A Principled Technologies report: Hands-on testing. Real-world results.

Unleash the power to create with the faster HP Z4 G5 Workstation Desktop PC

Get stronger performance vs. a previous-generation HP Z4 G4 Workstation PC

When your team has demanding tasks to complete—from photo or video editing, to 3D modeling, to AI-powered workflows—outdated laptops or workstations could be extending the time it takes to complete projects. By arming your creators and developers with powerful HP Z4 G5 workstations with Intel® Xeon® w3-2425 processors, you can help them unleash their creative potential and move on to their next big ideas.

At Principled Technologies, we compared the new HP Z4 G5 Workstation Desktop PC to its predecessor, the HP Z4 G4, on various AI tasks including medical imaging and language processing, as well as general productivity tasks, CPU and GPU performance, 3D modeling tasks, and creative workflows in Adobe® Creative Cloud®. Packed with the latest technology, including a new Intel Xeon w3-2425 processor, the HP Z4 G5 sped up content creation, increased everyday performance, and handled AI tasks faster than the older model.

Our results show that any teams that create content, complete 3D building designs, or use AI for medical imaging could likely see a boost in performance by investing in the new HP Z4 G5 Workstation Desktop PC.

Move projects to completion faster
with 23% better performance in PugetBench for After Effects

Finish 3D models faster
with 31% better performance in Maxon Redshift

Inference faster with neural networks
with 38% better performance in ResNet-50

Unleash the power to create with the faster HP Z4 G5 Workstation Desktop PC August 2024

The science behind the report:

Unleash the power to create with the faster HP Z4 G5 Workstation Desktop PC

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 Unleash the power to create with the faster HP Z4 G5 Workstation Desktop PC.

We concluded our hands-on testing on July 3, 2024. 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 June 4, 2024 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.

Table 1: Results of our testing.

Performance benchmarks	HP Z4 G4 Workstation Desktop PC	HP Z4 G5 Workstation Desktop PC	Comp win %
Cinebench 2024			
CPU Multi Core - Median (higher is better)	469.93	620.31	**32.00%**
CPU Single Core - Median (higher is better)	68	88	**29.41%**
Blender Benchmark v4.1.0 GPU Rendering			
Monster - Samples per Minute - Median (higher is better)	1441.27	2351.92	**63.18%**
Junkshop - Samples per Minute - Median (higher is better)	874.89	1108.03	**26.65%**
Classroom - Samples per Minute - Median (higher is better)	783.30	1176.67	**50.22%**
CrossMark Pro			
Overall Score - Median Run (higher is better)	1275	1344	**5.41%**
Productivity (higher is better)	1279	1297	**1.41%**
Creativity (higher is better)	1284	1491	**16.12%**
Responsiveness (higher is better)	1221	1097	**-10.16%**

Performance benchmarks	HP Z4 G4 Workstation Desktop PC	HP Z4 G5 Workstation Desktop PC	Comp win %
PugetBench for After Effects using Adobe After Effects			
Overall Score - Median (higher is better)	692	854	**23.41%**
PugetBench for Photoshop using Adobe Photoshop			
Overall Score - Median (higher is better)	5348	6100	**14.06%**
PugetBench for Photoshop using Adobe Photoshop			
Overall Score (Standard) - Median (higher is better)	6952	8014	**15.28%**
Maxon Redshift			
Render Time (hh:mm:ss) - Median (lower is better)	0:03:59	0:02:44	**45.73%**
ResNet50			
Queries Per Second - Median (higher is better)	9619.65	13335.1	**38.62%**
3D uNet			
Queries Per Second - Median (higher is better)	1.066	1.297	**21.67%**
RNNT r4.0			
Queries Per Second - Median (higher is better)	797.091	1152.03	**44.53%**
BERT 99			
Queries Per Second - Median (higher is better)	117.688	121.444	**3.19%**

System configuration information

Table 2: Detailed information on the systems we tested.

System configuration information	HP Z4 G4 Workstation Desktop PC	HP Z4 G5 Workstation Desktop PC
Processor		
Vendor	Intel®	Intel
Model number	Xeon® W-2235	Xeon w3-2425
Core frequency (GHz)	3.80	3.00
Number of cores	6	6
Cache (MB)	8.25	15
Memory module(s)		
Amount (GB)	64	64
Type	Samsung® DDR4	Hynix® DDR5
Speed (MHz)	2,933	5,600
Discrete graphics		
Number of cards	1	1
Vendor	NVIDIA®	NVIDIA
Model number	NVIDIA RTX A4000	NVIDIA RTX 4000 Ada
VRAM	16 GB GDDR6	20 GB GDDR6
Storage		
Amount (TB)	1	1
Type	SSD	SSD
Connectivity/expansion		
Wired internet	Intel I219-LM	Intel I219-LM
Wireless internet	N/A	N/A
Bluetooth	5.3	5.3
USB	6 USB 3.1 Gen	4 USB Type-A 5Gbps
Thunderbolt	N/A	N/A
Video	4x DisplayPort 1.4	4x DisplayPort 1.4a
Operating system		
Vendor	Microsoft	Microsoft
Name	Windows 11	Windows 11
Build number or version	23H2	23H2
BIOS		
BIOS name and version	P61 v02.92	U61 Ver. 01.02.01

