**AWS Documentation**

|  |
| --- |
|  |
| |  | | --- | | **AWS EC2** <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html> | | **AMIs and Instances**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instances-and-amis.html>  **Instance Types**  <https://aws.amazon.com/ec2/instance-types/>  **Root Device Volume**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html>  **Block Device Mapping**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/block-device-mapping-concepts.html>  **Storage**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Storage.html>  store persistent data in  Amazon S3 or  Amazon EBS volumes  **Identity and Access Management (IAM)**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/security-iam.html>  **Key Pairs and Linux Instances**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html>  **Security Groups for Linux Instances**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html> | | **Spot Instances**  [**https://aws.amazon.com/ec2/spot/**](https://aws.amazon.com/ec2/spot/)  **AWS Batch**  [**https://docs.aws.amazon.com/batch/latest/userguide/allocation-strategies.html**](https://docs.aws.amazon.com/batch/latest/userguide/allocation-strategies.html) |  |  | | --- | | **AWS EC2 Storage** | | **Creating an instance store-backed Linux AMI**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html>  The AMI that you specify when you launch your instance determines the type of root device volume  Only the following instance types support an instance store volume as the root device: C3, D2, G2, I2, M3, and R3.  **Creating an Amazon EBS-backed Linux AMI**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html>  **Root Device volume Storage**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html>  You can choose between AMIs backed by EC2 instance store or AMIs backed by EBS.  We recommend that you use AMIs backed by Amazon EBS, because they launch faster and use persistent storage.  **Creating Amazon EBS snapshots**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-creating-snapshot.html>  **Snapshots: incremental backups of your EBS volumes.**  You can easily create a snapshot from a volume while the instance is running and the volume is in use.  <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSSnapshots.html>  **AWS Open Data Registry (S3 Bucket)**  [**https://registry.opendata.aws/**](https://registry.opendata.aws/)  **Why can’t I connect to my Amazon S3 bucket from my Amazon EC2 instance? 2020-12-07**  <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-instance-access-s3-bucket/> | | **AWS Import/Export Calculator (S3, EBS, Glacier)**  <http://awsimportexport.s3.amazonaws.com/aws-import-export-calculator.html> |      |  | | --- | | **AWS EC2 Shared AMIs**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sharing-amis.html> | | **Public AMI Bookmark**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-bookmarks.html>  Example:  https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard:ami=ami-0abcdef1234567890  **Guidelines for shared Linux AMIs**  <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/building-shared-amis.html>  Use the following guidelines to reduce the attack surface and improve the reliability of the AMIs you create.  Contents  Update the AMI tools before using them  Disable password-based remote logins for root  Disable local root access  Remove SSH host key pairs  Install public key credentials  Disabling sshd DNS checks (optional)  Identify yourself  Protect yourself |  |  | | --- | | **EC2 High-Performance** | | **A Placement Group is a logical grouping of instances within a single Availability Zone.**  <http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html>  If you have a high-performance compute application and you need to minimize network latency between EC2 instances as much as possible. Create a placement group within an Availability Zone and place the EC2 instances within that placement group.  Placement groups are recommended for applications that benefit from low network latency, high network throughput, or both.  To provide the lowest latency, and the highest packet-per-second network performance for your placement group,  choose an instance type that supports enhanced networking.  **Increased Network Bandwidth for EC2 Instances JAN 2018**  <https://aws.amazon.com/blogs/aws/the-floodgates-are-open-increased-network-bandwidth-for-ec2-instances/>  Configure AMIs and EC2 instances to use the Elastic Network Adapter (ENA)  To take advantage of this additional bandwidth, make sure that you are using the latest, ENA-enabled AMIs on current-generation EC2 instances. ENA-enabled AMIs are available for Amazon Linux, Ubuntu 14.04 & 16.04, RHEL 7.4, SLES 12.  The actual bandwidth is dependent on the instance type and size.  **EC2 to S3** **– Traffic to and from Amazon Simple Storage Service (S3)** can now take advantage of up to 25 Gbps of bandwidth.  