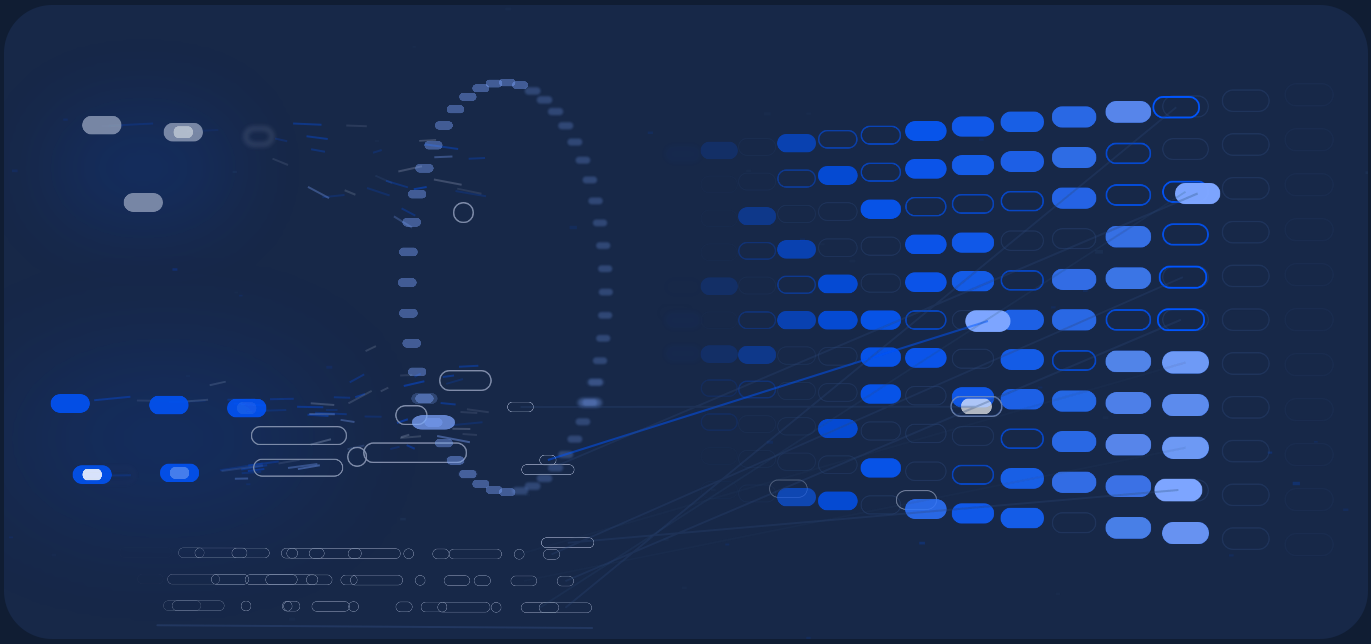
Rillor

METHODS NOTE

Reading GPU prices from cloud pricing APIs

The GPU pricing APIs of AWS, Azure, Google Cloud, Oracle and Lambda differ in access, units and coverage, so each needs its own conversion to dollars per GPU-hour.



Contents

01 Summary

02 Background

03 Method

04 Findings

05 Implications

06 Limits of this analysis

07 Sources

Summary

- AWS, Microsoft Azure, Google Cloud, Oracle Cloud Infrastructure and Lambda publish GPU prices through a cloud pricing API. Access ranges from none (Azure, Oracle) to an API key (Google Cloud, Lambda) to IAM permission (AWS) [1, 17, 22, 27, 29].
- No single API covers every pricing type. The AWS price list leaves out Spot, its Query API leaves out Savings Plans and its Capacity Block SKUs read $0.00, so one AWS GPU instance needs four APIs and a pricing page [1, 9, 10, 13].
- Units differ. AWS returns instance-hours plus upfront fees, Azure returns reservations as term totals labelled "1 Hour", Google Cloud returns units plus nanos per resource, Lambda returns US cents per instance-hour and Oracle returns a price per GPU-hour [13, 18, 22, 27, 30].
- Reaching U.S. dollars per GPU-hour takes a GPU count from a second source, care with fractional GPUs, a stated hour count per term and, at Google Cloud, a sum of SKUs [12, 15, 21, 26]. H100 on-demand listings read this way on 24 October 2026 ran from $3.99 to $12.29 per GPU-hour, for different products [18, 28].
- The time of the call is the one timestamp all five share. AWS Spot history reaches back 90 days and Google's Catalog history one calendar month, so a price series exists only where responses were stored [7, 23]. FOCUS 1.4 supplies shared terms for the stored prices [32, 33].

Background

GPU instances are priced by the hour, and the hourly numbers are large. On 24 October 2026 AWS listed its eight-GPU p5.48xlarge in US East (N. Virginia) at $55.04 per instance-hour on demand [13], and Azure listed its eight-GPU ND96isr H100 v5 in East US at $98.32 [18]. At 8,760 hours, that is about $482,000 and $861,000 for a year of one instance.