System configuration information	HP Z4 G4 Workstation Desktop PC	HP Z4 G5 Workstation Desktop PC
Dimensions		
Height (inches)	15.20	15.20
Width (inches)	17.5	17.50
Depth (inches)	6.65	6.65
Weight (pounds)	22.4	23.2

How we tested

Setting up the systems to test performance with benchmarks

Setting up and updating the OEM image

1. Boot the system.
2. To complete installation, follow the onscreen instructions, using the default selections when appropriate.
3. Set the Windows Power Plan to Ultimate Performance.
4. Set Screen and Sleep options to Never:
   - Right-click the desktop, and select Display settings.
   - Select System from the left column.
   - Click Power.
   - For all power options listed under Screen and Sleep, select Never.
5. Disable User Account Control notifications:
   - Select Windows Start, type UAC, and press Enter.
   - Move the slider control to Never notify, and click OK.
6. Run Windows Update, and install all updates available.
7. Launch run the HP Support Assistant utility app installed on each system, and update any drivers or BIOS files.
8. Boot to system BIOS, and select the following settings:
   - Click Advanced→Performance Management→Performance Control and set it to High Performance.
9. Download and install the latest NVIDIA drivers from https://www.nvidia.com/download/index.aspx.
10. Verify the date and time are correct and synchronize the system clock with the time server.
11. Pause Automatic Windows Updates:
    - Click Windows Start.
    - Type Windows Update settings, and press Enter.
    - From the Pause updates drop-down menu, select Pause for 5 weeks.

Capturing an image

1. Connect an external HDD to the system.
2. Click Windows Menu, and type Control Panel in the search bar. Click Control Panel→System, Security→Backup and Restore (Windows 7)→Create a system image.
3. Verify that the external HDD is selected as the save drive, and click Next.
4. Verify that all drives are selected to back up, and click Next.
5. Click Start backup.
6. When asked if you want to create a system repair disc, select No, and close the dialogs.

Restoring an image

1. Connect an external HDD to the system.
2. Press and hold the Shift key while restarting the system.
3. Select Troubleshoot.
4. Select Advanced options.
5. Select See more recovery options.
6. Select System image recovery.
7. Select the User account.
8. Enter the system password, and click Continue.
9. At the Restore system files and settings screen, select Next.
10. Verify that the external HDD is selected, and click Next.
11. Once the recovery has completed, click Finish.

Testing performance with Cinebench 2024

Setting up the test

1. Download and install Cinebench from https://www.maxon.net/en/downloads/cinebench-2024-downloads.

Running the test

1. Launch Cinebench.
2. Select File→Advanced benchmark.
3. Set the Minimum Test Duration to Off.
4. Select either CPU (Multi Core) or CPU (Single Core), and click Start.
5. Record the result.
6. Wait 15 minutes before re-running.
7. Repeat steps 1 through 6 two more times and record the median result.

Testing performance with Blender

Setting up the test

1. Download the Blender benchmark from https://opendata.blender.org/.

Running the test

1. Launch the Blender benchmark.
2. At the Welcome screen, click Next.
3. Select Blender version 4.1.0, and click Next.
4. At the Benchmark Scenes screen, click Next.
5. At the Benchmark Device screen, select the GPU option, and click Start Benchmark.
6. Record the results.
7. Wait 15 minutes before performing the next run.
8. Repeat steps 1 through 7 two more times and record the median result.

Testing performance with Maxon Redshift

Setting up the test

1. Download and install the Maxon app from https://www.maxon.net/en/try.
2. Launch the Maxon app.
3. Click the Maxon One 14-day trial, and download and install the Redshift application.

Running the test

1. To open a command prompt, click the Windows icon, type cmd, and press Enter.
2. Inside the command prompt, type cd C:\ProgramData\Redshift\bin, and press Enter.
3. To start the benchmark, type RunBunchmark.bat, and press Enter.
4. Record the results.
5. Wait 5 minutes before rerunning the test.
6. Repeat steps 1 through 5 two more times and record the median result.

BAPCo CrossMark benchmark testing

Setting up the test

1. Download and install CrossMark from the Microsoft Store or Apple App Store.

Running the test

1. Launch CrossMark.
2. Click Settings.
3. For Number of Iterations, choose 1.
4. Enter a valid email address, and click Back.
5. Click Run Benchmark.
6. Record the result.
7. Repeat steps 1 through 6 two more times.