Previously, traffic of this type had access to 5 Gbps of bandwidth.  This will be of benefit to applications that access large amounts of data in S3 or that make use of S3 for backup and restore.  **EC2 to EC2 (Cluster Placement Group)** **– Traffic to and from EC2 instances within a cluster placement group** can continue to take up to 10 Gbps of lower-latency bandwidth for single-flow traffic, or  25 Gbps of lower-latency bandwidth for multi-flow traffic. |  |  | | --- | | **AWS EC2 Container Service (ECS):**  <https://docs.aws.amazon.com/AmazonECS/> | | **ECS Developer Guids**  <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/>  **Launching an Amazon ECS Container Instance** <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/launch_container_instance.html>  **ECS Container Agent Configuration**  <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-agent-config.html>  **Storing Container Instance Configuration in Amazon S3**  <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-agent-config.html#ecs-config-s3>  **ECS Container Instance IAM Role**  <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/instance_IAM_role.html>  **ECS-optimized AMIs**  <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-optimized_AMI.html> | | **BiocDockerManager**  Designed to work analogous to BiocManager but for docker images.  **install packages in docker/R/bioconductor so that they persist?**  <https://support.bioconductor.org/p/74801/> |  |  | | --- | | **AWS EC2 Container Registry (ECR):** <https://aws.amazon.com/ecr/> | | **Using the Amazon ECR Public Gallery**  <https://docs.aws.amazon.com/AmazonECR/latest/public/public-gallery.html>  **Amazon ECR Public Gallery**  Linux  https://gallery.ecr.aws/?operatingSystems=Linux  1014 results  <https://gallery.ecr.aws/bitnami/spring-cloud-dataflow>  <https://gallery.ecr.aws/bitnami/jenkins> | |

|  |
| --- |
| **Bioconductor Docker** |
| |  | | --- | | **DockerHub Bioconductor**  <https://hub.docker.com/u/bioconductor/>  37 repositories 2020/12 | | **Docker Hub**  <https://hub.docker.com/_/microsoft-bioconductor-bioconductor-docker>  **Docker containers for Bioconductor**  <https://www.bioconductor.org/help/docker/>  Docker packages software into self-contained environments, called containers, that include necessary dependencies to run.  Containers can run on any operating system including Windows and Mac (using modern Linux kernels) via the Docker engine.  **Current Containers**  <https://www.bioconductor.org/help/docker/#current>  For each supported version of Bioconductor, we provide  bioconductor/bioconductor\_docker:RELEASE\_X\_Y  bioconductor/bioconductor\_docker:devel  Bioconductor’s Docker images are stored in Docker Hub; the source Dockerfile(s) are in Github. |   **Docker containers for Bioconductor**  <https://www.bioconductor.org/help/docker/>  **GitHub** **Bioconductor / bioconductor\_docker**  h[ttps://github.com/Bioconductor/bioconductor\_docker](https://github.com/Bioconductor/bioconductor_docker)  **Working with Bioconductor's Docker Containers**  <https://www.youtube.com/watch?v=-Jr8k90JQFI>  Video Jun 18, 2016   |  | | --- | | **Bioconductor Docker Images**  bioconductor/bioconductor\_docker  By bioconductor  Dec 07 2020  <https://hub.docker.com/r/bioconductor/bioconductor_docker>  **bioconductor/bioconductor\_docker**  <https://hub.docker.com/r/bioconductor/bioconductor_docker> <<<<<<<  By bioconductor  Updated 12/04/2020  Bioconductor Docker Images | | **DockerHub** Excellent instructions Bioconductor / bioconductor\_docker  <https://github.com/Bioconductor/bioconductor_docker>  **Using the containers**  <https://github.com/Bioconductor/bioconductor_docker#using-the-containers>  A well organized guide to popular docker commands can be found here. For convenience, below are some commands to get you started.  The following examples use the **bioconductor/bioconductor\_docker:devel image**  **Running the container**  <https://github.com/Bioconductor/bioconductor_docker#running-the-container>  The above commands can be helpful but the real basics of running a Bioconductor Docker involves  pulling the public image and running the container.  **Mounting Additional Volume**  <https://github.com/Bioconductor/bioconductor_docker#mounting-additional-volume>  One such option for docker run is -v to mount an additional volume to the docker image.  This might be useful for say mounting a local R install directory for use on the docker.  The path on the docker image that should be mapped to a local R library directory is **/usr/local/lib/R/host-site-library**  **Modifying the images**  <https://github.com/Bioconductor/bioconductor_docker#modifying-the-images>  There are two ways to modify these images:  1: Making changes in a running container and then committing them using the docker commit command.  