

# AWS Genomics Guide

*August 2017*



## Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

# Contents

Introduction	1
AWS Value Proposition for Genomics	1
Compliance and Security	2
Classifying data for compliance requirements	2
Deploy AWS environment to meet your needs	4
Access Management	5
Genomics on AWS	6
Analysis Stages in Genomics	6
Analysis of Genomic Data on AWS	7
Processing	15
Sharing	27
Public Datasets	28
Conclusion	29
Document Revisions	30

# Abstract

This whitepaper focuses on common strategies and best practices used successfully by Amazon Web Services (AWS) customers for analyzing genomics sequencing data and associated medical datasets. For more information regarding specific customer use cases, please refer to our customer [Healthcare and Life Sciences Web Portal](#). Our intention is to provide you with helpful guidance that you can use to facilitate your genomics initiatives using AWS services and features. However, we caution you not to rely on this whitepaper as legal advice for your specific use of AWS. We strongly encourage you to obtain appropriate compliance advice about your specific data privacy and security requirements, as well as applicable laws relevant to your human research projects and datasets.

# Introduction

## Welcome to the AWS Genomics User Guide!

Whether you are just getting started or have already been analyzing genomics data using the AWS Cloud, we hope that the AWS Genomics User Guide will provide you with some of the 'know-how' information that you need in order to use our services and features in the ways that will make the most sense for your data analytical objectives. Let us solve the mysteries of how to leverage the right resources for your genomics data processing and analytics jobs so that you can solve the mysteries surrounding health, disease, and evolution.

## AWS Value Proposition for Genomics

- AWS provides multiple advantages for building scalable, cost effective and secure genomic analysis pipelines. Here are some key advantages of using AWS for analysis in general that we will be providing a deeper dive discussion of in the following sections of this whitepaper:
- Genomics secondary-stage analysis pipelines are typically executed in “cohort” or “batch” workloads. As a result, infrastructure is only required for the time needed to execute the compute job. AWS provides elasticity to quick scale up or down and hence saves on infrastructure costs.
- Storing Genomics and Medical (e.g. imaging) data at different stages requires enormous storage in a cost-effective manner. Amazon Simple Storage Service (Amazon S3), Amazon Glacier and Amazon Elastics Block Store (Amazon EBS) provide the necessary solutions to securely store, manage and scale genomic file storage. Moreover, the storage services can interface with various compute services from AWS to process these files.
- AWS provides a wide choice of compute services that can be used to process diverse datasets in analysis pipelines. These range from managed services to virtual servers that can be combined with flexible purchasing options consisting of on demand, reserved and spot.
- Genomic sequencers that generate raw data files are located in labs on premises and AWS provides solutions to make it easy for customers to transfer these files to AWS reliably and securely.

- As of 07/31/2017, AWS has 16 regions, 43 availability zones and 77 edge locations across the globe. This number is continuously growing. Using this elaborate network of AWS points of presences, customers can build a secure platform to collaborate on research findings as a result of analyzing genomic and associated medical data sets.
- The AWS Partner Network has a vast ecosystem of independent software vendors (ISVs) and systems integrators (SIs) with domain expertise and products that are applicable for genomics workloads.
- The AWS Marketplace also includes a Healthcare & Life Sciences Industry vertical category that offers a broad range of solutions from 3rd party providers. Solutions include technical Research & Development focused applications, as well as solutions for managing Healthcare and Life Sciences related organizational operations.

## Compliance and Security

Security is job number one at AWS and we recommend prior to working with potentially sensitive data on AWS that you take the time to understand the security and compliance requirements surrounding it.

A typical workflow for addressing compliance needs is as follows:

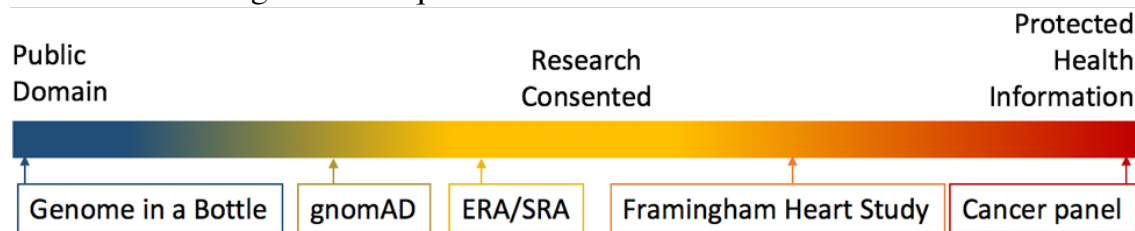
1. Classify data to determine necessary access controls and security requirements
2. Align AWS architectures and standard operating procedures to a compliance framework
3. Deploy AWS environment and controls that meet compliance requirements
4. Deploy data and applications on top of the AWS environment

