zone to your local time zone.
4. Change the name of your server, and reboot when prompted.
5. Open Server Manager again, and click Local Server.
6. Click Run updates.
7. Run updates, rebooting when prompted, until the server shows no new updates to install.

Installing SQL Server 2019 Enterprise

1. Download or copy the ISO to the server, and unzip it.
2. Double-click the Setup application.
3. Click Installation→New SQL Server Standalone installation or add features to an existing installation.
4. Choose the trial version, and click Next.
5. Check the I accept the license terms and Privacy Statement box, and click Next.
6. Check the Use Microsoft Update to check for updates (recommended) box, and click Next.
7. On the Install Rules page, click Next.
8. Check the boxes for the following features, and click Next:
   - Database Engine Services
   - Full-Test and Semantic Extractions for Search
   - Client Tools Connectivity
   - Client Tools Backwards Compatibility
9. Leave the Default instance, and click Next.
10. Leave the default Service Accounts, and click Next.
11. On the Server Configuration tab, choose Mixed Mode, and enter and confirm a Password for the SQL Server system administrator (sa) account.
12. Click Add Current User to Specify the SQL Server administrators.
13. Click Next.
14. Once you've passed the rule check, click Next.
15. Click Install.
16. When the installation is finished, go back to the SQL Server Installation Center, and click Install SQL Server Management Tools.
17. Download the SSMS file, and install with defaults.
18. Reboot the server when prompted.
19. Run Windows Update one more time to ensure there aren't any new updates for SQL (make sure Windows Updates are set to get updates for other Microsoft products).
20. Once you've installed all available updates, disable Windows Update service. Click Start, type `services` and disable the Windows Update service.

Locking pages in memory

1. Click Start, and type `Local Security Policy`. Open the program when it pops up in the search.
2. Expand Local Policies, and click User Rights Assignment.
3. In the right-hand pane, scroll down, and double-click Lock pages in memory.
4. Click Add User or Group, type `NT Service\MSSQLSERVER`, and click OK.
5. Click OK to close the Properties window, and close the Local Security Policy window.

Installing HammerDB 4.2

1. Download Hammer DB from https://hammerdb.com/download.html.
2. Double-click the .exe file, choose English, and click OK.
3. Click Yes.
4. Click Next.
5. Chose a destination location, and click Next.
6. Click Next.
7. Click Finish.

Creating an AMI of the baseline instance

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Running instances.
4. Place a checkmark next to the instance you wish to create an image from.
5. Click the Action drop-down, and select Image→Create Image.
6. Enter the Image name, and click Create Image.
7. In the menu on the left side of the page, navigate to Images→AMIs.

Creating an instance with the baseline image

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Images→AMIs.
4. Check the box next to the image you created in the previous step, and click Launch.
5. On the Choose Instance Type tab, select your VM size, and click Next: Configure Instance Details.
6. On the Configure Instance tab, set the following:
   - Number of instances: 1
   - Purchasing option: Leave unchecked.
   - Network: Default VPC.
   - Subnet: Choose the region you are working in.
   - Auto-assign Public IP: Enable.
   - Placement Group: Leave unchecked.
   - Capacity Reservation: Open
   - Domain join directory: No Directory
   - IAM role: None
   - Shutdown behavior: Stop
7. Click Next: Add Storage.
8. On the Add Storage tab, set the following:
   - Volume Type: Set your volume type. We chose io1.
   - Size: Set your size. We chose 50GB.
   - Delete on Termination: Unchecked.
   - Encryption: Not Encrypted.
9. Click Next: Add Tags
10. On the Add Tags tab, add any tags you wish to use.
11. Click Next: Configure Security Group.
12. On the Configure Security Group tab, leave the defaults, and click Review and Launch.
13. On the Review Tab, click Launch.
14. Choose the appropriate option for the key pair, and click Launch Instances.

Configuring SQL Server on the instances under test

In this section, we list the various SQL Server settings that we changed and the steps to do so.

