AWS Decision guide

Choosing a purchasing option for Amazon EC2

Choosing a purchasing option for Amazon EC2: AWS Decision guide

Table of Contents

Choosing a purchasing option for Amazon EC2

Add a subheading if desired, or remove.

Purpose	Help determine which Amazon EC2 purchasing option is the best fit for your budget and business needs.
Last updated	June 22, 2026
Covered purchasing options	• EC2 On-Demand Instances • EC2 Savings Plans • EC2 Spot Instances • EC2 Reserved Instances

Introduction

Amazon EC2 offers scalable computing capacity in the AWS cloud that enables you to develop and deploy applications faster without upfront hardware investments. As a cornerstone of the AWS cloud computing platform, Amazon EC2 offers the flexibility to launch any number of virtual servers, configure security and networking, manage storage, and scale resources up or down based on changing demand.

To accommodate diverse workloads, AWS offers multiple Amazon EC2 purchasing options spanning different operational requirements and preferences. When considering purchasing options for Amazon EC2, you must weigh several factors, including budget, demand and workload patterns, term commitment flexibility, and configuration conditions, such as region requirements. Each purchasing option helps you optimize costs, performance, and operational efficiency based on specific use cases and instance needs.

In this guide, we explain the available Amazon EC2 purchasing options, walk through the key factors to consider when choosing between them, and provide resources to help you get started with your selected approach.

Understanding Amazon EC2 purchase offerings

Amazon EC2 offers multiple purchasing options so you can balance cost, commitment, and flexibility for each workload. Some options trade upfront commitment for deeper discounts, while others preserve flexibility at a higher rate. Combining them strategically across your portfolio reduces overall spend without compromising performance or availability. The following sections describe each option.

Before you continue, note that two concepts are often confused but are independent. Purchasing options (On-Demand, Savings Plans, Reserved Instances, Spot) determine how AWS calculates your bill for the compute you use. Capacity Reservations (On-Demand Capacity Reservations and Capacity Blocks for ML) determine whether capacity is held for you when you need it, regardless of whether you use it. These two dimensions are independent, and you commonly layer them: a Capacity Reservation is billed at the equivalent On-Demand rate whether or not you consume the capacity. You can then discount that bill by combining the reservation with Savings Plans or a Regional Reserved Instance. The sections that follow cover purchasing options first, then dedicated hardware, and finally Capacity Reservations as a complementary capacity-availability tool.

Primary purchasing options

- **On-Demand Instances** provide pay-as-you-go pricing with no long-term commitments or upfront payments. They offer maximum flexibility, allowing you to start, stop, and terminate instances at any time while paying only for the compute time you use, with billing granularity as low as per-second for Linux, RHEL, and Windows instances.
- **Savings Plans** are a flexible pricing model that offers significant discounts in exchange for a commitment to a specific amount of compute usage (measured in $/hour) for a one or three-year period. Savings Plans relevant to Amazon EC2 come in two varieties: Compute Savings Plans automatically apply to any Amazon EC2 instance usage regardless of Region, instance family, operating system, or tenancy, and also cover Fargate and Lambda usage. Amazon EC2 Instance Savings Plans apply to a specific instance family within a Region, offering higher discounts in exchange for a more targeted commitment.

> **Note**
>
> There are four Savings Plan types: Compute Savings Plans (up to 66% off), Amazon EC2 Instance Savings Plans (up to 72% off), Database Savings Plans (up to 35% off for Amazon RDS, Aurora, DynamoDB, ElastiCache, and more), and SageMaker AI

Savings Plans (up to 64% off). This guide covers only the Savings Plans relevant to Amazon EC2.

- **Spot Instances** provide access to interruptible compute capacity based on demand at up to 90% discount. This capacity is made available by Amazon EC2 and can be reclaimed by AWS when necessary. Spot Instances are ideal for fault-tolerant, flexible workloads such as batch processing, data analytics, CI/CD pipelines, and stateless web servers. They work best when your applications can handle potential interruptions or when used with services that can automatically request replacement capacity. As a general guideline, Spot Instances are best suited for flexible workloads with variable demand that can tolerate interruptions, whereas Savings Plans are a better fit for steady-state workloads with predictable, consistent usage patterns where uninterrupted availability is required. Spot Instance pricing is independent of Savings Plans - Savings Plans discounts do not apply to Spot Instance usage.
- **Reserved Instances (RIs)** are an older purchasing option that AWS generally recommends replacing with Savings Plans, and newer Amazon EC2 instance families may not support Reserved Instance pricing. The set of services where RIs remain the only commitment-based discount has narrowed sharply: most database and analytics services are now covered by Database Savings Plans (Amazon RDS, Aurora, ElastiCache, DynamoDB, DocumentDB, Neptune, Keyspaces, Timestream, DMS, and OpenSearch, up to 35% off). Redshift uses its own commitment models rather than Savings Plans: Reserved Nodes for provisioned clusters (up to 75% off) and Serverless Reservations for Redshift Serverless (up to 45% off). For most workloads, Savings Plans (or the relevant service-specific reservation) are now the default.