## Classifying data for compliance requirements

AWS operates under a shared security responsibility model, where AWS is responsible for the security of the underlying cloud infrastructure and you are responsible for securing workloads and data you deploy in AWS. AWS does not

access or use customer content for any purpose other than as legally required and to provide the AWS services selected by each customer, to that customer and its end users. AWS never uses customer content or derives information from it for other purposes such as marketing or advertising.

The implication of the above is that is that you, as the data owner, will need to classify data to fit within the spectrum encompassing public domain through to Protected Health Information (PHI). Figure 1 shows a practical example of data classification for genomic sequence data.



**The spectrum of data classification for security and compliance.**

Genome-in-a-Bottle data are in the public domain; gnomAD, ERA, and SRA release some data within the public domain, but restrict access to individual genomes; all Framingham data restricted access for research use; finally, a cancer gene panel that is produced in the service of making treatment decisions would typically fall under regulatory requirements for Protected Health Information (PHI).

At the most basic level, AWS recommends following the guidelines within the AWS Security Best Practices documentation. When working with sensitive data, AWS recommends following “security by design” principles such as encrypting data in transit and at rest, securing network accessible resources, and robust logging of operations on data and compute resources. Aligning and documenting your operating procedures for management of data and compute resources to a recognized security and compliance framework, such as NIST 800-171, will provide the necessary controls for protecting the data. Doing so will also allow you to quickly onboard other sensitive data, since you are able to leverage the same templated infrastructure and operating procedures. More information on security by design is available in the [Security by Design principles](#) whitepaper.

For healthcare data that are considered PHI, all storage and analysis services will likely fall under the geography’s regulation where the data resides. For the United States, that would be the Health Information Portability and Accountability Act (HIPAA). If you classify some portions of your data and genomics as PHI, then HIPAA regulations will need to be met. There is no

HIPAA certification for a cloud provider such as AWS. In order to meet the HIPAA requirements applicable to our operating model, AWS aligns our HIPAA risk management program with FedRAMP and NIST 800-53, a higher security standard that maps to the HIPAA security rule. NIST supports this alignment and has issued SP 800-66, "An Introductory Resource Guide for Implementing the HIPAA Security Rule," which documents how NIST 800-53 aligns to the HIPAA Security rule. Following our guidance above to align your compliance and security practices to a compliance framework will go a long way towards deploying HIPAA Eligible applications stacks.

Additionally, AWS would be considered your "business associate" and U.S. law require that a contractual agreement be countersigned by a covered entity and AWS. This agreement is referred to as the Business Associate Agreement (BAA). The agreement outlines the AWS services and features that are HIPAA-eligible, and any constraints on these services that are necessary to meet compliance regulations. For example, Amazon S3 is a HIPAA-eligible service, but the BAA stipulates that any that you have determined is PHI must only be encrypted in transit and at rest, and that S3 event logs must be captured for a determined period to allow for forensic analysis in the event of a data breach. For more detailed information regarding HIPAA compliance and the AWS BAA, please refer to our [website](#).

In many cases, customers are also able to use non-HIPAA eligible services and features in their architecture by first implementing a process to de-identify any sensitive data prior to processing. Doing so would allow the de-identified data to be processed through an analytical resource that are not currently listed as part of the BAA (e.g. Amazon EFS, Amazon IoT). Covered services (e.g. Amazon RDS, Amazon DynamoDB) would maintain the PHI metadata, perhaps by establishing an arbitrary key-value mapping protocol, so that the de-identified results can later be re-associated with PHI. De-identification should be done in accordance with the BAA and as outlined by the [U.S. Department of Health & Human Services guidelines for methods of De-Identification](#).