PugetBench for Photoshop testing

We used the following applications:

- Adobe Photoshop
- PugetBench for Creators

Setting up the test

1. Launch Adobe Photoshop.
2. Click through the Tutorial pop-up tips.
3. Close Photoshop.
4. Purchase and download the PugetBench for Photoshop license from https://www.pugetsystems.com/pugetbench/creators/.
5. Click the Download PugetBench for Creators on Windows link.
6. After the download completes, double-click the installation file to install PugetBench.
7. Enter the license key in the license field. Click Activate.
8. Click Download Assets.

Running the test

1. Boot the system.
2. Open PugetBench for Creators.
3. Select the Photoshop test on the left side of the app.
4. Click Start Test.
5. When the benchmark finishes, record the overall score.
6. Close PugetBench for Creators and restart the system under test.
7. Wait 30 minutes before performing the next run.
8. Repeat steps 1 through 7 twice more and record the median result.

PugetBench for Premiere Pro testing

We used the following applications:

- Adobe Premiere Pro
- PugetBench for Creators

Setting up the test (Windows)

1. Launch Adobe Premiere Pro.
2. Click through the Tutorial pop-up tips.
3. Close Adobe Premiere Pro.
4. Purchase a PugetBench for Premiere Pro license from https://www.pugetsystems.com/pugetbench/creators/.
5. Click the Download PugetBench for Creators on Windows.
6. After the download completes, double-click the installation file to install PugetBench.
7. Enter the license key in the license field. Click Activate.
8. Click Download Assets.

Running the test

1. Boot the system.
2. Open PugetBench for Creators.
3. Select the Photoshop test on the left side of the app.
4. Click Start Test.
5. When the benchmark finishes, record the overall score.
6. Close PugetBench for Creators, and restart the system under test.
7. Wait 30 minutes before performing the next run.
8. Repeat steps 1 through 7 twice more and record the median result.

PugetBench for After Effects

We used the following application:

- Adobe After Effects

Setting up the test

1. Launch Adobe After Effects.
2. Click through the Tutorial pop-up tips.
3. Close Adobe After Effects.
4. Purchase and download the PugetBench for After Effects license from https://www.pugetsystems.com/labs/articles/PugetBench-for-Adobe-Creative-Cloud-1642/.
5. Click Get on Adobe Marketplace and log into your Adobe account.
6. Click the PugetBench After Effects Plug-in to install it.
7. Open Adobe After Effects.
8. Click Edit→Preferences→Scripting & Expressions.
9. Select Allow Scripts to Write Files and Access Network, and click OK.
10. Click Edit→Preferences→Memory & Performance.
11. The benchmark requires After Effects to have 12 GB of RAM space. Adjust the RAM reserved for other applications to get to 12 GB RAM reserved for After Effects, and click OK.
12. Click Composition→New Composition.
13. Accept the default Composition Settings, and click Ok.
14. Click Composition→Preview, and uncheck Cache Frames When Idle.
15. Click Window→Extensions→PugetBench for After Effects.
16. Next to License Key, click Change and enter your license key. Click Save.

Running the test

1. Open Adobe After Effects.
2. Click Window→Extensions→PugetBench for After Effects.
3. Click Run Benchmark.
4. Record the overall score when the benchmark finishes.
5. Close Adobe Photoshop and restart the system under test.
6. Wait 30 minutes before performing the next run.
7. Repeat steps 1 through 6 two more times.

Setting up the systems to run machine learning workloads

Configuring Ubuntu 22.04 on Windows 11 Windows Subsystem for Linux 2 (WSL 2)

1. In the system BIOS, confirm hardware virtualization is enabled.
2. Enable Hyper-V and Virtual Machine Platform in Windows:
   a. Open an elevated PowerShell terminal.
   b. Run the following commands, but decline the reboot until the final step:

```
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V-All
Enable-WindowsOptionalFeature -Online -FeatureName VirtualMachinePlatform
shutdown /r /t 0
```

3. Download and install drivers from NVIDIA: https://www.nvidia.com/download/index.aspx
    - If a professional GPU is installed, in the NVIDIA Control Panel, enable Error Correction Code.
4. To confirm GPU configuration, from an elevated terminal session, run nvidia-smi to list the GPUs, driver versions, and API versions, and verify that the system recognizes them.
5. Verify ECC is enabled:

```
nvidia-smi -q -d ECC.
```

6. Install Ubuntu 22.04:

```
wsl.exe --install Ubuntu-22.04.
```

7. Follow the prompts to create an Ubuntu user name and password, and exit the Ubuntu session.
8. Set default WSL instance to the new installation:

```
wsl --set-default Ubuntu-22.04.
```