Dedicated hardware

If you have compliance, regulatory, or operational requirements that require your instances to run on Amazon EC2 hardware dedicated to you, two options are available:

- **Dedicated Hosts** provide physical Amazon EC2 servers dedicated exclusively to your use. They offer visibility into the underlying server's physical cores and sockets, enabling you to deploy your instances on specific servers and maintain them there over time. Dedicated Hosts can help meet compliance requirements that may necessitate physical server isolation and provide more predictable performance. This option is particularly valuable for bringing your own existing server-bound software licenses (such as Windows Server, SQL Server, or SUSE Linux Enterprise Server) that are licensed per core or per socket, allowing you to reduce costs by avoiding the license fees normally included in standard Amazon EC2 pricing. Dedicated

Hosts are priced per physical host rather than per instance, so cost efficiency depends on how fully you utilize the available instance capacity on each host.

- **Dedicated Instances** run on hardware dedicated to a single AWS account but differ from Dedicated Hosts in that they do not provide visibility or control over instance placement. They offer physical isolation at the host hardware level from instances belonging to other AWS accounts, making them suitable for workloads with regulatory or compliance requirements that mandate tenant isolation.

Dedicated hardware pricing options include On-Demand rates, Dedicated Host Reservations, and coverage by Savings Plans. For more information, refer to the Amazon EC2 pricing page.

Capacity Reservations

Capacity Reservations allow you to reserve compute capacity in specific Availability Zones for any duration without making long-term commitments. You can create a Capacity Reservation at any time, and you can choose whether it starts immediately or at a future date. There are two types:

- **On-Demand Capacity Reservations (ODCR)** provide flexibility for workloads that have strict capacity requirements, such as business-critical events, high-availability requirements, regulatory compliance, or disaster recovery scenarios. They can be created and canceled as needed. Capacity Reservations can be configured along two dimensions:

  By access scope:

  - **Open Capacity Reservations:** Open Capacity Reservations are shared across your organization and available for any member account to use with instances that match the reservation attributes, such as instance type, platform, and Availability Zone.
  - **Targeted Capacity Reservations:** Only instances that have matching attributes (instance type, platform, Availability Zone, and tenancy) and that explicitly target the reservation can use the reserved capacity, providing more granular control over capacity allocation.

  By start time:

  - **Immediate Capacity Reservations:** Start within minutes upon creation and reserve capacity right away, ensuring instant availability for your workloads.
  - **Future-dated Capacity Reservations:** Start at a specified future date and time, with a required commitment duration, allowing you to plan and reserve capacity in advance for

scheduled events or anticipated demand spikes. The minimum commitment duration is 14 days.

Capacity Reservations are priced at the equivalent On-Demand rate regardless of whether the reserved capacity is used, and can be combined with Savings Plans or Regional Reserved Instances to reduce costs.

- **Capacity Blocks for ML** enable you to reserve GPU and accelerated instances for specific time windows, particularly valuable for ML workloads with predictable scheduling needs. Capacity Blocks can be reserved for durations ranging from 1 to 182 days, with the option to start within minutes from purchase or at a scheduled future date. This option is ideal for model training, fine-tuning, experimentation, and handling temporary inference demand spikes. Capacity Blocks are priced based on supply and demand trends, with rates updated periodically, and the reservation fee is charged upfront at the time of purchase. Pricing is not a discount mechanism: rates can be at, above, or below On-Demand depending on instance family, region, and demand, and for popular GPU instances Capacity Blocks often carry a premium over On-Demand as the price of assured availability. Unlike On-Demand Capacity Reservations, Capacity Block pricing cannot be combined with Savings Plans or Reserved Instance discounts.

Consider

> **Tip**
>
> **Tip:** If you are new to Amazon EC2 or are not sure about your usage patterns, consider starting with On-Demand Instances and using tools such as AWS Compute Optimizer and AWS Cost Explorer to monitor and analyze your usage before committing to a specific purchasing option. For more information, see the AWS Cost Optimization documentation and Tips for Right Sizing.

When selecting Amazon EC2 purchasing options, evaluate these key factors:

Commitment or term

Think of Amazon EC2 purchasing options as existing on a spectrum from pay-as-you-go flexibility to longer-term commitments of up to three years, with increasing cost savings as the commitment duration grows:

- **No commitment with On-Demand or Spot:** Pay-as-you-go pricing with no term and no upfront payment.
  - On-Demand suits uncertain, temporary, or variable needs that require uninterrupted capacity.
  - Spot offers up to 90% off the same zero-commitment model, in exchange for interruptibility. Best suited to fault-tolerant, scalable, or bursty workloads that can resume elsewhere when capacity is reclaimed.
- **Flexible commitment with Compute Savings Plans:** Commit to a specific hourly spend (in $/hour) for a 1- or 3-year term for up to 66% off. The commitment applies flexibly across instance families, Regions, operating systems, and tenancy, and also covers Fargate and Lambda.
- **Targeted commitment with Amazon EC2 Instance Savings Plans:** Commit to a specific hourly spend on a chosen instance family within a Region for up to 72% off. Deeper discount than Compute Savings Plans, in exchange for reduced flexibility. Reserved Instances offer a similar trade-off but are a legacy option; consider them only for services not yet covered by any Savings Plan type (such as Redshift Reserved Nodes for provisioned clusters).

Commitment as strategy

Commitment decisions do not need to be one-time decisions. Start by analyzing past usage to identify always-on services with stable, predictable baselines. These are strong candidates for Savings Plans coverage today. From there, you can layer additional commitments as new services stabilize, top up coverage as baselines grow, and choose between 1- and 3-year terms based on your confidence in each tranche.