## Deploy AWS environment to meet your needs

In order to meet the identified security controls for data and resources, we recommend that you align your systems and procedures to a security and compliance framework, such as NIST SP 800-171. Adopting and documenting a standard set of controls for the AWS resources, applications, and users to form a set of standard operating procedures is relatively straight forward, but does require some attention. To help with development and documentation of compliant system, AWS has developed a set of Quick Start enterprise accelerator



packages. These AWS Quick Starts are built by AWS solutions architects and partners to help you deploy secure solutions on AWS, based on AWS best practices for security and high availability. These reference deployments implement key technologies automatically on the AWS Cloud, often with a single click and in less than an hour. You can build your test or production environment in a few simple steps, and start using it immediately.

The security and compliance focused Quick Starts include a Security Control Matrix, which is an Excel Spreadsheet that maps the architecture decisions, components, and configuration in the Quick Start to security requirements within a given compliance framework. Utilizing the NIST SP 800-171 AWS Quick Start template and Security Control Matrix will allow you to develop a program to meet regulatory requirements for PHI.

More information on [NIST Compliance on AWS - Enterprise Accelerator](#)

### **More information**

The following represent resources for more specific information for working with:

- Research consented data - [Architecting for Genomics Security and Compliance on AWS](#)
- [Protected Health Information](#)
- [AWS Quick Starts for meeting Security and Compliance needs](#)

## **Access Management**

### **Cross-Account Identity and Access Management**

Research has true value when communities of interest are able to collaborate seamlessly. AWS gives researchers the capability to not only share tools and data, but also entire cloud environments based on prescriptive policies for security and access. This means that you can allow a researcher from a different organization access to your AWS environment while maintaining control of what data and systems they can see, and what they can do within the AWS account. This cross-account access is enable by utilizing [Amazon Identity and Access Management \(IAM\)](#). With IAM cross-account roles you can establish trust relationships between your AWS account and other AWS accounts. The trusting account who owns the resources that need to be shared sets up a IAM Role that other AWS accounts can assume. The IAM Role determines which resources,

services, and data another account can utilize, view, or modify. Once you allow an AWS account to assume the IAM Role, users within the other account will be able to obtain temporary security credentials that enable access to the specified AWS resources in your account. Once you revoke the right of an account to assume the IAM Role, they lose access to your data and other resources.

From a genomics research and collaboration standpoint using cross-account IAM roles means that you can give fined-grained control over specific resources, such as genomes and variants stored in Amazon S3, to other AWS users. A common use-case for cross-account IAM access would be to give another user access to a portion of your data that is relevant to the collaboration so that they can use it in place without needing to make redundant copies of the data within the other account.

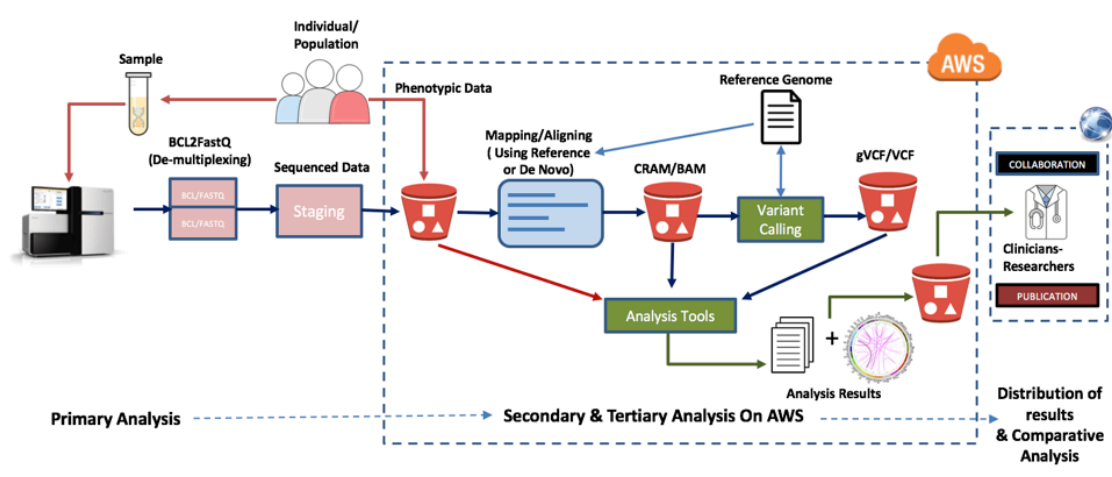
For more information on cross-account roles and how to use them, please refer to [Providing Access to an IAM User in Another AWS Account That You Own](#)

