VMware Private AI Foundation with NVIDIA

Sizing Guide

VMware Cloud Foundation 9.0

September, 2025

Table of Contents

Executive Summary

Artificial Intelligence (AI) has become a cornerstone of digital transformation across industries. The evolution of AI has taken a significant leap forward with the emergence of Generative AI (Gen AI).

Broadcom and NVIDIA offer a joint AI platform, called **VMware Private AI Foundation with NVIDIA**. By combining innovations from both companies, Broadcom and NVIDIA aim to unlock the power of AI and unleash productivity with lower total cost of ownership (TCO).

As enterprises are investigating and deploying AI, sizing is becoming a common conversation as customers start to look to deploy AI workloads from small proof of concept deployments to production implementations. Sizing the environment depends on the target use case as well as the intended scale at which those use cases will be deployed. The same use case will change based on the number of concurrent connections, for example.

This document will walk you through the core requirements that will be consistent across deployments and highlight common scenarios we see with our customers today. Once the core, shared, Management Domain is deployed, each new customer will utilize a new Workload Domain. The below sections will outline the increase per customer as use cases scale.

A common way for VCF 9 customers to deploy models, chat bots and Retrieval Augmented Generation (RAG) chatbots is through VMware Private AI Services. New in VCF 9, VMware Private AI Services is a suite of native AI capabilities integrated into the VMware Cloud Foundation platform, designed to help enterprises build and deploy AI applications securely within their private cloud. It provides the tools and infrastructure to use AI for tasks like code generation and data analysis while maintaining data privacy and control, enabling organizations to run a mix of AI and traditional workloads on a unified platform.

Customers who have adopted the solution and are consuming it increase their cluster sizes within 10 months. We will outline how to guide your customer through this scaling process.

We then will provide information on how you can investigate and adjust these to your custom deployment as you learn more about the AI requirements during your deployment with VMware Private AI Foundation with NVIDIA.

New Customer Pilot Requirements

This document outlines a lot of the details to understand sizing. The initial goal is to get a system in place for a customer to test out a few use cases. These Pilot Requirements are simplified explanations with more details to follow in the paper. The section on Selecting Servers will include links to the Broadcom Compatibility Guide.

Recommended Servers Specs:	
CPU:	2x 32 Core CPUs
RAM:	512-1024 GB
Network Cards:	Inference: 4x 25 GB NICs Training: 100GB + is recommended
Disks:	5-10x 7.68 NVMe Read Intensive Drives or similar.
GPUs:	2-4 GPUs per server -NVIDIA H200 GPU Recommended -NVIDIA L40s GPU Supported -NVIDIA RTX Pro 6000 GPU Supported. -NVIDIA H100 GPU Supported -NVIDIA A100 GPU Supported

*Assumption: existing VCF management domain

For consistency and overall design best practices, it is recommended all servers are identical. However, in a test environment, only one host needs a GPU to fully test workflows.

Use cases you can run in this scenario:

Use Case		
LLM- 8B	10 Concurrent Requests per second	1 model per GPU Expectation: 2x 8B LLMs
Chat Interface with RAG	Using the above LLM	Each deployment can isolate Data used Expectation: 5 deployments
GPU as a Service: AI Workstation	A vGPU can range from a fraction of a GPU to multiple GPUs. This is the Deep Learning VM.	Expectation: 7 instances per GPU
GPU as a Service: AI Kubernetes Cluster	A vGPU can range from a fraction of a GPU to multiple GPUs.	Expectation: 7 instances per GPU

VCF Core Management Deployment Requirements

The VMware Cloud Foundation (VCF) platform is the solution VMware provides to create a complete platform to deploy your applications on top of. This is also true for your Artificial Intelligence (AI) applications. VMware Private AI Foundation with NVIDIA runs in a Workload Domain (WLD), which is a cluster, or set of clusters, with isolation from the management domain.

This table shows the requirements needed in the Management Domain to run the necessary VCF appliances. For new deployments, the Management Cluster needs to be four hosts. Other options are available if using a brown field cluster with external storage.

VM	CPU	RAM (GB)	Disk (GB)
SDDC MGR	4	16	914
vCenter (Mgmt)	8	30	941
NSX MGR 1 (Mgmt)	6	24	300
NSX MGR 2 (Mgmt)	6	24	300
NSX MGR 3 (Mgmt)	6	24	300
NSX Edge Large	8	32	200
NSX Edge Large	8	32	200
Automation 1	24	96	529
Automation 2	24	96	529
Automation 3	24	96	529
LCM / Fleet Mgmt	4	12	194
DSM	8	16	850
Ops Mgr 1	8	32	274
Ops Mgr 2	8	32	274
Ops Mgr 3	8	32	274
Ops Cloud Proxy	4	16	265

Here we can see a total requirements in the management domain of:

Total Requirements in the Management Domain:	
CPU:	158 CPU cores
RAM:	610 GB RAM
Disk:	6.9 TB Allocated Storage

Creating a new Workload Domain and AI Cluster

To deploy AI workloads, a new WLD will be created to run them. Most of the VCF appliances will continue to run in the Management Domain. However, the services required to run AI workloads, such as Harbor or the Data Services Manager (DSM) will run in the Workload Domain.