Within each tranche, match the plan type to how stable the workload is. Amazon EC2 Instance Savings Plans offer the deepest discounts when you are confident you will stay on the same instance family and Region for the term. Compute Savings Plans trade some discount for the freedom to shift across instance families, Regions, and even Fargate and Lambda, which is the safer choice when architectural change is likely or when you want to take advantage of newer instance types that typically offer better price/performance over time.

In practice, a single purchasing option is rarely optimal. You can layer them: Savings Plans or other committed options for the predictable baseline, and On-Demand or Spot Instances for variable demand on top. See the Hybrid Strategy section for how to blend them.

Continuity and demand

Understanding how critical a workload is and how its demand varies over time can help determine which purchasing options best serve it.

Workload criticality

Match the purchasing option to how critical the workload is:

- **Mission-critical systems:** For applications where downtime means significant business impact (payment processing, trading platforms, core customer experiences), layer a capacity assurance mechanism on top of your chosen purchasing option. On-Demand Capacity Reservations provide assurance that the required resources are available when needed, and you can combine them with Savings Plans or Regional Reserved Instances to optimize the cost of that reserved capacity.
- **Important but resilient systems:** For services that matter but have built-in resilience (distributed systems with redundancy), a balanced approach using a mix of Savings Plans and On-Demand provides both cost efficiency and reliability, or Spot Instances where the workload is fault-tolerant.
- **Background or batch processing:** For workloads that can be delayed or rescheduled without significant impact, Spot Instances offer significant cost savings, as the occasional interruption presents minimal business risk.

> **Note**
>
> Where the workload is fault-tolerant, Spot Instances can further reduce costs.

Demand patterns

Your workload's demand pattern can be used to guide your Amazon EC2 purchasing decisions:

- **Steady-state workloads:** Applications with consistent, predictable usage patterns (databases, core infrastructure) benefit most from long-term commitments through Savings Plans.
- **Cyclical patterns:** Workloads with predictable but variable patterns (day/night cycles, weekly patterns, seasonal spikes) benefit from a layered approach: Savings Plans for the baseline and On-Demand Capacity Reservations or Capacity Blocks for predictable peaks.

- **Unpredictable bursts:** Systems with random or unpredictable spikes can benefit from maximum flexibility through On-Demand or Spot capacity.

Cost of readiness

Maintaining reserved but unused capacity represents a strategic choice between immediate availability and cost efficiency. Consider not just the direct cost of idle resources, but also the business impact of delayed capacity availability. For mission-critical use cases, reserving capacity through On-Demand Capacity Reservations even when not fully utilized can be the right business decision. To reduce the cost of readiness, combine Capacity Reservations with Savings Plans to apply billing discounts to both used and unused reserved capacity.

Cost considerations

Cost considerations for Amazon EC2 extend beyond price comparisons. Understanding the full economic impact of your Amazon EC2 purchasing decisions requires a multidimensional perspective.

The broader cost equation

The visible price of an Amazon EC2 instance is just one component of its true cost. When evaluating options, consider the following:

- **Operational overhead:** Savings Plans offer a straightforward path to cost optimization, discounts apply automatically with minimal ongoing management beyond periodic reviews of your commitment level. Reserved Instances can yield higher savings for specific configurations but require more active management of instance attributes, modifications, and renewals.
- **Opportunity cost:** Capital committed to upfront payments cannot be invested elsewhere. Consider your cost of capital and alternative uses for those funds when evaluating payment options.
- **Risk-adjusted cost:** The lowest nominal price is not always the lowest risk-adjusted cost. Spot Instances offer significant savings but introduce operational overhead that could result in increased engineering effort spent building resilience, or costs from service disruptions if not implemented correctly or not applied for the right workload.

When multiple commitment types are active, AWS applies them in a fixed order: Reserved Instances first, then Amazon EC2 Instance Savings Plans, then Compute Savings Plans, which may affect how discounts are allocated across your usage.

Strategic payment timing

Payment structure decisions should align with your business's financial strategy:

- **Maximizing cost efficiency:** If you have available capital and focus on maximum efficiency, you benefit most from All Upfront payments when available, effectively prepaying for compute at the highest discount.
- **Balanced approach:** Partial Upfront payments offer a middle ground, reducing the monthly commitment while still capturing significant discounts, ideal for most established businesses.
- **Growth-focused, cash-preserving:** If you prioritize growth and cash flow preservation, you might benefit from No Upfront options, accepting slightly lower discounts to maintain capital flexibility for business expansion.

Granular efficiency

Per-second billing fundamentally changes the economics of short-duration workloads. Tasks that run for minutes rather than hours (batch processing, CI/CD pipelines, data transformations) can achieve dramatic cost reductions through precise resource allocation and immediate termination when work completes. This granularity rewards architectural patterns that emphasize rapid startup, efficient processing, and prompt shutdown.

Instance flexibility

Instance flexibility represents your ability to adapt to changing requirements, new technologies, and evolving best practices. The value of flexibility varies dramatically based on your pace of change and innovation.

Flexibility as strategic value

Flexibility is a strategic asset with tangible business value:

- **High-innovation environments:** If you are rapidly evolving your architecture, adopting new services, or frequently changing workload characteristics, prioritize flexibility with Compute Savings Plans even at slightly lower discounts in exchange for the ability to adopt new, more efficient instance types or shift workloads between services, which often delivers greater long-term value than maximum short-term discounts.

- **Stable, mature workloads:** Applications with well-understood, stable requirements can lock in deeper discounts through Amazon EC2 Instance Savings Plans (or Reserved Instances, where applicable), trading scope for savings, since significant changes are unlikely during the commitment period.