docker commit  2: Using a Dockerfile to declare the changes you want to make (recommended way).  The second way is the recommended way. | | **Docker containers for Bioconductor**  <https://www.bioconductor.org/help/docker/#current>  Current Containers  For each supported version of Bioconductor, we provide  bioconductor/bioconductor\_docker:RELEASE\_X\_Y  bioconductor/bioconductor\_docker:devel | | **DockerHub Bioconductor**  <https://hub.docker.com/u/bioconductor/>  37 repositories 2020/12 | | **Rocker Project**  Docker Containers for the R Environment  <https://www.rocker-project.org/>  Bioconductor’s Docker images are stored in Docker Hub;  the source Dockerfile(s) are in Github.  Ensure you have Docker installed and start R inside a container with:  **docker run --rm -ti rocker/r-base**  Or get started with an RStudio® instance:  **docker run -e PASSWORD=yourpassword --rm -p 8787:8787 rocker/rstudio**  **Browser URL localhost:8787**  Log in with user/password rstudio/yourpassword  (Please set your own password; it cannot be rstudio).    **GitHub Rocker**  rocker-org / rocker  <https://github.com/rocker-org/rocker/tree/master/rstudio> | |

|  |
| --- |
| **AWS Data** |
| What type of data transfer is free for Amazon s3?  Pick the right AWS region for your S3 bucket.  The main benefit of having S3 and EC2 in the same region is the performance and lower transfer cost.   Data transfer is free  between EC2 and S3 in the same region.  Downloading file  from another AWS region will cost $0.02/GB.  May 18, 2019 |
| **New AWS public datasets available from the National Cancer Institute, Massachusetts Institute of Technology, Amazon, the National Renewable Energy Laboratory, and others**  Posted On: Jul 15, 2020  <https://aws.amazon.com/about-aws/whats-new/2020/07/new-aws-public-datasets/>  Twenty-three new or updated Amazon Web Services (AWS) public datasets from the National Center for Bioinformatics, Johns Hopkins University, University of Texas at Southwestern, National Oceanic and Atmospheric Administration (NOAA), the National Cancer Institute, National Herbarium of New South Wales, and others are now available in the following categories:  COVID-19 response:   * [COVID-19 Molecular Structure and Therapeutics Hub](https://registry.opendata.aws/molssi-covid19-hub/) from the Molecular Sciences Software Institute * [COVID-19 Genome Sequence Dataset](https://registry.opendata.aws/ncbi-covid-19) from the National Center for Biotechnology Information Life sciences: * [Cloud Genomic Indexes](https://registry.opendata.aws/jhu-indexes/) from Johns Hopkins University and the University of Texas at Southwestern * [Refgenie Genomic Assets](https://registry.opendata.aws/refgenie/) from University of Virginia * [Gabriella Miller Kids First Pediatric Research Program](https://registry.opendata.aws/kids-first/) from the National Cancer Institute * [The Cancer Genome Atlas](https://registry.opendata.aws/tcga/) from the National Cancer Institute * [Basic Local Alignment Sequence Tool (BLAST) Databases](https://registry.opendata.aws/ncbi-blast-databases/) from the National Library of Medicine * [National Herbarium of New South Wales](https://registry.opendata.aws/nsw-herbarium/) from the Royal Botanic Gardens and Domain Trust |
| **Access to Genomic Data for free**  In **S3**: <http://s3.amazonaws.com/1000genomes>  **IGSR: The International Genome Sample Resource**  Open human variation data  <https://www.internationalgenome.org/home>  **1000 Genomes Project FAQ**  If you have any other questions you can’t find the answer to please email [info@1000genomes.org](mailto:info@1000genomes.org)  <https://www.internationalgenome.org/faq> |

|  |
| --- |
| **AWS EMR**  <https://aws.amazon.com/emr/>  <https://aws.amazon.com/elasticmapreduce>  A web service that makes it easy to process large amounts of data efficiently.  Amazon EMR uses Hadoop processing combined with several AWS products to do such tasks as  web indexing, data mining, log file analysis, machine learning, scientific simulation, and data warehousing. |
| **EMR features**  <https://aws.amazon.com/emr/features/> |
| **EMR Notebooks**  <https://pages.awscloud.com/EMR_Notebooks.html> |

|  |
| --- |
| **Bioconductor Parallel, Cluster** |
| |  | | --- | | **Bioconductor/AWSParallel**  <https://rdrr.io/github/Bioconductor/AWSParallel/f/vignettes/AWSParallel-AWSBatchJobsParam-tutorial.Rmd> | |  |  |  | | --- | | **StarCluster**  <http://star.mit.edu/cluster/>  StarCluster is an open source cluster-computing toolkit for Amazon’s Elastic Compute Cloud (EC2) released under the LGPL license.  StarCluster has been designed to automate and simplify the process of building, configuring, and managing clusters of virtual machines on Amazon’s EC2 cloud.  StarCluster allows anyone to easily create a cluster computing environment in the cloud suited for distributed and parallel computing applications and systems.  Scroll down to Quick-Start Screencast | | **Quick-Start**  <http://star.mit.edu/cluster/docs/latest/quickstart.html>  **Installing StarCluster**  <http://star.mit.edu/cluster/docs/latest/installation.html>  **Starcluster on Ubuntu 14.04 - Quick Start - AMI Cookbook Recipe - Ubuntu 14**  <https://www.youtube.com/watch?v=2RBupgpi_ec>  2014 |  |  | | --- | | **Introduction to BiocParallel**  1Martin.Morgan@RoswellPark.org  February 10, 2019  <https://bioconductor.statistik.tu-dortmund.de/packages/3.8/bioc/vignettes/BiocParallel/inst/doc/Introduction_To_BiocParallel.pdf> | |  | |