Deployed to:	VM	CPU:	RAM (GB):	Disk (GB):
MGMT	vCenter	8	30	914
MGMT	NSX MGR 1	6	24	300
MGMT	NSX MGR 2	6	24	300
MGMT	NSX MGR 3	6	24	300
WLD	NSX Edge XL	16	64	200
WLD	NSX Edge XL	16	64	200
WLD	Supervisor Control Plane 1	8	24	48
WLD	Supervisor Control Plane 2	8	24	48
WLD	Supervisor Control Plane 3	8	24	48
WLD	Harbor (Supervisor Service)	1	3	1000

Here we can see a total requirements in the management domain of:

Total Requirements in the Management Domain:	
CPU:	26 CPU cores
RAM:	102 GB RAM
Disk:	1.8 TB Allocated Storage

Core requirements per new customer / WLD:

Total Requirements in the Management Domain:	
CPU:	57 CPU cores
RAM:	203 GB RAM
Disk:	1.6 TB Allocated Storage

Expanding AI Resources (in a Workload Domain)

A VMware Private AI Foundation with NVIDIA WLD cluster has a minimum requirement of three hosts. While each cluster can be scaled up to 64 hosts, most organizations usually choose a server count in the range of 28-36 hosts for operational reasons. Each WLD can contain multiple clusters. As customers expand, a new cluster can be created to hit higher scales. The process for adding a host is straightforward.

Steps to add a host to a VCF Cluster:

1. Install ESXi onto the physical host, including configuring the following:
    1. FQDN
    2. Certificate
    3. DNS
    4. NTP
2. Commission the host in VCF
3. Add the host to an existing Cluster.

In larger deployments, for Enterprise or for Service Providers, it is common to have a small pool of available hosts already commissioned in VCF. You must commission ESX hosts before you can use them to create a new workload domain or add them to an existing VCF domain. These hosts can then be quickly added to clusters as needed, sometimes behind a self service portal.

To add a new WLD environment, refer back to the Creating a new Workload Domain and AI Cluster section.

Understanding the AI Application Requirements

In a perfect world, the team requesting access to the platform will have clear requirements for their specific application. However, infrastructure teams are being tasked to create environments when the AI application is not fully defined. The large focus for sizing AI Workloads revolves around the inference layers. This is where people interact with the models and the AI Models recognize patterns and make a prediction to generate new content.

Questions to ask to understand requirements:

1. What is the objective of the AI application?
2. Primary purpose of this AI application: Inference or Training? Or a percentage split between these two options.
3. What LLMs or models do you plan to run? Number of Parameters?
4. How many concurrent requests will access the LLM model and / or Application? If you do not know concurrent requests, how many expected users?
5. What other models, ie embedding, reranker, speech recognition and transcription , etc do you plan to run? These models could be run independently or through the NVIDIA NIM architecture.
    1. Embedding Model:
        1. https://catalog.ngc.nvidia.com/orgs/nim/teams/nvidia/containers/nv-embedqa-e5-v5
    2. Reranker Model
        1. https://build.nvidia.com/nvidia/llama-3_2-nv-rerankqa-1b-v2/deploy
    3. Whisper Model:
        1. https://catalog.ngc.nvidia.com/orgs/nvidia/teams/riva/models/whisper_large
6. How much data do you plan to ingest into the AI application, i.e., RAG data set.
7. What level of redundancy do you want for the AI application?
8. Where will the data reside?
9. Can the database hosting the RAG data handle the I/O load?
10. What other data sources are needed for handling files / output?
11. What types of data are you planning to ingest into a RAG? Ie Word docs, markup, text

To help navigate this, we will focus on a common customer use case, Retrieval Augmented Generation (RAG) Chat bot. In our conversations with customers and in our collaboration with NVIDIA, enterprise loads are between 10 and 30 concurrent requests. We will show sizing for these ranges, then we can show an example where we scale that up. The concepts here can translate to other AI applications as you learn more about them.

Server Sizing basics with GPUs

When sizing workloads there are some basic rules of thumbs that we recommend following.

VM RAM for GPU Enabled Workload
A VM with a GPU in it should have enough RAM to support the VM. Meaning the Operating System and all the software loaded. Modern LLMs should not swap to VM RAM if working correctly. We estimate the RAM to be ok between 16-98 GB.

Previously we would recommend 1-2x the GPU RAM as Server / VM RAM. For example, if you assign a full L40s GPU to a VM, or Kubernetes node, that node will have 48 GB of GPU RAM. The VM should also have server RAM ranging from 48 to 96 GB. This is consistent whether a portion of a GPU, or multiple GPUs being assigned.