Geographic flexibility

Geographic flexibility considerations extend beyond simple regional versus zonal decisions:

- **Discount portability:** Compute Savings Plans apply across all Regions, so the discount follows the workload, whether it rotates between Regions daily (follow-the-sun), fails over for resilience, or shifts geographically over time. Amazon EC2 Instance Savings Plans are scoped to a specific Region, offering higher discounts in exchange for that geographic commitment.
- **Capacity across Availability Zones:** On-Demand Capacity Reservations are tied to a specific Availability Zone. For multi-AZ architectures or disaster recovery scenarios, consider creating Capacity Reservations across multiple Availability Zones to ensure failover capacity is available when needed.
- **Data gravity:** Compute resources tend to remain in the same Region as the data they process. Cross-Region egress costs, replication lag, locality requirements (Amazon S3, Amazon RDS, DynamoDB, and other services), and storage lifecycle commitments anchor stateful workloads to a specific Region. For data-heavy workloads, geographic flexibility matters less because the workload is unlikely to move. Stateless or compute-only workloads can shift more freely and benefit more from Region-portable discounts.
- **Regulatory evolution:** Data residency and compliance requirements continue to evolve globally. If you operate in multiple jurisdictions or face changing regulatory landscapes, you might need the flexibility to relocate workloads to different regions as requirements change.

Specialized requirements as constraints

Workloads pinned to specific hardware (particular CPU architectures, GPUs, accelerators, or memory configurations) have little flexibility to shift across instance families, which actually makes commitment-based discounts attractive: you are not giving up flexibility you would have used. For steady use of a known family in a Region, Amazon EC2 Instance Savings Plans (or Reserved Instances, where applicable) deliver the deepest discount. For instance types where capacity availability is limited and launch failure is unacceptable, layer On-Demand Capacity Reservations on top to reserve capacity, and discount the resulting On-Demand-equivalent

charges with Savings Plans or a Regional Reserved Instance. For scheduled GPU or accelerated workloads such as ML training and fine-tuning, Capacity Blocks for ML reserve accelerator capacity for a defined window and bill only for that window.

Usage predictability

Usage predictability fundamentally shapes your optimal purchasing strategy. Understanding the patterns, confidence levels, and variability in your workloads enables more strategic decisions about commitments and flexibility.

Predictability spectrum

Workloads exist on a predictability spectrum that directly influences optimal purchasing strategies:

- **Highly predictable workloads:** More infrastructure, databases, and steady-state applications with consistent, well-understood usage patterns represent ideal candidates for maximum commitment through Savings Plans. The predictability of these workloads transforms what would otherwise be a risk (long-term commitment) into an opportunity for substantial savings.
- **Seasonally predictable workloads:** Applications with identifiable patterns that vary by time of day, day of week, or season benefit from a layered approach. The predictable baseline is ideal for Savings Plans, while the predictable peaks might leverage On-Demand Capacity Reservations or Capacity Blocks for ML workloads.
- **Growing but predictable workloads:** For applications with steady growth trajectories, staggered commitments can be effective. Rather than committing to the full capacity at once, implement a rolling commitment strategy where portions of your capacity are committed at different times, creating a ladder effect that accommodates growth while maintaining discounts.
- **Unpredictable workloads:** Highly variable or unpredictable workloads benefit from maximum flexibility through On-Demand or Spot Instances.

Architectural adaptability

The design of your applications significantly impacts your ability to leverage different pricing models:

- **Stateless versus stateful:** Stateless applications that can easily scale horizontally across diverse instance types enable much greater purchasing flexibility than stateful applications with specific instance requirements.
- **Resilience patterns:** Applications designed with resilience patterns (circuit breakers, retry mechanisms, graceful degradation) can more effectively utilize Spot Instances, potentially accessing the deepest discounts available.
- **Resource efficiency:** Applications optimized for efficient resource utilization can often run on smaller or more diverse instance types, increasing flexibility in purchasing options and reducing overall costs. For example, a containerized microservices architecture that right-sizes each service independently can spread workloads across multiple smaller instance types rather than requiring a few large, specialized ones enabling broader use of Spot Instances and Compute Savings Plans.

Workload patterns for large customers

At scale, purchasing decisions are not based on a single workload. Instead, you must consider the combined requirements of multiple workloads distributed across different AWS accounts, Regions, and time zones. Patterns emerge at scale that create unique optimization opportunities:

- **Portfolio effect:** Across hundreds or thousands of workloads, individual spikes and dips tend to cancel out, producing a steadier aggregate baseline than any single workload would show. That stability supports larger Savings Plans commitments with less forecasting risk.
- **Cross-account optimization:** If you use consolidated billing, you can optimize across multiple AWS accounts by centralizing purchasing decisions through AWS Organizations. This allows Savings Plans and Regional Reserved Instance discounts to be shared across accounts, maximizing utilization rates. On-Demand Capacity Reservations can also be shared across accounts using AWS Resource Access Manager. For shared Capacity Reservations, billing for unused capacity can be assigned to a specific consumer account, helping you align costs with the teams that requested the capacity. Additionally, Interruptible Capacity Reservations allow teams to make unused reserved capacity temporarily available to other workloads within the organization the capacity owner retains control to reclaim it when needed, while other teams can use it for fault-tolerant workloads in the interim.
- **Seasonal aggregation:** While individual applications may have unpredictable seasonal patterns, large portfolios often exhibit more stable aggregate seasonal trends. Retail

workloads peak during holidays, while educational workloads peak during enrollment periods. Understanding these combined patterns enables more strategic capacity planning.