|  |
| --- |
| **AWS Bioinformatics** |
| |  | | --- | | **Amazon Publications** | | **Building High-Throughput Genomics Batch Workflows on AWS: Introduction (Part 1 of 4)**  <https://aws.amazon.com/blogs/compute/building-high-throughput-genomics-batch-workflows-on-aws-introduction-part-1-of-4/>  30 MAY 2017  **Genomics Data Transfer, Analytics, and MachineLearning using AWS Services**  AWS Whitepaper  <https://docs.aws.amazon.com/whitepapers/latest/genomics-data-transfer-analytics-and-machine-learning/genomics-data-transfer-analytics-and-machine-learning.pdf> | |

|  |
| --- |
| **Glossary** |
| |  | | --- | | **Elastic Compute Cloud (EC2)**  <https://aws.amazon.com/ec2>  A web service for launching and managing Linux/UNIX and Windows Server instances in Amazon's data centers. | | **Amazon Machine Image (AMI)**  An encrypted machine image stored in  Elastic Block Store (EBS) or Simple Storage Service (S3).  AMIs are **like a template of a computer's** **root drive**.  They contain the operating system and can also include;  software and layers of your application, such as database servers, middleware, web servers, and so on.  **Amazon EC2 Instance Types**  <https://aws.amazon.com/ec2/instance-types/>  **instance type**  A specification that defines the memory, CPU, storage capacity, and usage cost for an instance.  Some instance types are designed for standard applications, whereas others are designed for CPU-intensive, memory-intensive applications, and so on.  **Amazon EC2 T2 Instances**  <https://aws.amazon.com/ec2/instance-types/t2/> | |  | | **instance store**  Disk storage that is physically attached to the host computer for an EC2 instance, and therefore has the same lifespan as the instance.  When the instance is terminated, you lose any data in the instance store.  **instance store-backed AMI**  A type of Amazon Machine Image (AMI) whose instances use an instance store volume as the root device.  Compare this with instances launched from EBS-backed AMIs, which use an EBS volume as the root device.  **EBS-backed AMI**  <https://aws.amazon.com/ebs>  A type of Amazon Machine Image (AMI) whose instances use an Amazon EBS volume as their root device.  Compare this with instances launched from instance store-backed AMIs, which use the instance store as the root device.  **Elastic Block Store (EBS)**  <https://aws.amazon.com/ebs>  A service that provides block level storage volumes for use with EC2 instances.    **root device volume**  A volume that contains the image used to boot the instance (also known as a root device).  If you **launched instance from an AMI backed by instance store**,   this is an **instance store volume created from a template stored in S3**.  If you **launched instance from an AMI backed by EBS**, **T2.xlarge Instance** <https://aws.amazon.com/ec2/instance-types/t2/>  this is an **EBS volume created from an EBS snapshot**.    **volume**  A fixed amount of storage on an instance.  You can share volume data between containers and persist the data on the container instance when the containers are no longer running. | | **EC2 Auto Scaling**  <https://aws.amazon.com/ec2/autoscaling>  A web service designed to launch or terminate instances automatically based on user-defined policies, schedules, and health checks.  **Elastic File System (EFS)**  <https://aws.amazon.com/efs/>  A file storage service for EC2 instances.  Amazon EFS is easy to use and provides a simple interface with which you can create and configure file systems.  Amazon EFS storage capacity grows and shrinks automatically as you add and remove files.  **Simple Storage Service (S3)**  <https://aws.amazon.com/s3>  Storage for the internet.  You can use it to store and retrieve any amount of data at any time, from anywhere on the web.  **S3-Backed AMI**  See instance store-backed AMI.  **S3 Glacier**  <https://aws.amazon.com/glacier/>  A secure, durable, and low-cost storage service for data archiving and long-term backup.  You can reliably store large or small amounts of data for significantly less than on-premises solutions.  S3 Glacier is optimized for infrequently accessed data, where a retrieval time of several hours is suitable.  **persistent storage**  A data storage solution where the data remains intact until it is deleted.  