Ratio of Server RAM to GPU RAM
The same applies to the server, however the server will also have its own overhead and may run other non-GPU related workloads. A rule of thumb is to do 2-3x the total GPU RAM for server RAM. If you have a server with 4x NVIDIA H100 80GB GPUs, you would have a total of 320 GB of GPU RAM. The recommendation would be to size between 640 - 960

CPU Requirements
Generally 6 cores of CPU will be aligned to GPU, or similar portions if doing a fraction of a GPU.

Storage
vSAN ready nodes are recommended with NVMe read intensive drives. Overall quantity and size will be dictated by the use case at hand, however 5-10x drives of 7.68 TB seem to be common. When running an inference workload, storage is not critical to the operation. The speed of a storage platform will come into play when deploying, scaling out models, or ingesting data into an embedding database.

When ingesting data into a RAG system, the rule of thumb is that 1TB of documents will consume roughly 10 TB when embedded and added to the PGvector database. This is a common factor in the industry, but can vary significantly based on the types of documents, indexing settings (more chunks, more embeddings), and specific feedback from the data scientist group, such as embedding dimensionality, precision type and overlap configuration.

It is common for customers to have an S3 compatible storage endpoint to store files.

Network
Bandwidth and throughput can vary greatly between different AI use cases. For running AI workloads, or the inference phase, 4 x 25 GbE ports are the min recommended.
If you intend to take on more latency sensitive use cases or span GPU nodes you should consider 100GbE+ networking that supports RDMA end-to-end (RoCEv2).

Initial RAG for 10 concurrent requests (8B Model)

Retrieval Augmented Generation, or RAG, is a technique for enhancing the accuracy and reliability of generative AI models with facts fetched from sources that the LLM was not originally trained on, through a database. It fills a gap in how LLMs work. The configuration has multiple components that need GPUs, however the largest requirement is from the LLM itself. It is common for RAG applications to use other models than the completion model and embedding model, in this example we will include a reranker model.

When sizing GPU memory for an LLM, the biggest factor is the size of the LLM itself as well as the concurrent users. For example, an Llama 3.1 8B model can run on very little GPU for small tests. For 10 concurrent requests, the recommendation is 21 GB of GPU memory, or just under half of an L40s. For 30 concurrent requests that quickly jumps to 32 GB or almost two whole L40s GPUs. A more thorough walkthrough of how we got to these numbers is included in the appendix.

Storage will vary based on what you want to import into the RAG model.

Appliances for a RAG supporting 10 concurrent requests:

VM	CPU	RAM (GB)	Disk (GB)	GPU (GB)
Completion Model (LLM)	6	36	200	21
Embedding Model	4	20	50	10
Reranker Model	4	20	50	10
LLM Gateway	4	10	50	
Chat Front End	4	10	100	
DB -PGvector	4	8	1000	

This increases the requirements for the Workload domain by:

Total Requirements in the Management Domain:	
CPU:	26 CPU cores
RAM:	104 GB RAM
Disk:	1,450 TB Allocated Storage
GPU:	51 GB

A L40s GPU, has an estimated time to first token of 0.063 seconds. Alternatively an NVIDIA H100 80 GB GPUs with an estimated time to first token of 0.023 seconds. These times come from a blog referenced with a chart in a subsequent section. Times under 0.2 seconds are considered reasonable and acceptable. The H200s will provide a benefit at scaling up the configuration since it can handle roughly twice the load with the same number of cards.

Growing to 30 concurrent requests (8B Model)

The overall system is designed the same, but the components need to scale up to handle the increased load. In this example one NVIDIA H200 GPU is needed to run the Llama 3.1 8B model for 30 concurrent requests.

When scaling up, it can take two different forms:

1. Scale Out: Small models that can fit on a single GPU, will have multiple models deployed and sit behind a load balancer or LLM Gateway.
2. Scale Up: Larger models will leverage GPU profiles that will provide adequate GPU resources.

VM	CPU	RAM (GB)	Disk (GB)	GPU (GB)
Completion Model (LLM)	6	36	200	32
Embedding Model	4	20	50	12
Reranker Model	4	20	50	12
LLM Gateway	4	10	50	
Chat Front End	4	10	100	
DB -PGvector	4	8	1000	

This increases the requirements for the Workload domain by:

Total Requirements in the Management Domain:	
CPU:	26 CPU cores
RAM:	104 GB RAM
Disk:	1.5 TB Allocated Storage
GPU:	56 GB

Initial RAG for 10 concurrent requests (70B)

A 70B (Billion) parameter model has been trained on more information and ‘knows’ more than the smaller 8B or 14B models. Similar to the above examples, the design of the application remains consistent.

For 10 concurrent requests, the recommendation is 153 GB of GPU memory. This is where we start to see a difference in sizing between GPUs. If using NVIDIA L40s model GPUs, four will be needed. NVIDIA H200 NVL GPUs will require 2.