The providers publish these prices through APIs. AWS says its Query API can be used to "build cost control and scenario planning tools, reconcile billing data, forecast future spend for budgeting purposes, and provide cost benefit analyses" [1]. Microsoft notes that before its Retail Prices API, "the only way that you could retrieve prices for Azure services was to either use the Azure Pricing Calculator or use the Azure portal" [17]. Billing exports do not help before a purchase: "Today, FOCUS data shows a rate only once a charge has been billed at it" [35].

This note maps what each returns for GPU compute, what access it needs, what it leaves out and how its prices reach one unit. It is written for FinOps analysts, data engineers and procurement teams who pull GPU prices from several providers and compare them.

Method

What was examined. For each provider: the endpoints that return GPU compute prices, the access they require, documented paging and rate limits, the unit and currency of each price, the pricing types covered, the fields that identify the GPU and the timestamps returned.

Sources. Provider API references and user guides [1 to 9, 11, 12, 17, 20 to 24, 26, 27, 29, 31], a Microsoft skill reference on GitHub for one rule the Azure documentation leaves out [19], the AWS, Google Cloud and Lambda pricing pages [10, 25, 28], FOCUS 1.4 and the 1.5 release scope [32 to 35] and NVIDIA's H100 specifications [36], all read on 24 October 2026. Lambda's pricing page was read again on 7 November 2026 [37].

Live responses. Read-only requests went on 24 October 2026, 18:03 to 18:28 UTC, to the AWS Price List, Savings Plans and EC2 APIs [13, 14, 15], the AWS price list files [16], the Azure Retail Prices API [18] and Oracle's list price API [30]. Repeat requests returned the same values. Unauthenticated requests to Google's Catalog API and Lambda's instance-types endpoint returned HTTP 403 and 401. No key was created, so those two APIs are described from documentation only.

How. Documented behavior is quoted or paraphrased. Live prices are reported as published, with unit, region, operating system and pricing type. Per-GPU figures are arithmetic: an instance price divided by its GPU count, and a term total or upfront fee divided by the hours in the term, with the hour count stated. No price here is a transaction price or includes tax, storage, egress or support.

Findings

1. Five APIs, three kinds of access

Provider and API	Access	Documented limits	Price unit as returned
AWS Price List Query API and Bulk API	IAM permission [1]	GetProducts: 100 results per call [2], token bucket 10, refill 5 per second [3]	Instance-hours and upfront "Quantity" [13]
Azure Retail Prices API, 2023-01-01-preview	None [17]	1,000 records per call [17]	"1 Hour" on every row, term totals included [18]
Google Cloud Billing Catalog API v1, Pricing API v1beta and v2beta	API key [22, 24]	5,000 SKUs per page [22]	Units plus nanos per usage unit, tiered [22, 23]
Lambda Cloud API 1.10.0, instance types	Account API key [27]	About one request per second [27]	US cents per instance-hour [27]
Oracle list price API	None [29]	None found	Per metric, "GPU Per Hour" for GPUs [30]

Azure's API gives "an unauthenticated experience to get retail rates for all Azure services" [17], and Oracle's endpoint is a plain GET [29]. AWS requires IAM permission for both Price List APIs [1], although its price list files downloaded without request signing [16]. Google and Lambda require API keys [22, 27].

Those APIs' terms of use were not reviewed here. Anyone building a collector should read them, and each source's access requirements, first.

2. AWS spreads one instance's prices across four APIs

The price list. The Query API "uses product attributes and provides prices at the SKU level", and the Bulk API serves files "organized by AWS service and AWS Region" [1]. GetProducts returns each product as a JSON document inside a string [2], with Reserved, OnDemand and FlatRate terms [5].

The request used was GetProducts on AmazonEC2 at the us-east-1 endpoint, with exact-match filters instanceType p5.48xlarge, regionCode us-east-1, operatingSystem Linux, preInstalledSw NA and tenancy Shared. It returned six SKUs from price list version 20261023221355, published 23 October 2026 at 22:13:55 UTC [13].

capacitystatus	marketoption	On-demand price per "Hrs"
Used	OnDemand	55.04
UnusedCapacityReservation	OnDemand	55.04
AllocatedCapacityReservation	OnDemand	0.00
Used	CapacityBlock	0.00
AllocatedCapacityReservation	CapacityBlock	0.00
UnusedCapacityReservation	CapacityBlock	0.00

Source: [13]. Captured 24 October 2026, 18:05 UTC.

The on-demand price sits on the first SKU. The others describe capacity reservations and Capacity Blocks, so they repeat the rate or read zero. A query without capacitystatus and marketoption collects four zeros and a duplicate. Every row carried gpu "8" and gpuMemory "640 GB HBM3", and none named the GPU model [13].

The stated limits. The catalog "doesn't include Amazon Elastic Compute Cloud (Amazon EC2) Spot Instances", and "The Price List Query API doesn't support Savings Plan prices" [1]. Both APIs give prices "for informational purposes only", and "If there's a difference between the price list file and a service pricing page, AWS charges the prices on the service pricing page" [1].