- **Risk distribution:** At scale, you can implement more aggressive cost optimization strategies because you can distribute risk across your portfolio. A portion of workloads can leverage maximum Spot Instance usage while others maintain reserved capacity, achieving overall cost optimization while maintaining business continuity.

Choose

> **Tip**
>
> Before committing to a purchasing option, ensure your instances are right-sized using AWS Compute Optimizer. Oversized instances waste money regardless of the purchasing model you choose.

Selecting the right EC2 purchasing strategy requires balancing financial considerations with operational requirements. Most organizations implement a hybrid approach, combining multiple purchasing options to optimize both cost and performance. A common pattern is to use commitment-based options such as Savings Plans for baseline capacity, On-Demand or Spot Instances for fluctuating workloads, and On-Demand Capacity Reservations for mission-critical applications that require capacity assurance.

The following table will help you determine the broader purchasing options based on your specific requirements and usage patterns:

	Instant Requests		Discount Vehicles		Capacity Assurance	
	On-Demand Instances	Spot Instances	Compute Savings Plans	Amazon EC2 Instance Savings Plans	On-Demand Capacity Reservations	Capacity Blocks for ML
Term	No commitment (60-secon	No commitment	1-year or 3-year hourly spend commitment		No commitment required	Defined reservation window

	d billing minimum)		for immediate-use. Future-dated CRs require a commitment duration after start date	(up to 6 months duration)
Minimum duration	None	1 year	None (immediate-use), minimum 14 days if future-dated	Minimum 1 day. 1-day increments up to 14 days, 7-day increments up to 6 months. Reservation offerings available starting as soon as 30 minutes
Capacity benefit	No capacity reserved		Open-ended reservation in a specific AZ (instance type + tenancy + platform fixed)	Time-boxed reservation in a specific AZ for a defined window

Billing discount	No discount (baseline pricing)	Up to 90% discount vs On-Demand	Up to 66% discount vs On-Demand	Up to 72% discount vs On-Demand	No billing discount (can combine with Savings Plans or Regional RIs for discounts)	Prices based on supply and demand, updated periodically
Flexibility	High, any available instance type		Highest: works across Amazon EC2, Fargate, Lambda; any region, AZ, instance family, OS, tenancy	Specific instance family and region	AZ-specific, exact instance type matching	Limited to supported GPU/accelerator instance types

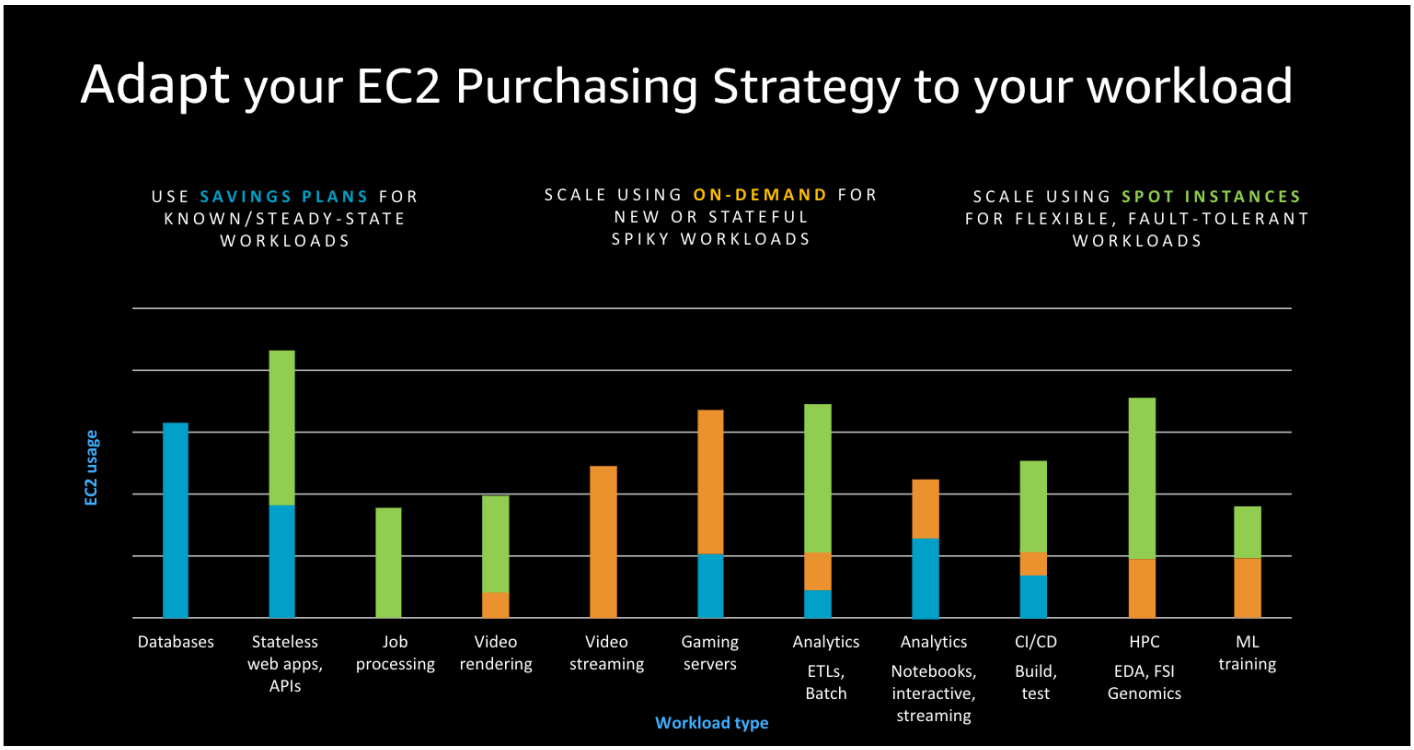
Payment options	Per-second (60s min) for Linux, Windows, Windows+SQL, RHEL, Ubuntu, Ubuntu Pro; per-hour for SUSE		No Upfront, Partial Upfront, All Upfront	Per-second On-Demand-rate billing for the entire life of the reservation, whether or not an instance is launched into it	Reservation fee charged upfront at time of purchase + OS fee while running
Availability assurance	Available subject to capacity	Subject to availability of unused Amazon EC2 capacity; can be interrupted	No capacity assurance	Capacity reserved in specified AZ	Capacity reserved for the scheduled window