Appliances for a RAG supporting 10 concurrent requests for a 70B Model:

VM	CPU	RAM (GB)	Disk (GB)	GPU (GB)
Completion Model (LLM)	12	36	200	153
Embedding Model	4	36	50	40
Reranker Model	4	36	50	40
LLM Gateway	4	10	50	
Chat Front End	4	10	100	
DB -PGvector	4	8	1000	

This increases the requirements for the Workload domain by:

Total Requirements in the Workload Domain:	
CPU:	32 CPU cores
RAM:	136 GB RAM
Disk:	1.5 TB Allocated Storage
GPU:	233 GB

Growing to 30 concurrent requests (70B)

The overall system is designed the same, but the components need to scale up to handle the increased load. In this example two H200 NVL GPUs are needed to run the Llama 3.1 70B model for 30 concurrent requests.

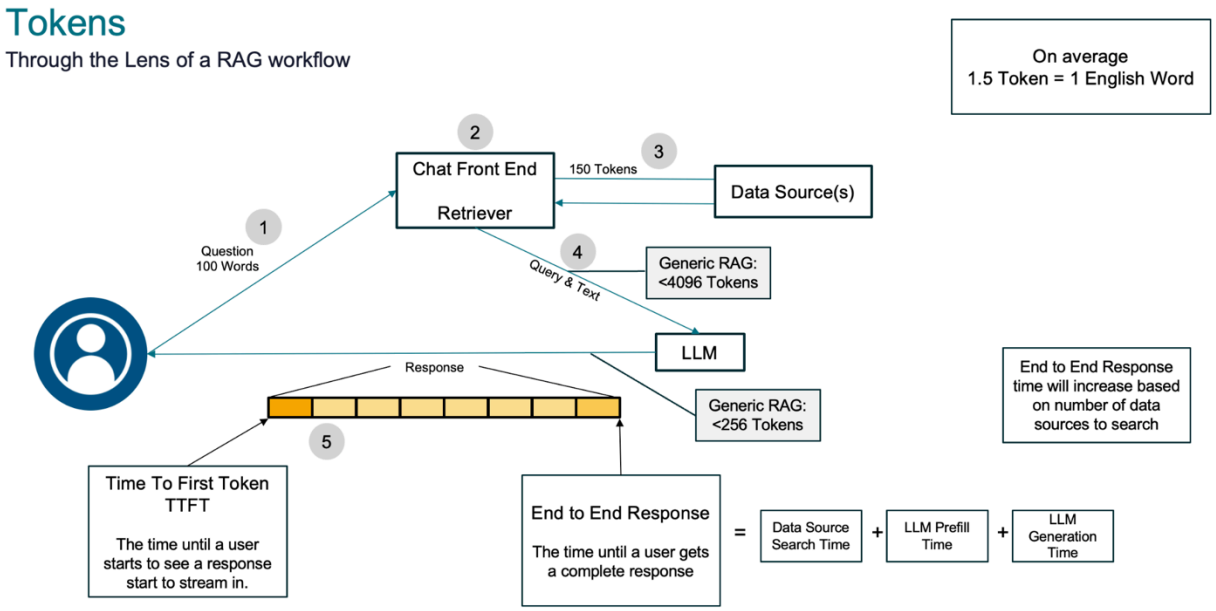
VM	CPU	RAM (GB)	Disk (GB)	GPU (GB)
Completion Model (LLM)	12	36	200	180
Embedding Model	4	36	50	50
Reranker Model	4	36	50	50
LLM Gateway	4	10	50	
Chat Front End	4	10	100	
DB -PGvector	4	8	1000	

This increases the requirements for the Workload domain by:

Total Requirements in the Workload Domain:	
CPU:	32 CPU cores
RAM:	136 GB RAM
Disk:	1.5 TB Allocated Storage
GPU:	280 GB

Time to First Token

When working with data scientists and AI engineers there are some key terms that will be helpful to know. This will help facilitate communication and help form an understanding between teams. This diagram walks through the logical flow of a request through a RAG application. English words are converted into tokens, which are then used to search data sources. The results are combined and sent to the LLM. The response from the LLM is a stream of tokens that are sent to the user.



1. A users sends a prompt to the Chat Front End
2. The front end embeds this prompt
3. The vector DB is searched for relevant information
4. The initial prompt and results from the vector search are sent to the LLM
5. The response is sent to the user. This happens in chunks, as the LLM processes it.

Sizing charts to review

The above examples make assumptions in order to provide direction for customers new to their AI journey. The chart below provides a quick view of expected response times from different models hosted on different GPUs. You can use this information to customize the scenarios based on your needs and available hardware.