Spot. DescribeSpotPriceHistory takes a StartTime "up to the past 90 days" and "also returns the last price change before the start time, which is the effective price as of the start time" [7]. With a one-minute window for p5.48xlarge, Linux/UNIX, us-east-1, it returned one price per Availability Zone, from 23.7429 to 37.3777 USD per instance-hour, last changed between 11:00 and 17:00 UTC that day [15]. That is $2.97 to $4.67 per GPU-hour, a 1.57-fold spread across zones. A request for 26 June to 16 July 2026, 100 to 120 days back, returned no records and no error [15].

Savings Plans. DescribeSavingsPlansOfferingRates "Describes the offering rates for Savings Plans you might want to purchase" [8]. Filtered to p5.48xlarge, us-east-1, Linux/UNIX and shared tenancy, it returned 24 results in USD per "Hrs" [14].

Plan type	1 yr, no upfront	1 yr, all upfront	3 yr, no upfront	3 yr, partial	3 yr, all upfront
Compute	43.344	40.4544	40.42138	37.4272	36.67866

Plan type	1 yr, no upfront	1 yr, all upfront	3 yr, no upfront	3 yr, partial	3 yr, all upfront
EC2Instance	34.6752	32.36352	23.77728	22.016	20.69504

Source: [14]. USD per instance-hour, captured 24 October 2026, 18:05 UTC.

This API states upfront plans as hourly rates, while the price list states Reserved Instances as an upfront fee plus an hourly charge [13]. AWS defines one year as 31,536,000 seconds and three years as 94,608,000 seconds [11], which is 8,760 and 26,280 hours. On that basis the two agree. The three-year standard All Upfront fee of 543,866 over 26,280 hours is 20.6951, the EC2Instance three-year all-upfront rate. The convertible All Upfront fee of 963,915 over 26,280 hours is 36.6787, the Compute plan's rate [13, 14].

Capacity Blocks. These reserve GPU instances for a fixed period, and DescribeCapacityBlockOfferings takes a reservation duration, up to 182 days, and an instance count [9]. The pricing page lists $47.757 per instance-hour, or $5.970 per accelerator-hour, for p5.48xlarge in US East (N. Virginia) [10]. AWS says "Reservation prices are updated regularly based on trends in supply and demand" and "The current prices are scheduled to be updated next in January, 2027" [10]. The price list's $0.00 rows carry none of this [13].

3. Azure returns hourly rates and term totals in one list

The API. One public endpoint, https://prices.azure.com/api/retail/prices, filtered with $filter expressions [17]. "Prices shown in USD currency are Microsoft retail prices. Other non-USD prices returned by the API are for your reference" [17]. retailPrice is "Microsoft Retail Prices without discount" [17].

Version matters. Savings plan rates are available only with the 2023-01-01-preview version, and "in the 2023-01-01-preview version and later, the value is case sensitive" [17]. In a test on 24 October 2026, the lowercase value standard_nd96isr_h100_v5 returned zero items under 2021-10-01-preview, under 2023-01-01-preview and with no version, while Standard_ND96isr_H100_v5 returned 12 under both [18]. Treat every version as case sensitive.

The response. With api-version 2023-01-01-preview and the filter armRegionName eq 'eastus' and armSkuName eq 'Standard_ND96isr_H100_v5', the API returned 12 items in USD and no next page [18].

meterName	type	reservationTerm	retailPrice	effectiveStartDate
ND96isrH100v5	Consumption	none	98.32	2023-12-01
ND96isrH100v5	Reservation	1 Year	551,221	2023-12-01
ND96isrH100v5	Reservation	3 Years	1,134,310	2023-12-01

meterName	type	reservationTerm	retailPrice	effectiveStartDate
ND96isrH100v5	Reservation	5 Years	1,722,566	2023-12-01
ND96isrH100v5 Spot	Consumption	none	18.169536	2026-01-01
ND96isrH100v5 Low Priority	Consumption	none	19.664	2023-12-01

Source: [18]. Linux rows, unitOfMeasure "1 Hour" on every row. Captured 24 October 2026, 18:05 UTC, unchanged at 18:22 UTC. The other rows were Windows and DevTestConsumption meters.

Four things need care.

- **Term totals under an hourly unit.** The reservation rows hold one value per term under unitOfMeasure "1 Hour" [18], as does Microsoft's own sample, a one-year reservation at 25007.0 [17]. A Microsoft skill reference for its Fabric service, written for LLM agents, states that reservation rows are term totals and must not be multiplied by 730 hours [19].
- **An undocumented term.** The field table describes reservationTerm as one year or three years [17]. The response also had a five-year term [18].
- **Pricing types in the meter name.** For Spot, "The meterName and skuName will both contain Spot" [20]. Low Priority rows appear the same way, and the sources read do not define them for these VMs [18].
- **Inconsistent labels.** ND96isr H100 v5 Linux rows carry no "Linux" label, while the NC24ads A100 v4 Linux product name ends in "Linux" [18]. A filter on the word fails for one of them.