Interruption risk	No interruption	Can be interrupted with 2-minute notice; instance is terminated by default (or stopped/hibernated if configured at launch)	No interruption		No interruption during reservation window
Billing start	When Amazon EC2 instance enters running state		When purchased (or on queued start date)	When reservation becomes active	Upfront at time of purchase (reservation fee); OS billed when instances run

Modification flexibility	No commitment terms to modify	Existing plans cannot be modified; can queue future purchases		Immediate-use: modify count/ instance attributes, split, or share via RAM. AZ binding is fixed at creation. Future-dated CRs restricted during commitment period	Cannot cancel or modify after purchase. Can purchase consecutive blocks (subject to availability).
Regional scope	All regions/AZs (instance type availability varies)	Global (any region any AZ)	Specific region	Single AZ only	Single AZ only (select regions)
Capacity planning	No planning required	Specify dollar commitment ($/hr)		Must specify instance type, count, AZ, platform, and tenancy	Specify instance type, count (1-64), start date, and duration

Sharing across AWS accounts	Not applicable	Automatically applies across all accounts in AWS Organizations (by default)	AWS Resource Access Manager	Instance CBs: Yes (via AWS RAM, same Org). UltraServer CBs: No
Queuing support	Not applicable	Yes (up to 3 years ahead)	Yes - future-dated CRs up to 120 days ahead (min 14-day commitment)	Reserve up to 8 weeks in advance
Cancellation policy	Can terminate anytime	Cannot cancel (7-day return for plans ≤$100/hr)	Immediate-use: can cancel anytime. Future-dated: cannot cancel during commitment period	Cannot cancel after purchase

Best use cases	New applications; short-term, irregular workloads that cannot be interrupted; proof of concepts	Batch processing; Fault-tolerant applications; CI/CD pipelines; Workloads flexible on time or instance type; Containerized workloads	Mixed compute usage (Amazon EC2, Fargate, Lambda); Multi-region workloads; Modernization initiatives; Organizations wanting simplest commitment model	Consistent Amazon EC2 usage within specific instance family; Regional workloads with predictable instance needs	Critical workloads needing capacity assurance; Seasonal capacity planning; Disaster recovery; Compliance requirements	ML model training and fine-tuning; ML experiments and prototypes; Temporary GPU/accelerator capacity needs
Combination strategy	Use for burst or variable capacity	Use as supplement to committed or On-Demand capacity. Best practice: diversify across instance types and AZs	Can combine with On-Demand Capacity Reservations for capacity assurance		Combine with Savings Plans or Regional RIs for billing discounts	Cannot combine with Savings Plans or RI discounts; standalone pricing



Hybrid EC2 purchasing strategy: Accelerating workloads with Spot without risking deadlines

> **Important**
>
> Adopt a hybrid EC2 purchasing approach: use On-Demand or committed options such as Savings Plans for baseline capacity that meets your deadlines, then supplement with Spot Instances to accelerate performance and reduce costs.

A common pattern is to define your workload's completion deadline, then provision baseline capacity using On-Demand Instances or committed options such as Savings Plans that can independently meet this timeline. During runtime, supplement this foundation with Spot Instances to accelerate completion. For example, adding Spot capacity equal to your baseline can reduce completion time significantly while lowering overall costs, even if some Spot capacity is reclaimed. This approach provides workload reliability through assured baseline capacity while capturing additional cost savings through Spot supplementation, particularly effective for batch processing, data analytics, and other fault-tolerant workloads.

Use your purchasing options

After you have determined which Amazon EC2 purchasing options best fit your workload requirements, the following resources can help you get started—from launching instances and reserving capacity to monitoring, analyzing, and optimizing your costs over time.

Purchase & Setup

- **Get started with the Amazon EC2 tutorial**

  Learn how to launch and configure your first Amazon EC2 instance using the Amazon EC2 launch instance wizard in the Amazon EC2 console.

  Get started with the Amazon EC2 tutorial

- **Launch On-Demand Instances**

  Provision instances at the standard per-second On-Demand rate with no upfront commitment, using the launch wizard, CLI, or SDK.

  Get started with Amazon EC2 instances

- **Purchase Savings Plans**

  Commit to a consistent amount of compute usage ($/hour) for a 1-year or 3-year term.