Options within AWS include: Amazon S3, Amazon RDS, Amazon DynamoDB, and other services.  **snapshot**  Amazon Elastic Block Store (Amazon EBS): A backup of your volumes that is stored in Amazon S3.  You can use these snapshots as the starting point for new Amazon EBS volumes or to protect your data for long-term durability. | | **Docker image**  A layered file system template that is the basis of a Docker container. Docker images can comprise specific operating systems or applications.  **Elastic Container Service (ECS)**  <https://aws.amazon.com/ecs>  A highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers on a cluster of EC2 instances.  **Elastic Container Registry (ECR)**  <https://aws.amazon.com/ecr>  A fully managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images.  Amazon ECR is integrated with Amazon Elastic Container Service (Amazon ECS) and AWS Identity and Access Management (IAM).  **Elastic Kubernetes Service (EKS)**  <https://aws.amazon.com/eks/>  A managed service that simplifies running Kubernetes on AWS without your needing to stand up or maintain your own Kubernetes control plane. | |  | | **Elastic IP address**  A fixed (static) IP address that you have allocated in Amazon EC2 or Amazon VPC and then attached to an instance.  Elastic IP addresses are associated with your account, not a specific instance.  They are elastic because you can easily allocate, attach, detach, and free them as your needs change.  Unlike traditional static IP addresses, Elastic IP addresses allow you to mask instance or Availability Zone failures by  rapidly remapping your public IP addresses to another instance. | | **Amazon Resource Name (ARN)**  A standardized way to refer to an AWS resource.  Example: arn:aws:iam::123456789012:user/division\_abc/subdivision\_xyz/Bob | | **Virtual Private Cloud (VPC)**  <https://aws.amazon.com/vpc>  A web service for provisioning a logically isolated section of the AWS Cloud virtual network that you define.  You control your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.  **private IP address**  A private numerical address (for example, 192.0.2.44) that networked devices use to communicate with one another using the Internet Protocol (IP).  All EC2 instancess are assigned two IP addresses at launch,  which are directly mapped to each other through network address translation (NAT):  a private address (following RFC 1918) and  a public address.  Exception: Instances launched in Amazon VPC are assigned only a private IP address.  **private subnet**  A VPC subnet whose instances cannot be reached from the internet.  **key pair**  A set of security credentials that you use to prove your identity electronically.  A key pair consists of a private key and a public key.  **secret access key**  A key that is used in conjunction with the access key ID to cryptographically sign programmatic AWS requests. Signing a request identifies the sender and prevents the request from being altered. You can generate secret access keys for your AWS account, individual IAM users, and temporary sessions.  **security group**  A named set of allowed inbound network connections for an instance. (Security groups in Amazon VPC also include support for outbound connections.) Each security group consists of a list of protocols, ports, and IP address ranges. A security group can apply to multiple instances, and multiple groups can regulate a single instance.  **session**  The period during which the temporary security credentials provided by AWS Security Token Service (AWS STS) allow access to your AWS account. | |  | | **Amazon EMR**  <https://aws.amazon.com/elasticmapreduce>  A web service that makes it easy to process large amounts of data efficiently.  Amazon EMR uses Hadoop processing combined with several AWS products to do such tasks as  web indexing, data mining, log file analysis, machine learning, scientific simulation, and data warehousing. | |

|  |
| --- |
| **Amazon EC2**  **Amazon Elastic Compute Cloud**  Secure and resizable compute capacity to support virtually any workload  <https://aws.amazon.com/ec2/>  Is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2’s simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment.  Amazon EC2 offers the broadest and deepest compute platform with choice of processor, storage, networking, operating system, and purchase model. We offer the fastest processors in the cloud and we are the only cloud with 400 Gbps ethernet networking. |
| AWS Free Tier  <https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=tier%2312monthsfree&awsf.Free%20Tier%20Categories=categories%23compute%7Ccategories%23analytics>  Gain free, hands-on experience with the AWS platform, products, and services  Explore more than 85 products and start building on AWS using the free tier. Three different types of free offers are available depending on the product used. See below for details on each product. |