Savings plans. "For the meters eligible for a savings plan, results have a corresponding section for savings plan" [17]. The ND96isr H100 v5 rows had none [18]. The Standard_NC24ads_A100_v4 Linux row in East US did: 3.0574052 for 1 Year and 2.2658737 for 3 Years, beside 3.673 on demand [18]. Within one API, then, a reservation is a term total and a savings plan is an hourly rate.

Converting the totals. Microsoft states no hour count for reservation terms. At an assumed 8,760 hours a year, the one-year total is $62.92 per instance-hour, or $7.87 per GPU-hour. The three-year total over 26,280 hours is $5.40 per GPU-hour, and the five-year total over 43,800 hours is $4.92 [18]. That is 64.0%, 43.9% and 40.0% of on-demand for the same hours. Spot is $2.27 per GPU-hour, 18.5% of on demand [18].

4. Google Cloud prices GPU machines by resource

Catalog API. The Cloud Billing Catalog API lists "all public SKUs within a service", with descriptions, public pricing, regions and category data, at up to 5,000 SKUs per page [22]. Each SKU has a resource group such as "GPU" and a usage type, for example OnDemand, Preemptible, Commit1Mo or Commit1Yr [22]. "The cost

of the SKU is units + nanos": $1.75 "is represented as units=1 and nanos=750,000,000" [22]. Rates can be tiered by usage, and "The total cost is computed by applying each of the tiered rates on usage" [22, 23].

The list method limits history. Without a time range, "the latest pricing will be returned (up to 12 hours old at most)" [23]. With one, "The time range has to be within a single calendar month in America/Los_Angeles timezone" [23]. Conversion to another currency uses a rate that "includes any surcharge collected for billing in non USD currency" [23].

Pricing API. The newer Pricing API is in Preview, under Google's Pre-GA terms, and "Pre-GA products and features are available "as is" and might have limited support" [24]. The page's examples list services and SKUs with v2beta and read prices with v1beta, and its API reference lists price methods under both versions [24]. Public calls return listPrice. Calls for a Cloud Billing account need billing.billingAccountPrice.get and add contractPrice and effectiveDiscountPercent [24]. In the documentation read, this is the one API of the five that returns a list price and a contract price side by side.

Machines are sums. "Google Cloud bills A3 instances for their attached GPUs, predefined vCPUs, memory, and bundled Local SSD disks" [26]. The a3-highgpu-8g has 208 vCPUs, 1,872 GB of memory, 6,000 GiB of Local SSD and eight H100 GPUs [26], so its hourly price is four kinds of SKU, each multiplied by its quantity. A3 Ultra offers Spot, Flex-start, reservations and commitments with no on-demand option, and A3 High in 1, 2 or 4 GPU sizes offers only Spot and Flex-start [26].

Spot timing. The VM pricing page says Spot prices "can change up to once every 30 days" in one note and "up to once every day" in another [25]. A daily read covers both.

5. Lambda prices instances in cents, Oracle prices GPUs by part number

Lambda. GET /api/v1/instance-types "Retrieves a list of the instance types currently offered on Lambda's public cloud" with "resource specifications, pricing, and regional availability" [27]. Requests are "generally limited to one request per second" [27]. Each item holds an instance type (name, description, gpu_description, price_cents_per_hour and specs including gpus) and regions_with_capacity_available [27]. The price is "The price of the instance type in US cents per hour", an integer for the whole instance [27]. The schema has no field for term, pricing type or time [27].

The pricing page states prices per GPU, which rise as nodes shrink [28]. On 24 October 2026 H100 SXM 80 GB was $3.99 per GPU-hour in an 8-GPU instance, $4.09 in a 4-GPU, $4.19 in a 2-GPU and $4.29 in a 1-GPU instance. H100 PCIe 80 GB was $3.29 [28]. H100 1-Click Clusters for two weeks to one year were $6.16, $5.85 and $5.54 per GPU-hour for 16, 64 and 256 GPUs [28]. Every table carries the footnote "plus applicable sales tax/VAT/GST" [28]. On 7 November 2026 the page listed instance prices taking effect on 16 November 2026 at 00:00 UTC: H100 SXM at $4.59 per GPU-hour in an 8-GPU instance, $4.69 in a 4-GPU, $4.79 in a 2-GPU and $4.89 in a 1-GPU instance, and H100 PCIe at $3.79. The cluster prices were unchanged [37].

Oracle. Oracle documents a public products endpoint filtered by partNumber and currencyCode, and "Valid JSON is returned as a response" [29]. The documentation and the live response disagree in shape. The schema puts currencyCode, model and value in one object and types value as a string [29]. The sample nests a prices array in each currency object and shows value as a number [29]. The live USD response put prices under "currencyCodeLocalizations", with numeric values [30]. A parser built from either documented form fails on it.