  Explore the guide

- **Create a Spot Instance request**

  Request spare Amazon EC2 capacity at discounts of up to 90% off On-Demand prices; instances can be interrupted with a two-minute notice when AWS needs the capacity back.

  Read the guide

- **Create an Auto Scaling group**

  Configure an Auto Scaling group that combines On-Demand and Spot Instances using a single configuration. Define your instance type diversification, allocation strategies, and the proportion of On-Demand to Spot capacity. Auto Scaling handles provisioning, scaling, and automatic replacement of interrupted Spot Instances.

  Read the guide

Monitoring & Analysis

Monitor your instance performance and usage patterns for opportunities to right-size your Amazon EC2 instances.

- **Monitor Savings Plans utilization and coverage**

  Regularly check your Savings Plans utilization and coverage metrics to ensure you are maximizing the value of your commitments.

  Read the guide

- **Build Custom CloudWatch dashboards**

  Visualize your Amazon EC2 cost and utilization and application metrics in one place with customizable dashboards with custom widgets that help identify trends and optimization opportunities for more informed decision-making.

  Read the guide

- **Monitor and optimize capacity with Amazon EC2 Capacity Manager**

  Use Amazon EC2 Capacity Manager to monitor, analyze, and manage your On-Demand, Spot, and Capacity Reservation usage across all accounts and Regions from a single interface. Identify underutilized Capacity Reservations, analyze usage patterns, and take action to optimize your Amazon EC2 capacity and costs.

  Read the guide

Scaling & Automation

- **Get started with Amazon EC2 Auto Scaling**

  Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application.

  Get started with the tutorial

- **Scheduled scaling**

  Set up time-based scaling for predictable workload patterns, such as business hours, weekends, or seasonal traffic variations.

Read the guide

- **Target tracking scaling policies**

  Learn how to automatically adjust your Amazon EC2 capacity based on demand metrics like CPU utilization, network traffic, or custom application metrics to maintain optimal performance and high utilization for your Amazon EC2 instances for better cost efficiency without manual intervention.

  Read the guide

Budget & Cost Controls

- **Set spending thresholds with AWS Budgets**

  Establish budget thresholds and automated alerts to maintain visibility and control over your Amazon EC2 spending.

  Read the guide

- **Resource tagging strategies**

  Implement consistent tagging across your Amazon EC2 resources early in your deployment to accurately track costs by department, project, environment, or other business dimensions, enabling detailed cost allocation and analysis.

  Read the guide

Right-sizing & Continuous Optimization

- **Use AWS Compute Optimizer to find right-size candidates**

  Learn how to monitor and analyze your current Amazon EC2 instance performance and usage patterns to strategize how you can right size your Amazon EC2 instances and optimize your costs over time.

  Read the guide

- **AWS Billing and Cost Management**

Use AWS Cost Optimization Hub to identify savings opportunities, including tailored Savings Plans recommendations based on your historical usage. Monitor your Savings Plans utilization and coverage in AWS Cost Explorer to maximize the value of your commitments over time.

Read the guide

Spot optimizations

AWS offers a broad and continuously expanding selection of Amazon EC2 instance types to fit virtually any workload. A set of unused Amazon EC2 instances with the same instance type (for example, m5.large) in a particular Availability Zone (for example, us-east-1a) constitute a single Spot capacity pool. When you request Spot capacity, AWS allocates compute resources from matching capacity pools that have unused Amazon EC2 instances in your specified Availability Zones. By including multiple instance types and Availability Zones in your Spot requests, you expand the number of capacity pools available to fulfill your request, significantly increasing your chances of obtaining the compute capacity you need.

AWS uses capacity-allocation strategies to launch instances from the deepest Spot capacity pools (pools with the largest spare capacity), which inherently reduces the likelihood of Spot interruptions. With Spot Instances, you can maximize compute cost savings while learning how to minimize the impact of interruptions through proper planning and implementation strategies.

Requesting and maintaining Spot capacity

- **Getting started with Spot Instances**

  Learn the fundamentals of Amazon EC2 Spot Instances, including how to request Spot capacity, understand pricing dynamics, and identify suitable workloads for this pricing model.

  Read the guide

- **Understanding pricing model**

  Spot Instances use a simplified pricing model where prices are set by Amazon Amazon EC2 and adjusted gradually based on long-term supply and demand trends. This approach provides predictable pricing without the complexity of bidding mechanisms that were used historically. If a Spot Instance is interrupted within the first hour of use, you are not charged for that partial hour of usage.

View current Spot prices | View pricing history | Savings calculator | Billing for interruptions

- **Requesting and maintaining Spot capacity**

  AWS provides APIs for requesting Spot capacity: RunInstances, Amazon EC2 Fleet, and Auto Scaling Groups. RunInstances offers basic functionality with limited options, while Amazon EC2 Fleet and Auto Scaling Groups provide extensive configuration capabilities for large-scale production deployments.

  RunInstances API | Amazon EC2 Fleet | Amazon EC2 Auto Scaling Group

- **Spot allocation strategies**

  The Spot allocation strategy determines how Amazon EC2 selects and prioritizes Spot capacity pools (combinations of instance types and Availability Zones) when launching Spot Instances. When launching instances, the API uses the allocation strategy that you specify to select the specific pools from all possible pools. There are five allocation strategies: Price capacity optimized, Capacity optimized, Capacity Optimized Prioritized, Diversified, and Lowest price (not recommended). These strategies have significant impact on which capacity pools will be selected to provision the requested capacity. In most cases, the recommended strategy is Price capacity optimized strategy, which balances cost and reduces the possibility of interruptions. This strategy requests Spot Instances from the pools that have the lowest chance of interruption in the near term, then requests Spot Instances from the lowest priced of these pools.