| AWS Pricing Calculator  <https://calculator.aws/#/addService> |
| Amazon Elastic Block Store (EBS)  <https://aws.amazon.com/ebs/>  allows you to create persistent block storage volumes and attach them to Amazon EC2 instances.  Amazon Elastic Block Store (EBS) is an easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale. A broad range of workloads, such as relational  and non-relational databases, enterprise applications, containerized applications, big data analytics engines, file systems, and media workflows are widely deployed on Amazon EBS.  You can choose from six different volume types to balance optimal price and performance. You can achieve single digit-millisecond latency for high performance database workloads such as SAP HANA or gigabyte per second throughput for large, sequential workloads such  as Hadoop. You can change volume types, tune performance, or increase volume size without disrupting your critical applications, so you have cost-effective storage when you need it.  Designed for mission-critical systems, EBS volumes are replicated within an Availability Zone (AZ) and can easily scale to petabytes of data. Also, you can use EBS Snapshots with automated lifecycle policies to back up your volumes in Amazon S3, while ensuring  geographic protection of your data and business continuity. |
| **AWS Associate Solutions Architect Notes**<https://github.com/Apjo/AWSAssociateSolutionsArchitectNotes/blob/master/README.md>Notes prepared while doing the online course on AWS Associate Solutions Architect.These notes are a mixture of notes from a course taught by Ryan Kroonenburg, and one taught by EissaEC2 (Elastic Compute Cloud)- Provides resizable compute capacity in the cloud.- You have root access to the EC2 instances, be able to restart, terminate, reboot.- You need to have a key & key pair to access the instance- Reduces the time required to obtain and boot new server instances to minutes,allowing you to quickly scale capacity, both up and down, as computing requirements change.- 2 types of block store devices namely    Elastic Block store which are persistent and Network attached virtual drives,these are not directly connected to the host where the instance is but are attached to the network,where as Instance-store are not persistent(ephemeral), basically a virtual hard drive on the host allocated to this EC2 instanceandEBS-backed EC2 instance has a EBS root volume, and Instance-store backed EC2 instance has instance-store root volume7: **Analytics:**ElasticMapReduce(EMR):A web service that makes it easy to quickly and cost-effectively process vast amounts of data.Uses Hadoop, an open source framework to distribute your data and process across a resizeable cluster of Amazon EC2 instances.It can also run other distributed framework such as Spark and Presto.EMR is used in a variety of applications including log analysis, web-indexing, data warehousing, machine learning, financial analysis, scientific simulation, and Bioinformatics, customers launch millions of Amazon’s EMR clusters each yearAllows you root access(i.e. login via SSH)Cloud Search / Elastic Search:Search engine for your website or your application,Cloud search is fully managed service provided by AWS, Elastic search uses open source frameworkData Pipeline: Allows to move data from one place to anotherQuick Sight: used for creating visualizations, dashboards for BI/Analytics |
| **Introducing new Amazon EBS general purpose volumes, gp3  Posted On: Dec 1, 2020**  Today AWS announced the availability of gp3, the next-generation general purpose SSD volumes for Amazon Elastic Block Store (Amazon EBS) that enable customers to  provision performance independent of storage capacity and provides up to 20% lower price-point per GB than existing gp2 volumes.  With gp3 volumes, customers can scale IOPS (input/output operations per second) and throughput without needing to provision additional block storage capacity, and pay only for the resources they need.  General purpose SSD volumes make it easy and cost effective for customers to meet the IOPS and throughput requirements for transaction-intensive workloads, such as virtual desktops, test and development environments, low-latency interactive applications, and boot volumes. With existing general-purpose SSD (gp2) volumes, performance is tied to storage capacity, enabling customers to get higher IOPS and throughput for their applications by provisioning a larger storage volume size. But customers want to scale performance and throughput without paying for storage that they don’t need.  