## Genomics on AWS

### Analysis Stages in Genomics

#### Primary and Secondary Analysis

Genomic sequencing is comprised of multiple steps that involve taking samples from individuals and running them through sequencers and then processing the resulting raw files to derive insights. Whole genome sequencing for a single sample can generate between 500 MB to a TB of raw genomic data. Typically, primary analysis is done “on-machine” (by software packages installed on the sequencers) to assess the quality metrics of the sequencing run per sample. These Quality Control (QC) metrics are often mined and associated with downstream analytics. The raw sequencing data is transferred and processed to derive useful information that includes multiple iterations of reading and writing data as it progresses through various stages of processing. This process is known as secondary analysis. The various stages of data processing are part of a secondary analysis pipeline that accepts raw data at one end and churns out processed data on the other.



## Tertiary Analysis

Tertiary analysis refers to a wide range of investigations utilizing diverse data sources that include outputs from primary or secondary analysis stages. Data sources need to be aggregated and analyzed in meaningful ways in order to glean informative insights from the comparative studies being considered. This stage of analysis can involve ETL processing, computationally intensive joining of data points from different sequencing platforms and/or with associated clinical metadata, as is the case in many genotype-phenotype association studies.

## Analysis of Genomic Data on AWS

Genomics is an area of scientific research concerning the sequencing and analysis of the genetic content of an entire organism; this primarily includes DNA, RNA (functional genomics), Epigenetics, Phylogenetic (comparative genomics), as well as the applications of Genomics to advances in technologies (i.e. association studies), regulatory applications, and Human Health (Personalized Medicine and Population Studies). AWS helps researchers take full advantage of rapidly emerging sequencing technologies by creating scalable, high-performance and cost-effective computing and storage solutions that enable genomics, health and big-data workflows. The main steps of any largescale data analytical project involve the following:

### Acquisition

AWS customers within the genomics domain often have vast amounts of raw genomic data which needs to be brought into the AWS cloud for subsequent analysis. Broadly speaking, data acquisition for these customers often relies on a combination of two approaches; either streaming data into the cloud directly from sequencers or batch movement of the data. The following section reviews

each of these approaches in detail along with the AWS cloud components and architectures used to implement them.

## Streaming Approach

Genomic sequencers vary somewhat in the way data is output depending on make and model. More recently, cloud enabled sequencers contain integrated software clients that utilize the S3 API or similar to move digitized genomic data onto the cloud efficiently and reliably. S3 is a highly durable, secure web-based object store that can store and retrieve any amount of data from anywhere on the Internet. The integrated software clients often utilize multi-part upload capability to transfer multiple channels ('parts') of data from the same files concurrently and asynchronously, thus enabling large amounts of data to be transferred to AWS in shorter intervals.

Although integration of cloud enabled software clients directly onto sequencing instrumentation is an efficient method of enabling data transfer, ultimate rates of transfer will depend on the capabilities of the supporting network infrastructure. To help ease some frequently encountered connectivity challenges, genomic data producers will often take advantage of hybrid network architectures that serve to connect their on premise network directly with the AWS cloud. There are several services offered by AWS to support hybrid network integration including VPN services and direct connection (AWS Direct Connect) to an AWS Region. Using the default configuration, transfer to S3 uses a secure protocol (HTTPS) and takes place over the public Internet. When connecting using a hybrid network architecture this transfer can be completed over a non-public connection to the AWS cloud. S3 Transfer Acceleration can also be used to enhance the transfer of data into the AWS cloud. This service utilizes Amazon edge locations to complete transfers using the Amazon network backbone.

Services offered by AWS such as EC2, RDS, EMR and Redshift are hosted within a Virtual Private Cloud and are used as infrastructure to support the processing and analysis of genomic data. A VPC is an MPLS (Multi-Protocol Label Switching) network that offers a logically isolated IP address space partitioned from any other network. Connectivity into a VPC from the outside is managed through the use of hardware or software based endpoints including VPNs, that can be managed from AWS or using a VPN provider.

## Batch Approach

Most often, sequencing instruments are configured to store their output using on premise storage from a locally available SAN or NAS and organize their results as files based on sample or run identifier and the date/time acquired. Frequently, applications are configured to watch directories for this output so that they can transfer the complete results as a batch onto S3. There are several other options for managing output as a batch including cloud based file caching and archiving solutions, file synchronization services and shared file clustering solutions. Amazon provides the AWS Storage Gateway, which is deployed on-premises to provide a file interface to S3 and allows for easy transfer between local storage solutions and the AWS cloud. There are many other partners and AWS marketplace providers that specialize in S3 integrated storage solutions. For more insight into which solution is most appropriate visit the AWS Storage Partner Solutions portal.