9. Update WSL 2 to latest release:

```
wsl --update --pre-release.
```

10. Reboot the system:

```
shutdown /r /t 0
```

11. Open a new Terminal session to Ubuntu 22.04.
12. Configure path:

```
sudo nano /etc/profile
```

13. Add to the end of the file:

```
  export PATH="/home/ptuser/.local/bin:$PATH"
```

14. Update Ubuntu:

```
sudo apt update && sudo apt upgrade.
```

15. Exit Ubuntu:

```
exit
```

Configuring a Collective Mind (CM) machine learning environment

1. Reconnect to your Ubuntu-22.04 WSL instance:

```
wsl --distribution Ubuntu-22.04
```

2. Install prerequisites:

```
sudo apt install python3 python3-pip python3-venv git wget curl zlib1g unzip -y
```

3. Install CM:

```
python3 -m pip install cmind
```

4. Test CM:

```
cm test core
```

5. Use CM to pull the MLCommons GitHub repository.

Docker setup and config

1. Remove any previous Docker packages:

```
for pkg in docker.io docker-doc docker-compose docker-compose-v2 podman-docker containerd runc; do
sudo apt-get remove $pkg; done
sudo apt update
sudo apt install ca-certificates curl -y
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

2. Add the repo to Apt sources:

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://
download.docker.com/linux/ubuntu \ $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | sudo
tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update && sudo apt install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin -y
```

3. Test Docker installation:

```
sudo docker run hello-world
sudo usermod -aG docker $USER
newgrp docker
```

4. Test docker user's permissions:

```
    docker run hello-world
```

5. Enable Docker service:

```
sudo systemctl enable docker.service
sudo systemctl enable containerd.service
```

Build Docker container and run

```
cm pull repo gateoverflow@cm4mlops
cm docker script --tags=build,nvidia,inference,server --docker_cache=no --docker_cm_
repo=gateoverflow@cm4mlops
```

Running machine learning testing scripts

We ran scripts to execute the machine learning workloads and test performance once the container was finished building. We provide those scripts below.

3d-unet-99

```
cm run script --tags=run-mlperf,inference,_performance-only,_full  \
   --division=open \
   --category=edge \
   --device=cuda \
   --model=3d-unet-99 \
   --precision=float32 \
   --implementation=nvidia \
   --backend=tensorrt \
   --scenario=Offline \
   --execution_mode=valid \
   --power=no \
   --adr.python.version_min=3.8 \
   --clean \
   --compliance=no \
   --quiet \
   --time
```

bert-99

```
cm run script --tags=run-mlperf,inference,_performance-only,_full  \
   --division=open \
   --category=edge \
   --device=cuda \
   --model=bert-99 \
   --precision=float32 \
   --implementation=nvidia \
   --backend=tensorrt \
   --scenario=Offline \
   --execution_mode=valid \
   --power=no \
   --adr.python.version_min=3.8 \
   --clean \
   --compliance=no \
   --quiet \
   --time
resnet-50
cm run script --tags=run-mlperf,inference,_performance-only,_full  \
   --division=open \
   --category=edge \
   --device=cuda \
   --model=resnet50 \
   --precision=float32 \
   --implementation=nvidia \
   --backend=tensorrt \
   --scenario=Offline \
   --execution_mode=valid \
   --power=no \
   --adr.python.version_min=3.8 \
   --clean \
   --compliance=no \
   --quiet \
   --time
```

rnnt

```
cm run script --tags=run-mlperf,inference,_r4.0,_performance-only,_full  \
   --division=open \
   --category=edge \
   --device=cuda \
   --model=rnnt \
   --precision=float32 \
   --implementation=nvidia \
   --backend=tensorrt \
   --scenario=Offline \
   --execution_mode=valid \
   --power=no \
   --adr.python.version_min=3.8 \
   --clean \
   --compliance=no \
   --quiet \
   --time
```

Read the report at https://facts.pt/mBqMeZ6 ▶

This project was commissioned by HP.

Principled Technologies®

Facts matter.®

DISCLAIMER OF WARRANTIES; LIMITATION OF LIABILITY:
Principled Technologies, Inc. has made reasonable efforts to ensure the accuracy and validity of its testing, however, Principled Technologies, Inc. specifically disclaims any warranty, expressed or implied, relating to the test results and analysis, their accuracy, completeness or quality, including any implied warranty of fitness for any particular purpose. All persons or entities relying on the results of any testing do so at their own risk, and agree that Principled Technologies, Inc., its employees and its subcontractors shall have no liability whatsoever from any claim of loss or damage on account of any alleged error or defect in any testing procedure or result.

In no event shall Principled Technologies, Inc. be liable for indirect, special, incidental, or consequential damages in connection with its testing, even if advised of the possibility of such damages. In no event shall Principled Technologies, Inc.'s liability, including for direct damages, exceed the amounts paid in connection with Principled Technologies, Inc.'s testing. Customer's sole and exclusive remedies are as set forth herein.