

Amazon Neptune FAQs

[aws.amazon.com \(https://aws.amazon.com/neptune/faqs/\)](https://aws.amazon.com/neptune/faqs/)

General

Q: What is Amazon Neptune?

Amazon Neptune is a fast, reliable, fully-managed graph database service that makes it easy to build and run applications that work with highly connected datasets. SQL queries for highly connected data are complex and hard to tune for performance. Instead, with Amazon Neptune you can use open and popular graph query languages to execute powerful queries that are easy to write and perform well on connected data. The core of Neptune is a purpose-built, high-performance graph database engine optimized for storing billions of relationships and querying the graph with milliseconds latency. You can use Neptune for graph use cases such as recommendation engines, fraud detection, knowledge graphs, drug discovery, and network security. Amazon Neptune is fully managed and handles the time-consuming tasks such as provisioning, patching, backup, recovery, failure detection and repair. You pay a simple monthly charge for each Amazon Neptune database instance you use. There are no upfront costs or long-term commitments required.

Q: What popular graph query languages does Amazon Neptune support?

Amazon Neptune supports both the open source Apache TinkerPop Gremlin graph traversal language and the W3C standard Resource Description Framework's (RDF) SPARQL query language.

Q: Can I use Apache TinkerPop Gremlin and RDF/SPARQL on the same Neptune instance?

Yes, each Neptune instance provides both a Gremlin Websocket Server and a SPARQL 1.1 Protocol REST endpoint. The data is separated between the stacks, i.e. you cannot execute a Gremlin traversal over RDF data or vice-versa. This is intended to allow you to experiment and see which one is the best fit for your application. In production, as the resources are shared on a single instance, we recommend clients accessing an instance via only one of the languages, i.e. Gremlin or SPARQL.

Q: How can I migrate from an existing Apache TinkerPop Gremlin application to Amazon Neptune?

Amazon Neptune provides an Apache TinkerPop (<http://tinkerpop.apache.org/>) Gremlin Server that supports both Websocket and REST connections. Once you provision an instance of Amazon Neptune, you can configure your existing TinkerPop application to use the endpoint provided by the service. See also [Accessing the Graph via Gremlin \(https://docs.aws.amazon.com/neptune/latest/userguide/access-graph-gremlin.html\)](https://docs.aws.amazon.com/neptune/latest/userguide/access-graph-gremlin.html).

Q: Do I need to change client drivers to use Amazon Neptune's Gremlin Server?

No, Amazon Neptune's Gremlin server supports clients that are compatible with Apache TinkerPop version 3.3 using both Websocket and REST connections.

Q: How can I migrate from a triple store with a SPARQL endpoint to Amazon Neptune?

Amazon Neptune provides an HTTP REST endpoint that implements the SPARQL 1.1 Protocol (<https://www.w3.org/TR/sparql11-protocol/>). Once you provision a service instance, you can configure your application to point to the SPARQL endpoint. See also Accessing the Graph via SPARQL. (<https://docs.aws.amazon.com/neptune/latest/userguide/access-graph-sparql.html>)

Q: Do I need to change client drivers to use Amazon Neptune's SPARQL Endpoint?

No, Amazon Neptune's SPARQL endpoint will work with any client that supports the SPARQL 1.1 Protocol.

Q: Is Neptune ACID (Atomicity, Consistency, Isolation, Durability) compliant?

Yes, Neptune is ACID compliant with immediate consistency.

Q: Why are Amazon RDS permissions and resources required to use Amazon Neptune?

Amazon Neptune is a purpose-built, high-performance graph database engine. For certain management features such as instance lifecycle management, encryption-at-rest with Amazon Key Management Service (KMS) keys and security groups management, Neptune leverages operational technology that is shared with Amazon RDS.

Q: Does Amazon Neptune have a service level agreement (SLA)?

Yes. Please see the Amazon Neptune SLA (<https://aws.amazon.com/neptune/sla/>).

Performance

Q: Do I need to create indices on my data with Amazon Neptune?

No, existing graph database users are often forced to try and outguess the vendor implementation. Explicitly maintaining indices is just one aspect of that. Amazon Neptune does not require you to create specific indices to achieve good query performance, and it minimizes the need for such second guessing of the database design.

Q: What types of graph query workloads are optimized to work with Amazon Neptune?

Amazon Neptune is designed to support graph applications that require high throughput and low latency graph queries. With support for up to 15 read replicas, Amazon Neptune can support 100,000s of queries per second.