These are estimates and based on the theoretical math involved in the sizing. Real world experiences will vary depending on a number of factors in the environment. For a more detailed understanding of the sizing process and how to do the sizing yourself, please refer to this blog.
https://blogs.vmware.com/cloud-foundation/2024/09/25/llm-inference-sizing-and-performance-guidance/

Models	NVIDIA GPU	Sizing Ranges (Minimum Require GPUs <1 sec Response Time)		Recommended Sizing (Target <.2 sec Response Time)		
		Concurrent Request	GPU Memory (GB)	# of GPU	Time to First Token (Seconds)	End To End Response (Seconds)
8 Billion	L40S (48 GB)	10	21	1	0.063	4.9
		30	32	2	0.063	4.9
	H100 NVL (80 GB)	10	21	1	0.023	1.1
		30	32	1	0.023	1.1
	H200 NVL (141 GB)	10	21	1	0.022	0.9
		30	32	1	0.022	0.9
14 Billion	L40S (48 GB)	10	37	1	0.115	9.0
		30	53	2	0.058	4.5
	H100 NVL (80 GB)	10	37	1	0.043	2.1
		30	53	1	0.043	2.1
	H200 NVL (141 GB)	10	37	1	0.041	1.7
		30	53	1	0.041	1.7
70 Billion	L40S (48 GB)	10	153	4	0.137	10.8
	H100 NVL (80 GB)	10	153	2	0.102	4.9
		30	180	3	0.068	3.3
	H200 NVL (141 GB)	10	153	2	0.098	4.1
		30	180	2	0.098	4.1

Models: These are the Large Language Models (LLM) that are referred to in the industry as completion models because they complete the answer to the user's question. The size of these models are based on how much information was used to train it and are referred to by the number of parameters. In this chart we show three different size models, 8, 14 and 70 Billion. The larger the model, the more it knows, but it will also require more resources to run.

NVIDIA GPU: These are the 3 most relevant GPUs available today for this solution. An A100 is also supported, but not common for new purchases. Their results would land slightly slower than the NVIDIA H100 PCIe version shown here.

GPU Memory (GB): This shows the amount of GPU RAM, or frame buffer, needed to run the model given the concurrent requests selected. This number will dictate how many GPUs will be needed to adequately run the model.

Concurrent Requests: This shows how many active requests the model will serve at one time for this sizing.

# of GPUs: This shows the minimum number of GPUs needed to run the model. More GPUs can be added to decrease the end to end response time. This will happen in one of two ways.

1. Multiple copies of the model can run on different GPUs, with a load balancer in front of them.
2. Larger models will run across multiple GPUs at the same time.

Time to First Token: This is the theoretical time that the first response will be received.

End to End Response: This is the theoretical time for the full response to be received.

VCF Components

The following is a list of VCF components deployed together to build an On-Prem Cloud Platform that provides resiliency, consistency and operational efficiency.

VCF Feature	Description	
vSphere Kubernetes Service (VKS)	Provides services utilized by the solution as well as the main deployment target	Required
NSX	Used for network services, including LBs, segments, etc.	Required
Automation	Self service portal. Place to edit templates.	Required
Data Services Manager (DSM)	Deploy and manage Databases with API endpoint to tie into templates	Highly Recommended
Harbor	Model Gallery for hosting models, such as LLMs. Hosts containers	Highly Recommended
Operations	Monitoring and action based tasks	Highly Recommended
Log Insight	Provides central logging point for deployments	Highly Recommended

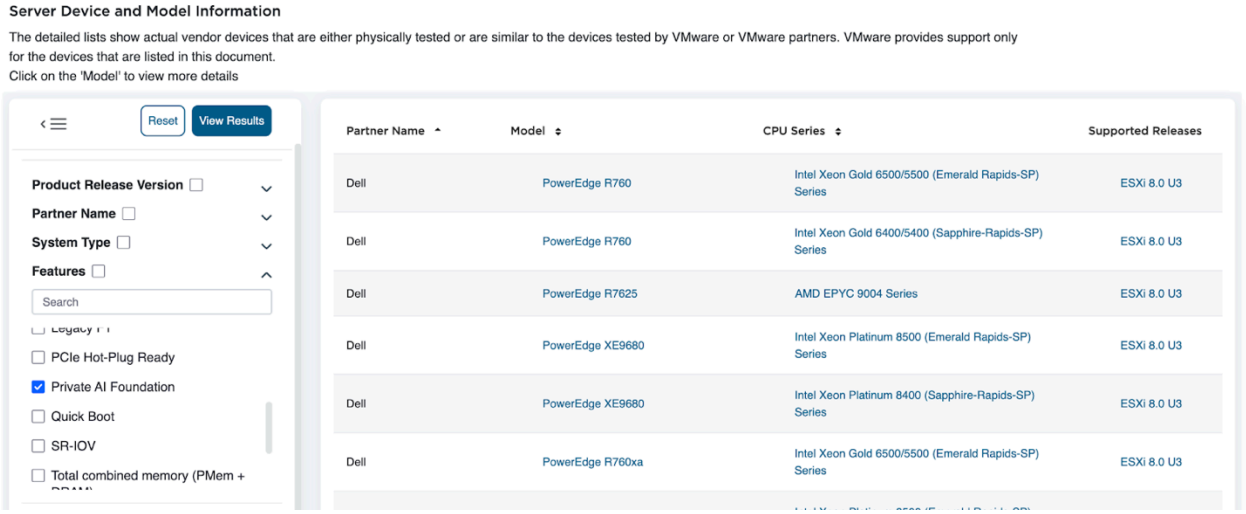
Other Broadcom Software that can be used with or as an addition to the above:

Broadcom Division	Description	
Avi	Advanced Load Balancing Solution	Highly Recommended
vDefend	Advanced Security Capabilities	Highly Recommended

Selecting Servers

Most server vendors have servers that are specifically created for AI workloads. These have gone through extra testing to ensure they are up to the task at hand. GPUs require more power and generate more heat, requiring larger power supplies and increased air flow. NVIDIA also has a list of servers that have been verified to work with specific GPUs.