## Storage

Genomic sequencing results in the production of significant quantities of data which is then reduced as it proceeds through subsequent analysis stages. For example, a whole genome sequenced with an Illumina sequencer to a depth of 30X will result in a fastq output of approximately 200 GB. As it moves into the alignment stage and aligned to a reference genome, it is reduced to a set of about 80 GB of BAM files. Finally, the variant calling operation, which identifies the approximately 0.1% of the genome that differs between individuals, is stored in VCF format and about 125 MB in size. It should also be noted that the variant call information, stored in VCF format, typically contains the most valuable data utilized for analysis downstream and is often also shared across or stored with a knowledge base.

Genomic data sets can expand to very large volumes during population sequencing efforts. For example, the [1000 Genome data](#) hosted on AWS contains ~80 million variants from over 2500 individuals and is more than 200 TB in size. For a production scale population sequencing effort that retains all data, both raw and the output of each analysis stage, this can quickly become a pressing concern as far as cost and storage management are concerned. It is also worth noting that the platform for data storage needs to be highly secure and reliable as it retains all output of the sequencing effort.

Amazon S3 has 11 9's of durability and 4 9's of availability and can serve as the ideal storage platform for a sequencing pipeline. Data can be placed into S3

directly from sequencing instrumentation or from existing on premises storage. S3 interfaces seamlessly with the AWS compute services and as a result, allows for the decoupling of storage and compute in the overall architecture. S3 also provides [VPC endpoints](#) so that users can access objects in S3 from a private subnet within a VPC and without having to traverse the Internet. S3 also integrates with the Amazon Key Management Service (KMS) and allows for server side or client side encryption making it highly secure.

Please refer to the [Amazon S3 developer guide](#) for more details about the various features S3 has to offer.

The pattern of access to genomic data sets can vary considerably during the course of pipeline execution. Some files can be accessed repeatedly while others are accessed only once. S3 has several options for storage class and it is recommended that customers move files with genomic data to an appropriate storage class in S3 to help save on storage costs.

### Optimizing Storage on S3 for Genomic Data

As mentioned in the previous section, Customers have the option to move data to a more cost-effective storage class on S3 based on the access patterns. S3 provides Standard and Infrequent storage classes that customers should make use of based on the access requirements. Data that is never needed for computation or analysis can be moved to Amazon Glacier for long term archival. You can utilize lifecycle policies to automatically transfer data files to an alternate storage class after a fixed interval of time. For more details on storage classes and their characteristics, please look at the [Amazon S3 documentation](#). For managing data on S3, customer can make use of storage management features of S3. These new features are designed for providing more control to customers providing them with out of the box tools. These tools are easy to use and are fully managed by AWS. Moreover, the new features are accessible via the AWS console, API or CLI so you can integrate these features into your existing applications. Customers can enable these features on a bucket or a prefix and get access patterns, data transfer volumes, storage volumes and object count without writing any code. This information can then be used to do things like:

- Tag objects that belong to a certain sequencer run and use the tag as a filter in an IAM policy for controlling access to those objects.

- Transition infrequently accessed genomic files to S3 IA and files that are never accessed to Glacier.
- Keep a track of your data volumes on S3 and its access patterns over a period of time.
- Get a list of objects in a bucket in CSV format and use it in programming constructs for further analysis.

By utilizing S3 storage management features, customers can just concentrate on taking actions based on information instead of worrying about how to retrieve that information in the first place.

Please look at the following link for more details on these features:

<https://aws.amazon.com/about-aws/whats-new/2016/11/revolutionizing-s3-storage-management-with-4-new-features/>










### ***Example of genomic data tiering with Amazon S3***

There are various file formats that one may encounter during sequencing. These include indexed formats like BAM, BigBed and 2bit and non-indexed ones like SAM, FASTA and PSL. The compute intensive algorithms for genomic data processing reads raw sequencing files from S3, process it, and then write it back into S3 resulting in millions of variant call files (VCF) that need to be stored, analyzed and distributed to consumers in a secure and scalable manner. Moreover, there is a need for better management of these files so customers can categorize them into tiered storage for managing spiraling storage costs.