Q: Does Amazon Neptune perform query optimization?

Yes, Amazon Neptune uses query optimization for both SPARQL queries and Gremlin traversals.

Q: Is Amazon Neptune built on a relational database?

No, Amazon Neptune is a purpose-built, high-performance graph database engine. Neptune efficiently stores and navigates graph data, and uses a scale-up, in-memory optimized architecture to allow for fast query evaluation over large graphs.

Pricing

Q: How much does Amazon Neptune cost?

Please see our pricing page for current pricing information.

Q. In which AWS regions is Amazon Neptune available?

Please see our pricing page for current information on regions and prices.

Q. Amazon Neptune replicates each chunk of my database volume six ways across three Availability Zones. Does that mean that my effective storage price will be three or six times what is shown on the pricing page?

No. Amazon Neptune's replication is bundled into the price. You are charged based on the storage your database consumes at the database layer, not the storage consumed in Amazon Neptune's virtualized storage layer.

Q: What are IOs in Amazon Neptune and how are they calculated?

IOs are input/output operations performed by Amazon Neptune against an SSD-based virtualized storage layer. Every database page read operation counts as one IO. Amazon Neptune issues reads against the storage layer in order to fetch pages not present in the buffer cache. Each page is 16KB in Amazon Neptune.

Amazon Neptune was designed to eliminate unnecessary IO operations in order to reduce costs and to ensure resources are available for serving read/write traffic. Write IOs are only consumed when pushing transaction log records to the storage layer for the purpose of making writes durable. Write IOs are counted in 4KB units. For example, a transaction log record that is 1024 bytes will count as one IO operation. However, concurrent write operations whose transaction log is less than 4KB can be batched together by the Amazon Neptune database engine in order to optimize I/O consumption. Unlike traditional database engines Amazon Neptune never pushes modified database pages to the storage layer, resulting in further IO consumption savings.

Hardware and Scaling

Q: What are the minimum and maximum storage limits of an Amazon Neptune database?

The minimum storage is 10GB. Based on your database usage, your Amazon Neptune storage will automatically grow, up to 64 TB, in 10GB increments with no impact to database performance. There is no need to provision storage in advance.

Q: How do I scale the compute resources associated with my Amazon Neptune DB Instance?

You can scale the compute resources allocated to your DB Instance in the AWS Management Console by selecting the desired DB Instance and clicking the Modify button. Memory and CPU resources are modified by changing your DB Instance class.

When you modify your DB Instance class, your requested changes will be applied during your specified maintenance window. Alternatively, you can use the "Apply Immediately" flag to apply your scaling requests immediately. Both of these options will have an availability impact for a few minutes as the scaling operation is performed. Bear in mind that any other pending system changes will also be applied.

Backup and Restore

Q: How do I enable backups for my DB Instance?

Automated backups are always enabled on Amazon Neptune DB Instances. Backups do not impact database performance.

Q: Can I take DB Snapshots and keep them around as long as I want?

Yes, and there is no performance impact when taking snapshots. Note that restoring data from DB Snapshots requires creating a new DB Instance.

Q: If my database fails, what is my recovery path?

Amazon Neptune automatically maintains 6 copies of your data across 3 Availability Zones and will automatically attempt to recover your database in a healthy AZ with no data loss. In the unlikely event your data is unavailable within Amazon Neptune storage, you can restore from a DB Snapshot or perform a point-in-time restore operation to a new instance. Note that the latest restorable time for a point-in-time restore operation can be up to 5 minutes in the past.

Q: What happens to my automated backups and DB Snapshots if I delete my DB Instance?

You can choose to create a final DB Snapshot when deleting your DB Instance. If you do, you can use this DB Snapshot to restore the deleted DB Instance at a later date. Amazon Neptune retains this final user-created DB Snapshot along with all other manually created DB Snapshots after the DB Instance is deleted. Only DB Snapshots

are retained after the DB Instance is deleted (i.e., automated backups created for point-in-time restore are not kept).

Q: Can I share my snapshots with another AWS account?

Yes. Neptune gives you the ability to create snapshots of your databases, which you can use later to restore a database. You can share a snapshot with a different AWS account, and the owner of the recipient account can use your snapshot to restore a DB that contains your data. You can even choose to make your snapshots public – that is, anybody can restore a DB containing your (public) data. You can use this feature to share data between your various environments (production, dev/test, staging, etc.) that have different AWS accounts, as well as keep backups of all your data secure in a separate account in case your main AWS account is ever compromised.