The URL below provides the ability to search the Broadcom Compatibility Guide for Servers that are certified with this solution. The filter for 'Private AI Foundation' is found under features.
https://compatibilityguide.broadcom.com/



Please see the link below to the latest list of certified server models.

https://blogs.vmware.com/cloud-foundation/2024/05/02/vmware-private-ai-foundation-with-nvidia-server-guidance/

Summary

As your customers progress from initial deployments to full-scale production, accurately sizing the solution becomes essential. This sizing is heavily influenced by the specific use cases and the scale at which they are deployed, particularly in terms of concurrent connections. A consistent set of core requirements forms the foundation across all deployments, anchored by a shared management domain. From there, each new customer introduces a unique workload domain, with clearly measurable increases in infrastructure demand. Notably, we observe a common trend where customers expand their cluster sizes within 10 months of adoption, underscoring the importance of planning for scalability from the outset. This proposal outlines real-world scenarios drawn from existing customer experiences and provides guidance on how to adapt sizing strategies as AI requirements evolve. Continuous evaluation and refinement will ensure the deployment remains aligned with organizational needs.

- **Sizing Matters More Than Ever**
  As customers move to production, right-sizing based on use case and scale is critical.

- **Core Requirements Are Foundational**
  A consistent baseline exists across deployments—everything builds on the shared management domain.

- **Customer Growth Is Predictable**
  Cluster expansion typically occurs within 10 months of adoption, because of this, you should plan for scale.

- **Scenarios Reflect Real-World Trends**
  Common patterns from current customers offer valuable guidance.

- **Tailor as You Learn**
  AI workloads evolve—continue refining your deployment to match your unique demands.

Ready to get started on your AI and ML journey? Check out these helpful resources:

Read this blog for the recent announcements with VMware Private AI Foundation with NVIDIA
Complete this form to contact us!
Read the VMware Private AI Foundation with NVIDIA solution brief.
Learn more about VMware Private AI Foundation with NVIDIA.
Connect with us on Twitter at @VMwareVCF and on LinkedIn at VMware VCF.

Appendix - VMware License requirements (**required**/optional)

Management Domain		
VCF Licenses	Per CPU Cores	256
AVI Load Balancer		
vDefend		
Customer Environment (Pilot Environment)		
VCF Licenses	Per CPU Cores	192
VPAIF-N Licenses	Per CPU Cores	192
AVI Load Balancer		
vDefend		
Expansion (10 servers) to AIaaS		
VCF Licenses	Per CPU Cores	640
VPAIF-N Licenses	Per CPU Cores	640
AVI Load Balancer		
vDefend		
DataLake? GreenPlum / Tanzu Data Services		

Appendix - NVIDIA License requirements

- Management Domain - no licenses required
- NVIDIA AI Enterprise is needed for each GPU that will be used in the system.
- NVIDIA L40s and RTX 6000 GPUs which can be used for graphics as well as AI / ML workloads, will require an additional purchase of NVIDIA AI Enterprise licenses
- H100 / H200 series GPUs come with NVIDIA AI Enterprise licenses for 5 years.

Appendix - Initial Deployment Scope of Services

Many factors can adjust the implementation time frames, either to speed up the time frame or extend it. These timelines are provided as a starting point to convey the level of effort.

Proposed Initial Deployment Activities-Broadcom Professional Services
• Design and Deploy VMware Cloud Foundation • Design and Deploy NSX with VMware Cloud Foundation • Deploy Security Hardening for VMware Cloud Foundation • Design and Implement Aria Suite Lifecycle and Access Management With VMware Cloud Foundation • Design and Deploy VMware Cloud Foundation Operations for VMware Cloud Foundation • Design and Deploy VCF Operations for Logs for VMware Cloud Foundation • Design and Deploy VMware Cloud Foundation® Operations for networks for VMware Cloud Foundation • Design and Deploy VCF Automation for VMware Cloud Foundation • Implement Private AI Foundation with NVIDIA • Integration into Customer Self Service Portal

Design and Deployment Phase	Time Frame
Planning, HW requirements, IP specs, etc	3 weeks
HW rack, stack, etc	2 weeks
VCF Management Domain deploy	3 weeks
Workload Management Domain	2 weeks
VMware Private AI Foundation with NVIDIA initial Deployment	3 weeks
Customization of Automation Templates for multiple customers	2 weeks

Appendix - Walking through a sizing example

This section will walk through in more detail how we got to the sizing in the 10 concurrent requests for an 8B Model. Using the blog from the section called Sizing charts to review, we have downloaded the python script.