Let's take an example of a whole genome sequencing run on an Illumina sequencer to demonstrate storage tiering in S3. In this workflow, you generate some or all of the following file formats:

- **Base Call (BCL):** Raw files generated from the sequencers containing base calls per cycle.
- **FASTQ:** These file stores biological sequence and quality score information.
- **BAM:** binary file that contains sequence alignment data.
- **Variant Call Files (VCF):** Stores gene sequence variations in text format.



	S3 Standard	S3 Infrequent Access (IA)	Glacier	Purge
BCL				
FASTQ				
BAM				
VCF				

The following is an example of how we can utilize different storage tiers in S3 for efficiently storing genomic files. The raw BCL files are read once and are rarely touched after that. So, they can be archived to Glacier in case you need them for something later. The FASTQ may be accessed for some time but they can then be purged because you can regenerate them if you need to. BAM files are heavily accessed but the access patterns are known to decrease once after the variant calling operation. Hence, you can move them to S3 Infrequent access after a certain period of time. Lastly, VCF files are small in size but highest in value as it can be used as input for many analysis processes downstream. Hence, you can retain that in S3 standard.

As noted, the above is just an example on who you may think about using tiered storage for your genomic files. You should choose the transitions based on your access patterns.

## Reference data

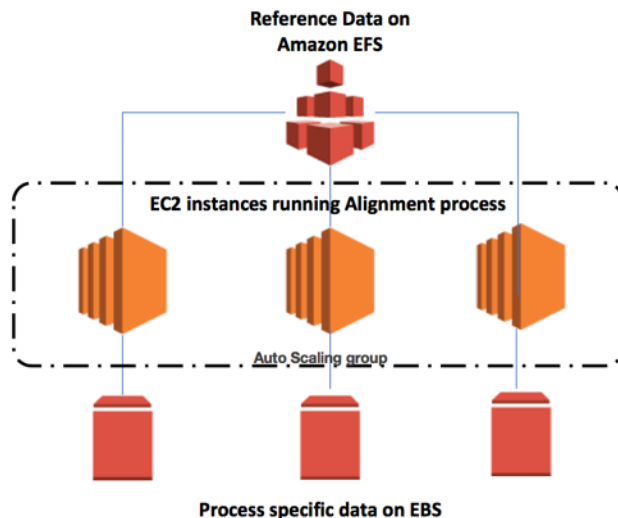
The analysis workflow for a single genome usually consists of an alignment operation that involves comparing it against a reference genome to find genetic variations. The dataset containing information about this reference genome is also known as reference data and can range from 1-5 GB in size. Since the alignment operation is a batch workflow that can be scaled horizontally, the same reference genome data has to be mounted on multiple EC2 instances



performing the alignment and variant calling operations. There are two options that customers can utilize for working with reference data:

### Option 1:

Amazon Elastic File System (EFS) provides scalable network files storage (NFS) for EC2 instances. [Amazon EFS](#)



Multiple EC2 instances can access an EFS file system at the same time, allowing EFS to provide a common data source for workloads and applications running on more than one Amazon EC2 instance. The reference genome data can be stored on an EFS volume that can be mounted on multiple EC2 instances performing the compute operations. Data that is specific to a processing instance can be mounted to EC2 using Amazon Elastic Block Store (EBS) volume or can utilize the ephemeral storage (instance store) of the EC2 instance.

For more details on storage options on AWS, please look at the following [whitepaper](#).