Q: Will I be billed for shared snapshots?

There is no charge for sharing snapshots between accounts. However, you may be charged for the snapshots themselves, as well as any databases you restore from shared snapshots. Learn more about Amazon Neptune pricing.

Q: Can I automatically share snapshots?

We do not support sharing automatic DB snapshots. To share an automatic snapshot, you must manually create a copy of the snapshot, and then share the copy.

Q: How many accounts can I share snapshots with?

You may share manual snapshots with up to 20 AWS account IDs. If you want to share the snapshot with more than 20 accounts, you can either share the snapshot as public, or contact support to increase your quota.

Q: In which regions can I share my Amazon Neptune snapshots?

You can share your Neptune snapshots in all AWS regions where Amazon Neptune is available.

Q: Can I share my Amazon Neptune snapshots across different regions?

No. Your shared Amazon Neptune snapshots will only be accessible by accounts in the same region as the account that shares them.

Q: Can I share an encrypted Amazon Neptune snapshot?

Yes, you can share encrypted Amazon Neptune snapshots.

Q: Can I use Amazon Neptune snapshots outside of the service?

No, Amazon Neptune snapshots can only be used inside of the service.

High Availability and Replication

Q: How does Amazon Neptune improve my database's fault tolerance to disk failures?

Amazon Neptune automatically divides your database volume into 10GB segments spread across many disks. Each 10GB chunk of your database volume is replicated six ways, across three Availability Zones. Amazon Neptune is designed to transparently handle the loss of up to two copies of data without affecting database write availability and up to three copies without affecting read availability. Amazon Neptune storage is also self-healing. Data blocks and disks are continuously scanned for errors and repaired automatically.

Q: How does Amazon Neptune improve recovery time after a database crash?

Unlike other databases, after a database crash Amazon Neptune does not need to replay the redo log from the last database checkpoint (typically 5 minutes) and confirm that all changes have been applied, before making the database available for operations. This reduces database restart times to less than 60 seconds in most cases. Amazon Neptune moves the buffer cache out of the database process and makes it available immediately at restart time. This prevents you from having to throttle access until the cache is repopulated to avoid brownouts.

Q: What kind of replicas does Neptune support?

Amazon Neptune supports Read Replicas, which share the same underlying volume as the primary instance. Updates made by the primary are visible to all Amazon Neptune Replicas.

Feature *Amazon Neptune Replicas*

Number of replicas Up to 15

Replication Type Asynchronous (milliseconds)

Performance impact on primary Low

Act as failover target Yes (no data loss)

Automated failover Yes

Q: Can I have cross-region replicas with Amazon Neptune?

No, Amazon Neptune does not support cross-region replicas.

Q: Can I prioritize certain replicas as failover targets over others?

Yes. You can assign a promotion priority tier to each instance on your cluster. When the primary instance fails, Amazon Neptune will promote the replica with the highest priority to primary. If there is contention between 2 or more replicas in the same priority tier, then Amazon Neptune will promote the replica that is the same size as the primary instance.

Q: Can I modify priority tiers for instances after they have been created?

You can modify the priority tier for an instance at any time. Simply modifying priority tiers will not trigger a failover.

Q: Can I prevent certain replicas from being promoted to the primary instance?

You can assign lower priority tiers to replicas that you don't want promoted to the primary instance. However, if the higher priority replicas on the cluster are unhealthy or unavailable for some reason, then Amazon Neptune will promote the lower priority replica.

Q: How can I improve upon the availability of a single Amazon Neptune database?

You can add Amazon Neptune Replicas. Amazon Neptune Replicas share the same underlying storage as the primary instance. Any Amazon Neptune Replica can be promoted to become primary without any data loss and therefore can be used for enhancing fault tolerance in the event of a primary DB Instance failure. To increase database availability, simply create 1 to 15 replicas, and Amazon Neptune will automatically include them in failover primary selection in the event of a database outage.

Q: What happens during failover and how long does it take?

Failover is automatically handled by Amazon Neptune so that your applications can resume database operations as quickly as possible without manual administrative intervention.