We enter the following command:
python3 LLM_size_pef_calculator.py -g 1 -p 4096 -r 256 -c 10

The first section shows how much GPU RAM we need to run this. To focus on the 8B model, you can see the estimated requirement is in the first row, 21.31 GB

```
******************** Estimate LLM Memory Footprint ********************
| Model           |   Input Size (tokens) |   Output Size (tokens) |   Concurrent Requests | KV Cache Size per Token   | Memory Footprint   |
|-----------------+-----------------------+------------------------+-----------------------+---------------------------+--------------------|
| Llama-3.1-8B    |                  4096 |                    256 |                    10 | 0.000122 GiB/token        | 21.31 GB           |
| Llama-3.1-70B   |                  4096 |                    256 |                    10 | 0.000305 GiB/token        | 153.28 GB          |
| Mistral-7B-v0.3 |                  4096 |                    256 |                    10 | 0.000122 GiB/token        | 19.31 GB           |
| Qwen2.5-14B     |                  4096 |                    256 |                    10 | 0.000183 GiB/token        | 37.37 GB           |
```

Next we can see expected performance for the settings we provided. For the 8B mode we can see the expected TTFT, E2E Latency.
Some highlights:

Expected result for TTFT for an L40s is is 0.063 and E2E of 4.9 sec
Expected result for TTFT for an H200 NVL is 0.022 and 0.9 sec