  Read the guide

- **Implement observability and monitoring**

  Implement observability and cost management practices to make data-driven decisions about workload suitability and cost savings maximization. Cost allocation tags enable detailed tracking of Spot workloads by categorizing them across teams, projects, and environments, helping quantify actual savings and identify new opportunities. Setting up budgets, alerts, and dashboards with tagged resources allows you to analyze interruption patterns, implement governance controls, and optimize your Spot strategy based on usage trends and cost metrics.

  Organizing and tracking costs using AWS cost allocation tags | Filtering the AWS costs data that you want to view | Savings from purchasing Spot Instances

Minimizing interruption risks

- **Instance diversification**

  Instance diversification is the most straightforward recommendation to implement that delivers the biggest improvement in Spot experience. Leverage a combination of Spot allocation strategies (such as Price Capacity Optimized) and broad instance types and Availability Zones to maximize access to Spot capacity. Reduce interruption risk by spreading your workload across multiple instance types, sizes, and Availability Zones. Applications that can run on diverse instance types have significantly higher availability and lower interruption rates.

- **Attribute-based instance selection**

  Attribute-based instance selection lets you specify compute requirements (vCPU, memory, architecture) rather than exact instance types. This allows your fleet to automatically select from a much larger pool of suitable instances, improving availability and Spot capacity access. It simplifies configuration while enabling broad diversification across instance families, sizes, and generations, including newly launched types.

  Read the guide

- **Implementing geographic flexibility**

  Expand your Spot Instance deployments across multiple Availability Zones and Regions to access more spare capacity and reduce the impact of capacity constraints in any single location. Use the Spot placement score API to identify the most promising Regions and Availability Zones for your workload requirements.

  Spot placement score | Spot placement score tracker

- **Flexibility on time**

  Schedule your Spot workloads during periods of lower demand to increase the likelihood of obtaining and maintaining Spot capacity with fewer interruptions.

  Read the guide

Minimizing interruption impacts

- **Leveraging notices for Amazon EC2 Spot interruptions**

Two minutes before a Spot Instance is interrupted by AWS, Amazon EC2 issues a Spot Instance interruption notice as an EventBridge event and as items in the instance metadata. Leverage the time to gracefully save state, drain connections, and redirect traffic to maintain application availability.

Spot Instance interruption notices | Taking Advantage of Amazon EC2 Spot Instance Interruption Notices

- **Responding to rebalance recommendations**

  EC2 rebalance recommendations signal when Spot Instances are at elevated interruption risk. This lets you proactively migrate workloads to new or existing Spot Instances. Auto Scaling Groups can automatically rebalance Spot capacity based on these recommendations.

  Amazon EC2 instance rebalance recommendations | Capacity rebalancing in auto scaling to replace at-risk Spot Instances

- **Reducing compute time of individual tasks**

  Leverage a divide-and-conquer approach to break large jobs into smaller, faster-completing tasks. This reduces the chance of interruption before task completion and minimizes interruption impact, as tasks that are completed within two minutes are unaffected by Spot interruptions.

  Explore the guide

- **Workload checkpointing**

  Implement a checkpointing mechanism to save your job's current state (files, progress, and parameters) to persistent storage at regular intervals, or when receiving a Spot interruption notice. You can resume work from the last saved checkpoint on a different Amazon EC2 instance.

  Checkpointing HPC Applications | Save Up to 90% with Long-Running HPC Jobs | Cost Optimization with Ansys LS-DYNA

Advanced Spot practices

- **Spot hibernation**

Use Spot Instance hibernation to save in-memory state during interruptions for faster recovery. Hibernated instances can only resume in the same capacity pool (instance type and Availability Zone), and if that pool never regains availability the instance may never resume. Consider whether indefinite waiting is acceptable versus restarting on new instances elsewhere. Hibernation comes with constraints: RAM under 150 GB, 60-day maximum duration before termination, EC2-only resumption, plus restrictions on instance family, root volume type, and encryption. See the link below for more details.

Read the guide

- **Workload prioritization**

  Implement workload prioritization so that critical tasks run on On-Demand or committed instances.

  Read the guide

- **Spot placement score**

  Maximize your chances of securing Amazon EC2 Spot Instances by leveraging the Spot placement score feature, which helps you identify the most promising Regions and Availability Zones for your workload requirements.

  Read the guide

- **Test Spot Instance interruptions using AWS FIS**

  Learn how to simulate Spot Instance interruptions using AWS Fault Injection Service (FIS) and build more reliable, cost-effective applications that can handle Amazon EC2 capacity reclamation gracefully.

  Get started with the tutorial

Explore

- **AWS Compute Optimizer**

  Get recommendations to optimize the performance and cost of your AWS resources.

  Explore the solution

- **AWS Well-Architected Cost Optimization Workshop**

This workshop contains hands-on-labs to help you learn, measure, and improve your architectures by optimizing costs using best practices from the Cost Optimization pillar of the AWS Well-Architected Framework.

Explore the workshop

Document history

The following table describes the important changes to this decision guide. For notifications about updates to this guide, you can subscribe to an RSS feed.

Change	Description	Date
Initial release	Initial release of the decision guide.	June 22, 2026