The response held 668 items and one lastUpdated, 8 October 2026 at 14:38 UTC [30]. Selected "GPU Per Hour" lines, in USD:

Part number	Display name	Per GPU-hour
B98415	Oracle Cloud Infrastructure - Compute - GPU - H100	10
B110519	OCI - Compute - GPU - H200	10
B110978	OCI - Compute - GPU - B200	14
B110979	OCI - Compute - GPU - GB200	16
B112140	OCI - Compute - GPU - GB300	18
B109485	OCI - Compute - GPU - MI300X	6

Source: [30]. Captured 24 October 2026, 18:03 UTC.

Items carry no region field, and only the PAY_AS_YOU_GO model appeared [30]. A single-part request with no currency returned 50 currencies [30]. An "H100T" line, at 10.75, carries no definition in the sources read [30]. Lines such as "OCI - NVIDIA AI Enterprise - H100", at 2.5, use the same "GPU Per Hour" metric and category [30]. The name points to NVIDIA software, so a filter on the metric alone mixes software with GPU lines.

6. What the APIs leave out

Missing from the API read	Where it is published
AWS Spot prices in the price list [1]	DescribeSpotPriceHistory, 90 days back [7]
AWS Savings Plans in the Query API [1]	DescribeSavingsPlansOfferingRates [8]

Missing from the API read	Where it is published
AWS Capacity Block fees, shown as $0.00 [13]	DescribeCapacityBlockOfferings and the pricing page [9, 10]
AWS GPU model in price attributes [13]	DescribeInstanceTypes [12, 15]
Azure savings plans before 2023-01-01-preview [17]	The preview version [17]
Azure GPU count [18]	The VM size pages [21]
Google contract prices on public calls [24]	Billing-account calls to the Pricing API [24]
Lambda term, pricing type and timestamp [27]	The pricing page, for clusters and per-GPU prices [28]
Oracle region, node size and term [30]	The shapes page, for node configurations [31]

Some costs sit outside every API read here. Lambda adds sales tax [28], and Google bills disks and network apart from A3 machine pricing [26]. Storage, egress and support fees were outside this note's scope.

7. Matching a SKU to a GPU class and configuration

Each API puts the GPU model, count and configuration somewhere else.

- **AWS.** Price attributes give count and memory but no model [13]. DescribeInstanceTypes supplies the model: GpuDeviceInfo has a name, a manufacturer, "The number of GPUs for the instance type" and "The size of each GPU as a fraction of a full GPU" [12]. It returned H100, count 8, 81,920 MiB per GPU for p5.48xlarge, and B200, count 8, for p6-b200.48xlarge [15]. It names no form factor.
- **Azure.** The response has no GPU fields [18]. The size page does: "The ND H100 v5 series starts with a single VM and eight NVIDIA H100 Tensor Core GPUs", each with a dedicated 400 Gbps InfiniBand connection [21].
- **Google Cloud.** SKUs carry a description and a taxonomy, with categories such as "GPUs", "GPUs Preemptible" and "L4" in Google's example [24]. The machine family page gives the form factor: A3 High, Mega and Edge attach "NVIDIA H100 SXM GPUs" [26].
- **Lambda.** gpu_description names model, memory and form factor, as in the example "H100 (80 GB SXM5)", and specs.gpus gives the count [27].
- **Oracle.** The part name gives the model only [30]. The shapes page maps it to BM.GPU.H100.8: eight H100 GPUs, 112 OCPUs, 2,048 GB of CPU memory and 16 NVMe drives of 3.84 TB [31], with "1 OCPU on x86 (AMD and Intel) = 2 vCPUs" [31]. At $10 per GPU-hour that shape is $80 an hour [30]. Whether the GPU

price includes the shape's CPU, memory and disk is not stated in the sources read.

A model name does not fix the variant. The H100 SXM has 80 GB, 3.35 TB/s and 900 GB/s NVLink. The H100 NVL has 94 GB, 3.9 TB/s and 600 GB/s NVLink in a PCIe card [36]. AWS's "H100" does not say which [15]. Lambda priced one H100 SXM at $4.29 and one H100 PCIe at $3.29 on 24 October 2026, and lists $4.89 and $3.79 from 16 November 2026 [28, 37].

FOCUS 1.4's SKU Price Details column defines a GpuCount key, "Number of GPUs available", and an InstanceType key [34]. It defines no key for GPU model or form factor, and custom keys must begin with "x_" [34].

8. From instance price to GPU-hour

Divide AWS and Azure instance prices by the GPU count from DescribeInstanceTypes or the size page [15, 21], Lambda's cents by 100 and then by specs.gpus [27], and Google's summed component SKUs by the machine's GPUs [26]. Oracle's price is per GPU already [30].

Fractional GPUs. For g6f.large, DescribeInstanceTypes returned an L4 with Count 0, LogicalGpuCount 1 and GpuPartitionSize 0.125, and the price list gave gpu "0.125" and $0.202 per hour [13, 15]. Dividing by Count divides by zero. Dividing by the partition size gives $1.616 per full-GPU-hour.