- If you have an Amazon Neptune Replica, in the same or a different Availability Zone, when failing over, Amazon Neptune flips the canonical name record (CNAME) for your DB primary endpoint to a healthy replica, which is in turn is promoted to become the new primary. Start-to-finish, failover typically completes within 30 seconds. Additionally, the read replicas endpoint doesn't require any CNAME updates during failover.
- If you do not have an Amazon Neptune Replica (i.e. single instance), Neptune will first attempt to create a new DB Instance in the same Availability Zone as the original instance. If unable to do so, Neptune will attempt to create a new DB Instance in a different Availability Zone. From start to finish, failover typically completes in under 15 minutes.

Your application should retry database requests in the event of connection loss.

Q: If I have a primary database and an Amazon Neptune Replica actively taking read traffic and a failover occurs, what happens?

Amazon Neptune will automatically detect a problem with your primary instance and begin routing your read/write

traffic to an Amazon Neptune Replica. On average, this failover will complete within 30 seconds. In addition, the read traffic that your Amazon Neptune Replicas were serving will be briefly interrupted.

Q: How far behind the primary will my replicas be?

Since Amazon Neptune Replicas share the same data volume as the primary instance, there is virtually no replication lag. We typically observe lag times in the 10s of milliseconds.

Security

Q: Can I use Amazon Neptune in Amazon Virtual Private Cloud (Amazon VPC)?

Yes, all Amazon Neptune DB Instances must be created in a VPC. With Amazon VPC, you can define a virtual network topology that closely resembles a traditional network that you might operate in your own datacenter. This gives you complete control over who can access your Amazon Neptune databases.

Q: Does Amazon Neptune support encrypting my data in transit and at rest?

Amazon Neptune supports HTTPS encrypted client connections and also allows you to encrypt your databases using keys you manage through AWS Key Management Service (KMS). On a database instance running with Amazon Neptune encryption, data stored at rest in the underlying storage is encrypted, as are its automated backups, snapshots, and replicas in the same cluster. Encryption and decryption are handled seamlessly. For more information about the use of KMS with Amazon Neptune, see the Amazon Neptune User Guide.

Q: Can I encrypt an existing unencrypted database?

Currently, encrypting an existing unencrypted Neptune instance is not supported. To use Amazon Neptune encryption for an existing unencrypted database, create a new DB Instance with encryption enabled and migrate your data into it.

Q: How do I access my Amazon Neptune database?

Access to Amazon Neptune databases must be done through the HTTP port entered on database creation within your VPC. This is done to provide an additional layer of security for your data. Step by step instructions on how to connect to your Amazon Neptune database is provided in the Amazon Neptune User Guide.



[Learn more about pricing](#)

Review pricing information for Amazon Neptune.

Learn more



Sign up for a free account

Instantly get access to the AWS Free Tier.

Sign up (<https://portal.aws.amazon.com/gp/aws/developer/registration/index.html>)



Start building on the console

Get started building with Amazon Neptune on the AWS Management Console.

Sign in (<https://console.aws.amazon.com/neptune/>)

Register for re:Invent Bootcamps

Learn from AWS experts at exam prep bootcamps - space is limited!



(https://reinvent.awsevents.com/learn/bootcamps/?sc_icampaign=aware_reinvent_bootcamps2019_aws&sc_ichannel=ha&sc_icontent=awssm-2448&sc_iplace=2up&trk=ha_awssm-2448)

Get AWS Certified

AWS Certifications can help you advance your career and boost your earning power



(https://pages.awscloud.com/tc_get-aws-certified.html?sc_icampaign=aware_getcertified_evergreen2019&sc_ichannel=ha&sc_icontent=awssm-2655&sc_iplace=2up&trk=ha_awssm-2655)

AWS re:Invent | December 2 – 6, 2019 | Las Vegas, Nevada

Reserved seating opens October 15th. Register now to save your spot. [View session catalog >>](#)

AWS re:Invent

re:II IVEI IL

(https://reinvent.awsevents.com/learn/?sc_icampaign=Event_reInvent_2019_1up_DG5_VIP&sc_ichannel=ha&sc_icontent=awssm-3019-a&sc_ioutcome=Strategic_Events&sc_iplace=1up&trk=ha_a131L0000058G7nQAE~ha_awssm-3019~ha_awssm-3019-a&trkCampaign=AWS_reInvent_2019)

Page Content

General Performance Pricing Hardware and Scaling Backup and Restore High Availability and Replication Security

aws.amazon.com (<https://aws.amazon.com/neptune/faqs/>)