Set the SQL Server memory reserve and max degree of parallelism (MAXDOP)

1. Open the SQL Server Management Studio.
2. Right-click the SQL Instance, and click Properties.
3. Click Advanced node, and scroll down to the Max Degree of Parallelism and change the value. We set our MAXDOP on each VM instance to match the instance's number of VCPUs (8, 16, 64). Click OK.
4. Right-click the SQL Instance, and go to Memory.
5. Set the Max Memory to 90% of the total memory in the system. Click OK, and close the Properties window.
6. Right-click the SQL instance, and restart the service. When prompted, click Yes.

Configuring the tempdb database

1. Open the SQL Server Management Studio.
2. Expand Databases and System databases, and right-click tempdb.
3. Add files and change the starting size as necessary. We added 7 files for 8 total files, and resized the files to 1,024 MB each.
4. Right-click the SQL instance, and restart the service. When prompted, click Yes.
5. To move the tempdb to the database drive, open a new Query and run the following modified for the number of tempdb files your system has

```
USE [master]
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = tempdev , FILENAME = 'E:\TempDB\tempdb.mdf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp2 , FILENAME = 'E:\TempDB\tempdb_mssql_2.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp3 , FILENAME = 'E:\TempDB\tempdb_mssql_3.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp4 , FILENAME = 'E:\TempDB\tempdb_mssql_4.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp5 , FILENAME = 'E:\TempDB\tempdb_mssql_5.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp6 , FILENAME = 'E:\TempDB\tempdb_mssql_6.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp7 , FILENAME = 'E:\TempDB\tempdb_mssql_7.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp8 , FILENAME = 'E:\TempDB\tempdb_mssql_8.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = templog , FILENAME = 'E:\TempDB\templog.ldf' )
GO
```

6. Right click the SQL instance, and restart the service. When prompted, click Yes.

Running the tests

In this section, we list the steps to run the HammerDB TPROC-H test on the instances under test. For the maximum number of users we ran, we followed HammerDB TPC-H recommendations for the size database we were testing. Additionally, to show the scaling of each instance pair, we ran with fewer users. Note that before each test, we first ran a single-stream test to cache the database into memory before running the test with the actual number of streams required for the run.

1. On the instance under test, restore the database under test to so that the database and log files reside on the io1 SSD.
2. Open SQL Server Management Studio.
3. Right-click Databases→New Database.
4. Name the database. We named ours `tpch`.
5. To add seven more database files for a total of eight, click Add.
6. Name the database files, and click OK.
7. Ensure your SQL settings and tempdb are configured properly according to the instructions above and the instance you're running on.
8. Open HammerDB.
9. Select Options→Benchmark.
10. Choose MSSQL Server and TPC-H.
11. Expand SQL Server→TPC-H→Schema Build.
12. Double-click Options, change the driver to ODBC Driver 17 for SQL Server, set the scale to match your database, set MAXDOP to match SQLs, and check the box for Clustered Columnstore. Click OK.
13. Double-click Build. This build could take several hours. Once the database build is complete, reboot.
14. Once the database reboots, open HammerDB.
15. Select Options→BenchMark.
16. Choose MSSQL Server and TPC-H.
17. Expand Schema Build and double-click Options. Set the scale to match your database, set MAXDOP to match SQLs. Click OK.
18. Expand Driver Script, double-click Options, and click OK.
19. Expand Virtual User, and double-click Options.
20. Choose 1 user.

21. Check the boxes for Show Output, Log Output to Temp, and Use Unique Log Name.
22. Click OK.
23. Double-click Create users.
24. To capture performance metrics on the system, start Performance monitor set to record CPU, Memory, and drive usage information.
25. To begin the run, click Start.
26. When the run finishes, stop HammerDB and Perfmon, and save the HammerDB results file and Perfmon output.
27. Under Virtual User, double-click Options, and set the number of users to the appropriate count for the multi-stream test.
28. Double-click Create users.
29. To capture performance metrics on the system, start Performance monitor and set it to record CPU, Memory, and drive usage information.
30. To begin the run, click Start.
31. When the run finishes, stop Perfmon, and save the HammerDB results file and Perfmon output.
32. Reboot the instance.
33. Repeat the test two more times for a total of three runs at each user count, and report the median run.

Read the report at http://facts.pt/RWuDLHo ▶

This project was commissioned by Intel.

Principled Technologies®

Facts matter.®