Stated hours. AWS defines its terms in seconds [11], so its conversions are exact. Microsoft gives no hour count, so every Azure figure here states 8,760 hours a year. Oracle's console estimates assume "24 hours a day, 31 days a month (744 hours)" [29]. The Fabric skill reference turns reservations into months by dividing by 12 or 36 [19]. The hour count belongs in the record beside the result.

H100 listing, 24 October 2026	Pricing type	USD per GPU-hour
AWS p5.48xlarge, US East (N. Virginia)	On-demand	6.88
AWS p5.48xlarge, US East (N. Virginia)	Capacity Block	5.970
AWS p5.48xlarge, six us-east-1 zones	Spot	2.97 to 4.67
AWS p5.48xlarge, us-east-1	EC2 Instance Savings Plan, 3 yr, all upfront	2.587
Azure ND96isr H100 v5, East US	On-demand	12.29
Azure ND96isr H100 v5, East US	Reservation, 1 yr at 8,760 h	7.87
Azure ND96isr H100 v5, East US	Spot	2.27

H100 listing, 24 October 2026	Pricing type	USD per GPU-hour
Oracle B98415, no region	Pay as you go	10.00
Lambda H100 SXM, 8-GPU instance (page)	On-demand	3.99

Sources: [10, 13, 14, 15, 18, 28, 30].

One unit does not make one product. Each row has its own host, network, node size and terms, and the Lambda row comes from a page. What posted GPU cloud prices tell you sets out the adjustments a fair comparison still needs.

9. Recording capture time and raw responses

- **AWS.** Files carry publicationDate, "The date and time in UTC format when a service price list file was published", and a version that changes with each file [5]. The version index lists every historical version [4]. On 24 October 2026 the EC2 index listed 123 versions back to December 2015, one per calendar month in 2026, with October effective from the 1st [16]. "Price list files can change anytime", and AWS offers SNS notifications for each change or once a day [6]. The us-east-1 EC2 file alone was about 482 MB [16].
- **AWS Spot.** Each record carries the time of a price change, 90 days back [7, 15].
- **Azure.** Rows carry an optional effectiveStartDate [17], 1 January 2026 for the ND H100 v5 Spot row [18].
- **Google Cloud.** Without a time range, effectiveTime is "a time within the last 12 hours", and history stays within one calendar month [23].
- **Lambda.** No time field [27].
- **Oracle.** One lastUpdated for the whole list [30].

The call time is the one timestamp all five share. A collector that stores it in UTC beside the unchanged response can rebuild any later reading. One that keeps only parsed prices cannot recover a field it skipped, such as an Azure five-year term or an AWS capacitystatus. The proposed FOCUS SKU Price dataset works the same way: "History comes from each price's effective dates, and from keeping successive snapshots where a provider does not deliver past prices" [35]. Datasets that state their limits describes the provenance such a record carries.

10. Aligning captured prices with FOCUS terms

FOCUS is "an open specification supported by the FinOps Foundation that normalizes billing data", and 1.4 is the current published version [35].

- **List Unit Price** is "the suggested service-provider-published unit price for a single Pricing Unit of the associated SKU, exclusive of any discounts", and its values "should only be viewed in the context of their row and not aggregated to produce a total" [32]. Azure's retailPrice "without discount" [17] and Google's listPrice [24] fit it. An Azure reservation total needs its unit restated first.
- **Pricing Category** allows Standard, Dynamic, Committed and Other [33]. Dynamic covers "any product or service with a unit price the service provider can change without notice, like interruptible or low priority resources" [33]. Spot, Low Priority and Preemptible prices are Dynamic on that definition, on-demand prices are Standard, and Savings Plan, reservation and commitment rates are Committed.
- **SKU Price Details** carries GpuCount and InstanceType, with custom keys for the rest [34].

FOCUS defines these columns for billed charges [35], so applying them to captured list prices aligns definitions and makes no dataset FOCUS conformant. The 1.5 release, with ratification set for 3 December 2026, has a SKU Price dataset "in final review" that publishes "what a SKU costs, before anyone buys it" [35]. Work not approved by the 8 October Member approval deadline moves to the next release, so its place in 1.5 was open on 24 October 2026 [35].

Implications

For FinOps analysts. Read each price where it binds: for AWS, the service pricing page where it differs from the file [1]. Keep pricing types apart, tagged Standard, Dynamic or Committed [33]. Convert term totals and upfront fees with a stated hour count, and store the count [11, 18].

For data engineers. Build one parser per API and test it against the live response on every run, since Oracle's live shape matched neither its schema nor its sample [29, 30]. Filter AWS on capacitystatus and marketoption, Azure on exact armSkuName casing and Oracle on known hardware part numbers [13, 18, 30]. Store every raw response with its UTC call time, because Spot history ends at 90 days and Google's Catalog history at one month [7, 23]. Respect documented limits [3, 27] and each source's terms.

For procurement teams. An API list price is a public listing for one configuration, region and term. Google's Pricing API shows an account's contract price beside it [24]. Compare a quote with the listing for the same GPU variant, node size, region and term.