Next generation gp3 volumes offer the ability to independently provision IOPS and throughput, separate from storage capacity. This enables customers to scale performance for transaction-intensive workloads without needing to provision more capacity, so they only pay for the resources they need. The new gp3 volumes also deliver a baseline performance of 3,000 IOPS and 125MB/s at any volume size. For use cases, where your application needs more performance than the baseline, customers can scale up to 16,000 IOPS and 1,000 MB/s for an additional fee. This makes the new gp3 volumes ideal for a wide variety of applications that require high performance at low cost, including MySQL, Cassandra, virtual desktops, and Hadoop analytics clusters.  Customers can easily migrate gp2 volumes to gp3 volumes using Elastic Volumes, which is an existing feature of Amazon EBS.  Elastic Volumes allow customers to modify the volume type, IOPS, or throughput of their existing EBS volumes without interrupting their Amazon EC2 instances. gp3 volumes are available in all AWS commercial and gov cloud regions. For more information, please see the gp3 announcement on the AWS News blog, and documentation. |
| **Volume Gateway**  **Hybrid cloud block storage with local caching**  <https://aws.amazon.com/storagegateway/volume/>  Volume Gateway presents cloud-backed iSCSI block storage volumes to your on-premises applications. Volume Gateway stores and manages on-premises data in Amazon S3 on your behalf and operates in either cache mode or stored mode. In the cached Volume Gateway mode, your primary data is stored in Amazon S3, while retaining your frequently accessed data locally in the cache for low latency access. In the stored Volume Gateway mode, your primary data is stored locally and your entire dataset is available for low latency access on  premises while also asynchronously getting backed up to Amazon S3. In either mode, you can take point-in-time copies of your volumes using AWS Backup, which are stored in AWS as Amazon EBS snapshots. Using Amazon EBS Snapshots enables you to make space-  efficient versioned copies of your volumes for data protection, recovery, migration, and various other copy data needs.  **Amazon Simple Storage Service (S3)**  Object storage built to store and retrieve any amount of data from anywhere  <https://aws.amazon.com/s3/>  (Amazon S3) is storage for the internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web.  Amazon ElastiCache  offers fully managed Redis and Memcached. Seamlessly deploy, run, and scale popular open source compatible in-memory data stores. Build data-intensive apps or improve the performance of your existing apps by retrieving data from high throughput and low latency in-memory data stores.  AWS Storage Gateway  <https://aws.amazon.com/storagegateway/>  AWS Storage Gateway is a hybrid storage service that enables your on-premises applications to seamlessly use AWS cloud storage. You can use the service for backup and archiving, disaster recovery, cloud data processing, storage tiering, and migration.  Amazon Elastic Container Registry  (ECR) is a fully-managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images. Amazon ECR eliminates the need to operate your own container repositories or worry about scaling the underlying infrastructure.  With Amazon ECR, there are no upfront fees or commitments. You pay only for the amount of data you store in your repositories and data transferred to the Internet.  Amazon RDS for PostgreSQL  Amazon RDS makes it easy to set up, operate, and scale PostgreSQL deployments in the cloud. With Amazon RDS, you can deploy scalable PostgreSQL deployments in minutes with cost-efficient and resizable hardware capacity. |

|  |
| --- |
| **Using Amazon EC2 to run large data analyses cheaply Spot instances on Amazon Elastic Compute Cloud (EC2)**  Jul 5, 2018  <https://gex.netlify.app/post/using-amazon-ec2-to-run-large-data-analysis-cheaply/> |
| Spot instances on Amazon Elastic Compute Cloud (EC2) allow researchers to do high performance computing at very low cost.  For example, a 64-core workstation with 256 GB of memory can be rented at about $0.7 per hour  Main topics covered this tutorial  Set up Key Pairs and Security Groups to enable SHH login,  Request spot instance (virtual machine),  SHH access to the instance via Putty and Filezilla,  Install Docker software,  Build a Docker image based on the Bioconductor Docker definition files,  Start R and compute from within the container,  Create a “volume” (a virtual hard drive) and attach it to running instances,  Take a snapshot of a volume, copy it across regions, and use it to create a volume,  Google Compute Engine set-up. |
| Start the Spot Advisor and enter your request.  Change the amount required to 1. Otherwise, 20 instances will be requested.  A 64-core instance with 256GB memory costs $0.67 per hour or about $16 per day.  Note that Amazon has web servers all across the world. You can switch regions from the top right of the screen.  Prices and availability of resources vary greatly across the region. |