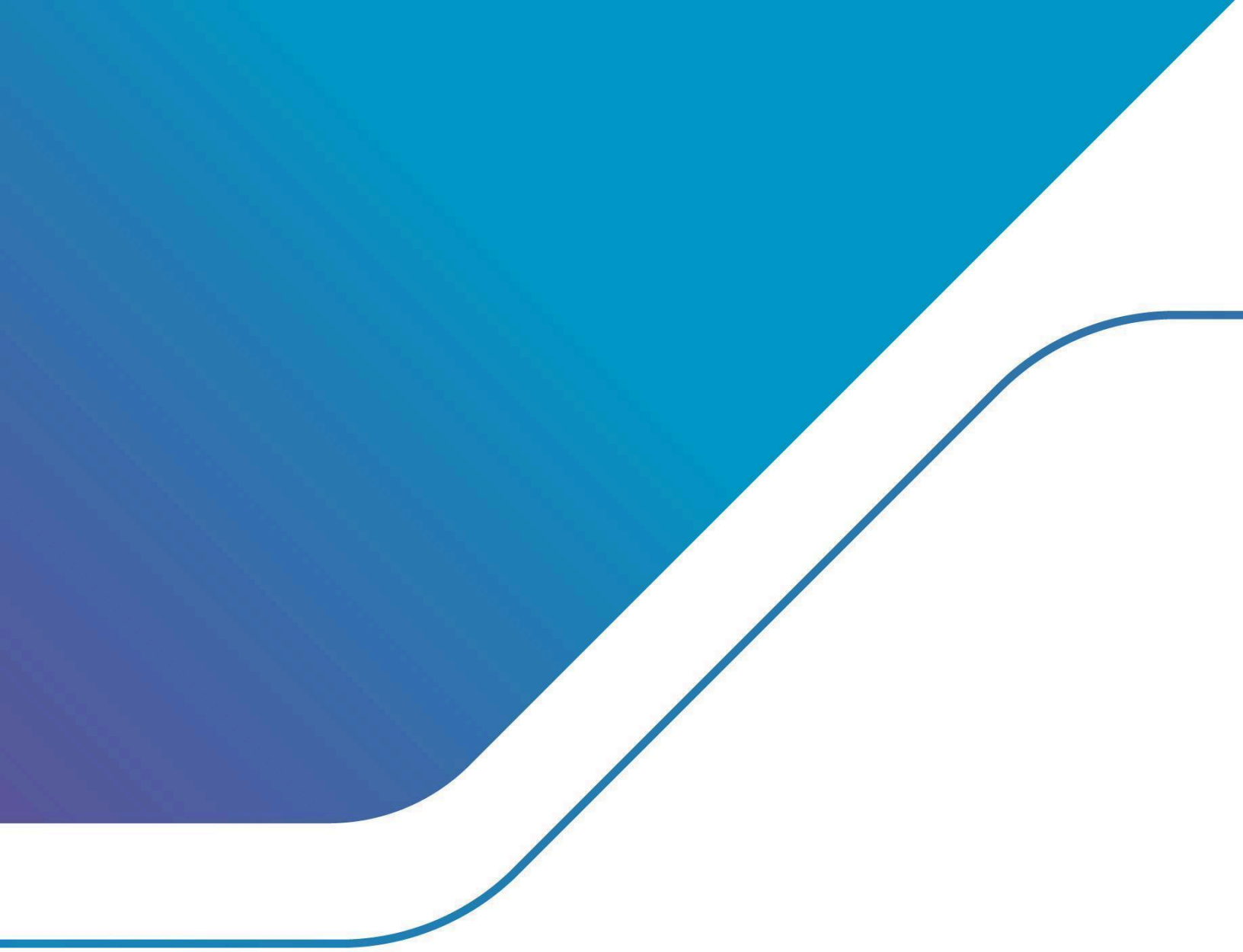
```
******************** Estimate LLM Capacity and Latency ********************
| Model           | GPU      |   Input Size (tokens) |   Output Size (tokens) |   Concurrent Requests |   Max # KV Cache Tokens | Prefill Time   | TPOT (ms)   | TTFT    | E2E Latency   | Output Tokens Throughput   |
|-----------------+----------+-----------------------+------------------------+-----------------------+-------------------------+----------------+-------------+---------+---------------+----------------------------|
| Llama-3.1-8B    | L40s     |                  4096 |                    256 |                    10 |                  262144 | 0.044 ms       | 18.519 ms   | 0.063 s | 4.9 s         | 52.01 tokens/sec           |
| Llama-3.1-8B    | H100 NVL |                  4096 |                    256 |                    10 |                  638976 | 0.019 ms       | 4.103 ms    | 0.023 s | 1.1 s         | 226.81 tokens/sec          |
| Llama-3.1-8B    | H200 NVL |                  4096 |                    256 |                    10 |                 1024000 | 0.019 ms       | 3.333 ms    | 0.022 s | 0.9 s         | 274.75 tokens/sec          |
| Llama-3.1-8B    | MI300X   |                  4096 |                    256 |                    10 |                 1441792 | 0.012 ms       | 3.019 ms    | 0.015 s | 0.8 s         | 311.07 tokens/sec          |
| Llama-3.1-70B   | L40s     |                  4096 |                    256 |                    10 |                       0 | 0.387 ms       | 162.037 ms  | 0.549 s | 43.1 s        | 5.94 tokens/sec            |
| Llama-3.1-70B   | H100 NVL |                  4096 |                    256 |                    10 |                       0 | 0.168 ms       | 35.897 ms   | 0.203 s | 9.9 s         | 25.92 tokens/sec           |
| Llama-3.1-70B   | H200 NVL |                  4096 |                    256 |                    10 |                    3276 | 0.168 ms       | 29.167 ms   | 0.197 s | 8.2 s         | 31.40 tokens/sec           |
| Llama-3.1-70B   | MI300X   |                  4096 |                    256 |                    10 |                  170393 | 0.107 ms       | 26.415 ms   | 0.134 s | 7.2 s         | 35.55 tokens/sec           |
| Mistral-7B-v0.3 | L40s     |                  4096 |                    256 |                    10 |                  278528 | 0.039 ms       | 16.204 ms   | 0.055 s | 4.3 s         | 59.44 tokens/sec           |
| Mistral-7B-v0.3 | H100 NVL |                  4096 |                    256 |                    10 |                  655360 | 0.017 ms       | 3.590 ms    | 0.020 s | 1.0 s         | 259.21 tokens/sec          |
| Mistral-7B-v0.3 | H200 NVL |                  4096 |                    256 |                    10 |                 1040384 | 0.017 ms       | 2.917 ms    | 0.020 s | 0.8 s         | 313.99 tokens/sec          |
| Mistral-7B-v0.3 | MI300X   |                  4096 |                    256 |                    10 |                 1468176 | 0.011 ms       | 2.642 ms    | 0.013 s | 0.7 s         | 355.51 tokens/sec          |
| Qwen2.5-14B     | L40s     |                  4096 |                    256 |                    10 |                  181580 | 0.081 ms       | 34.028 ms   | 0.115 s | 9.0 s         | 28.31 tokens/sec           |
| Qwen2.5-14B     | H100 NVL |                  4096 |                    256 |                    10 |                  352802 | 0.035 ms       | 7.538 ms    | 0.043 s | 2.1 s         | 123.43 tokens/sec          |
| Qwen2.5-14B     | H200 NVL |                  4096 |                    256 |                    10 |                  609484 | 0.035 ms       | 6.125 ms    | 0.041 s | 1.7 s         | 149.52 tokens/sec          |
| Qwen2.5-14B     | MI300X   |                  4096 |                    256 |                    10 |                  888012 | 0.022 ms       | 5.547 ms    | 0.028 s | 1.5 s         | 169.29 tokens/sec          |
```

Appendix - Static vs Dynamic Memory Allocation

When working with a GPU the memory is consumed in two key ways, static and dynamic.

Static memory is a fixed size based on the model's parameters, quantization and batch size. As the name suggests, this memory footprint will not change from usage. Static memory is optimized for efficiency.

Dynamic memory is variable and allocated at runtime. The key areas affecting this size will be the KV Cache and external information. KV Cache is the key value store of previous tokens and grows over use with the context window. The external information is the data passed into the process through a RAG solution, coming from a vector DB.



vmware® by Broadcom