Rillor's GPU and neocloud pricing method reads official provider pricing APIs and published pricing pages within each source's terms, records each observation's UTC capture time, keeps each raw response unchanged and converts each price to U.S. dollars per GPU-hour.

Limits of this analysis

- All observations date from 24 October 2026, except Lambda's prices from 16 November 2026, read on 7 November [37]. API versions, fields and prices change, and Google's Pricing API is in Preview [24].
- No live Google Cloud or Lambda response was captured. The note gives no Google GPU price and does not test whether Lambda's API price for an instance equals the page's per-GPU price times its GPU count.
- Terms of use were not reviewed, and rate limits were found only for AWS and Lambda [3, 27].
- The EC2 version index shows one version per month in 2026 [16], while AWS says files can change at any time [6]. Whether intra-month versions are kept is not known from these sources.
- Microsoft does not document the hours behind reservation totals, Low Priority meters for these VMs, the five-year term or savings plan eligibility. Azure conversions assume 8,760 hours a year.
- Oracle's sources do not say whether the per-GPU price covers the shape's CPU, memory and disk, or what "H100T" means.
- Neocloud APIs beyond Lambda and third-party price aggregators were not examined.
- No price here includes tax, storage, egress, support or negotiated discounts.

Sources

1. Calling AWS services and prices using the AWS Price List, AWS Billing and Cost Management User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/price-changes.html

2. GetProducts, AWS Billing and Cost Management API Reference. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/aws-cost-management/latest/APIReference/API_pricing_GetProducts.html

3. Quotas and restrictions, AWS Billing User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/billing-limits.html

4. Getting price list files manually, AWS Billing User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/using-the-aws-price-list-bulk-api-fetching-price-list-files-manually.html

5. Reading the service price list file for an AWS service, AWS Billing User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/reading-service-price-list-file-for-services.html

6. Setting up price update notifications, AWS Billing User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/price-notification.html

7. DescribeSpotPriceHistory, Amazon EC2 API Reference. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_DescribeSpotPriceHistory.html

8. DescribeSavingsPlansOfferingRates, Savings Plans API Reference. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/savingsplans/latest/APIReference/API_DescribeSavingsPlansOfferingRates.html

9. DescribeCapacityBlockOfferings, Amazon EC2 API Reference. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_DescribeCapacityBlockOfferings.html

10. Amazon EC2 Capacity Blocks for ML pricing. Amazon Web Services, viewed 24 October 2026. https://aws.amazon.com/ec2/capacityblocks/pricing/

11. Reserved Instances for Amazon EC2 overview, Amazon EC2 User Guide. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-reserved-instances.html

12. GpuDeviceInfo, Amazon EC2 API Reference. Amazon Web Services, viewed 24 October 2026. https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_GpuDeviceInfo.html

13. AWS Price List Query API, GetProducts responses for p5.48xlarge and g6f.large in us-east-1 (AmazonEC2, Linux, shared tenancy, price list version 20261023221355), read-only requests. Amazon Web Services, retrieved 24 October 2026, 18:05 UTC. https://api.pricing.us-east-1.amazonaws.com

14. Savings Plans API, DescribeSavingsPlansOfferingRates response for p5.48xlarge in us-east-1 (Linux/UNIX, shared tenancy), read-only request. Amazon Web Services, retrieved 24 October 2026, 18:05 UTC. https://savingsplans.amazonaws.com

15. Amazon EC2 API, DescribeSpotPriceHistory response for p5.48xlarge (Linux/UNIX) and DescribeInstanceTypes responses for p5.48xlarge, p6-b200.48xlarge and g6f.large in us-east-1, read-only requests. Amazon Web Services, retrieved 24 October 2026, 18:05 UTC. https://ec2.us-east-1.amazonaws.com

16. AWS Price List Bulk API files for Amazon EC2: service version index, current region index and current us-east-1 price list file. Amazon Web Services, retrieved 24 October 2026. https://pricing.us-east-1.amazonaws.com/offers/v1.0/aws/AmazonEC2/index.json

17. Azure Retail Prices overview, Cost Management REST API. Microsoft Learn, updated 7 August 2024, viewed 24 October 2026. https://learn.microsoft.com/en-us/rest/api/cost-management/retail-prices/azure-retail-prices

18. Azure Retail Prices API responses for Standard_ND96isr_H100_v5 and Standard_NC24ads_A100_v4 in East US (api-version 2023-01-01-preview), with a filter case test under 2021-10-01-preview. Microsoft, retrieved 24 October 2026, 18:05 UTC. https://prices.azure.com/api/retail/prices?api-version=2023-01-01-preview&$filter=armRegionName%20eq%20%27eastus%27%20and%20armSkuName%20eq%20%27Standard_ND96isr_H100_v5%27

19. Azure Pricing API Reference, e2e-fabric-cost-estimation skill, microsoft/skills-for-fabric. Microsoft on GitHub, last changed 30 July 2026, viewed 24 October 2026. https://github.com/microsoft/skills-for-fabric/blob/main/skills/e2e-fabric-cost-estimation/resources/pricing-api-reference.md

20. About Azure Spot Virtual Machines. Microsoft Learn, updated 6 February 2026, viewed 24 October 2026. https://learn.microsoft.com/en-us/azure/virtual-machines/spot-vms

21. ND-H100-v5 size series. Microsoft Learn, updated 25 September 2026, viewed 24 October 2026. https://learn.microsoft.com/en-us/azure/virtual-machines/sizes/gpu-accelerated/ndh100v5-series

22. Get publicly available Google Cloud pricing information (Cloud Billing Catalog API). Google Cloud documentation, updated 22 October 2026, viewed 24 October 2026. https://docs.cloud.google.com/billing/v1/how-tos/catalog-api

23. Method: services.skus.list, Cloud Billing Catalog API v1 reference. Google Cloud documentation, viewed 24 October 2026. https://docs.cloud.google.com/billing/docs/reference/rest/v1/services.skus/list

24. Get Google Cloud pricing information (Pricing API). Google Cloud documentation, updated 22 October 2026, viewed 24 October 2026. https://docs.cloud.google.com/billing/docs/how-to/get-pricing-information-api

25. VM instance pricing, Compute Engine. Google Cloud, viewed 24 October 2026. https://cloud.google.com/products/compute/pricing

26. GPU machines in accelerator-optimized machine family, Compute Engine documentation. Google Cloud documentation, updated 22 October 2026, viewed 24 October 2026. https://docs.cloud.google.com/compute/docs/accelerator-optimized-machines

27. Lambda Cloud API OpenAPI specification, version 1.10.0. Lambda, viewed 24 October 2026. https://docs.lambda.ai/api/cloud/spec.json

28. GPU cloud pricing, archived capture of 25 October 2026 showing the rates viewed on 24 October 2026. Lambda, via the Internet Archive Wayback Machine, viewed 7 November 2026. https://web.archive.org/web/20261025040216/https://lambda.ai/pricing

29. Estimating Monthly Costs: Accessing List Pricing for OCI Products. Oracle Cloud Infrastructure documentation, viewed 24 October 2026. https://docs.oracle.com/en-us/iaas/Content/Billing/Tasks/signingup_topic-Estimating_Costs.htm

30. Oracle list price API response, all products in USD, with a single-part request for B98415 in all currencies. Oracle, retrieved 24 October 2026, 18:03 UTC. https://apexapps.oracle.com/pls/apex/cetools/api/v1/products/?currencyCode=USD

31. Compute Shapes. Oracle Cloud Infrastructure documentation, viewed 24 October 2026. https://docs.oracle.com/en-us/iaas/Content/Compute/References/computeshapes.htm

32. FOCUS Specification v1.4, List Unit Price. FinOps Open Cost and Usage Specification, viewed 24 October 2026. https://focus.finops.org/docs/specification/v1-4/columns/cost-and-usage/list-unit-price/

33. FOCUS Specification v1.4, Pricing Category. FinOps Open Cost and Usage Specification, viewed 24 October 2026. https://focus.finops.org/docs/specification/v1-4/columns/cost-and-usage/pricing-category/

34. FOCUS Specification v1.4, SKU Price Details. FinOps Open Cost and Usage Specification, viewed 24 October 2026. https://focus.finops.org/docs/specification/v1-4/columns/cost-and-usage/sku-price-details/

35. FOCUS 1.5 Release Scope: Confirmed, In Review and Deferred. FOCUS Working Group, viewed 24 October 2026. https://focus.finops.org/focus-1-5-release-scope/

36. NVIDIA H100 GPU, product specifications. NVIDIA Corporation, viewed 24 October 2026. https://www.nvidia.com/en-us/data-center/h100/

37. AI cloud pricing. Lambda, viewed 7 November 2026. https://lambda.ai/pricing

About this report

Published by Rillor on 7 November 2026. Superintelligence (SI) tools assist research and drafting, and every figure cites its source. Rillor builds and secures SI systems, supplies the data they learn from and delivers the infrastructure they run on. We apply SI to financial and compute markets.

Cite as: Rillor. Reading GPU prices from cloud pricing APIs. 7 November 2026. https://rillor.com/insights/gpu-prices-from-cloud-pricing-apis

Notices

Rillor is not a registered investment adviser or commodity trading advisor and does not provide investment or trading advice. Research, forecasts, data and software described on this site are for research and engineering use. Nothing here is an offer or recommendation to buy or sell any security, commodity interest or digital asset. Past or simulated results do not indicate future results.

NVIDIA, AMD, Ceph, Kubernetes and related product names are trademarks of their owners and are used only to identify products. No affiliation or endorsement is implied.

Work with Rillor

Send the GPU classes, regions and pricing types you need, and how often you need them. Contact us at rillor.com/contact?type=index or contact@rillor.com. More on GPU and neocloud pricing: rillor.com/markets/gpu-pricing