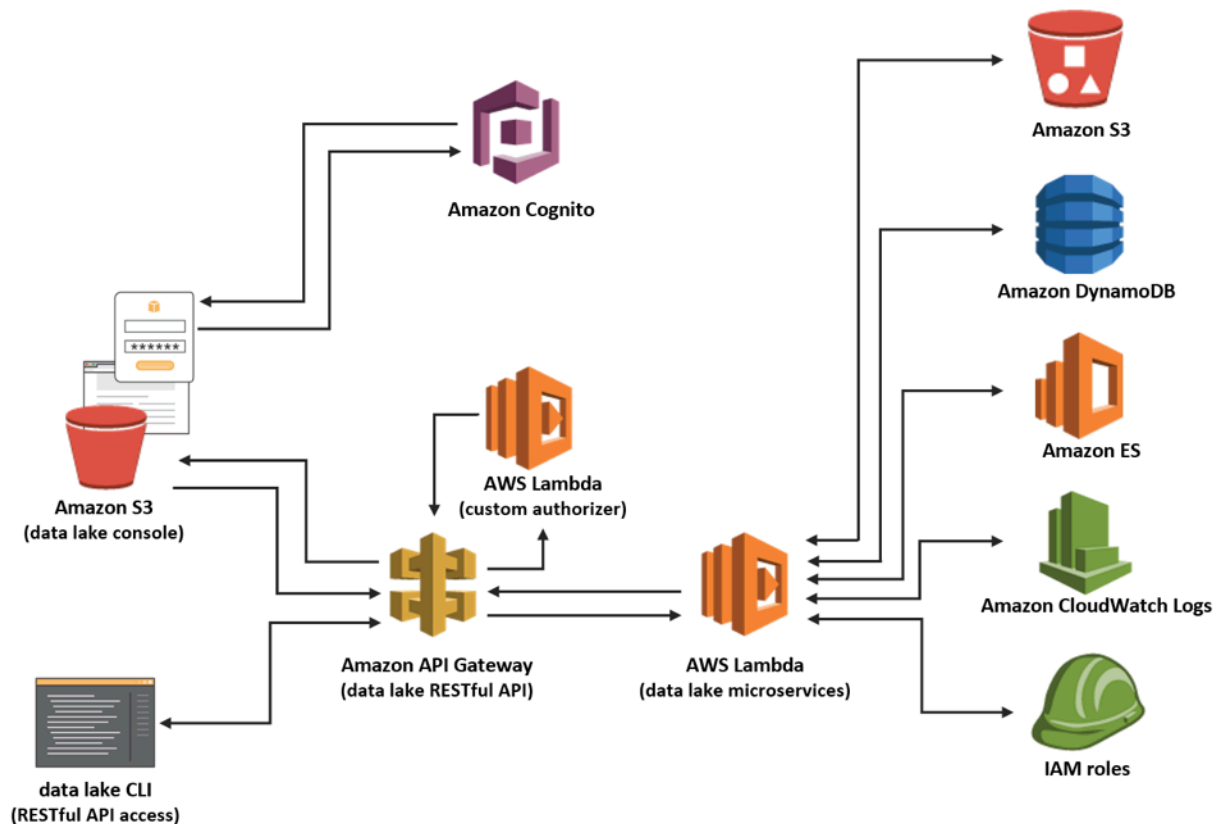
### Option 2:

Customers can spin up multiple EC2 instances and download the reference data directly from S3. In this approach, multiple EC2 instances concurrently download the same object from S3 into local EBS volumes. The alignment

operation can be carried out in parallel and the result file is written back into S3 as a new object.

## Data Lakes

Often times genomics researchers want to keep datasets and their iterations within organizational boundaries. They want to leverage a data lake to complement, rather than replace, existing Data Warehouses. A data lake can be used as a source for both structured and unstructured data. These high interest datasets can easily be converted into a well-defined schema for ingestion into a Data Warehouse, or analyzed ad hoc to quickly explore unknown datasets and discover new insights. Collections of Datasets/Data Warehouses only become valuable to an organization when these data repositories can easily be indexed and searched, leading to opportunities to aggregate and enrich existing datasets into new, even more valuable, datasets. Within this need in mind, AWS created the Data Lake Solution. This solution automatically configures the core AWS services necessary to easily tag, search, share, and govern specific subsets of data across a company or with other external users. The solution deploys a console (and an API) that users and applications can access to search and browse available datasets for their business needs. It is delivered in the form of a package of open source code deployed via a Cloud Formation template. As such developers supporting research organizations have the ability to change the solution to meet the specific needs of their organization.



### Serverless Data Lake on AWS

## Processing

When developing your genomics applications, you might choose to scale your analysis horizontally, using more nodes or vertically, using more resources on a single node, or both. With AWS, you can easily both lower the time for your analysis to complete and concurrently increase your throughput, the number of analyses that can be completed in a given amount of time. You can scale an individual analysis over multiple nodes for improved time-to-insight while concurrently scaling the number of processes for improved throughput.

Individual analysis steps such as read alignment or variant calling can be modularized into "massively" or "embarrassingly" parallel processes that can be scaled horizontally for improved latency. Below, we will discuss common architectural patterns for processing genomes at scale on the AWS Cloud. At the

end of this section, we will highlight customer solutions that relate to the architectural patterns we describe below.

## Batch Processing in Genomics

Pipelines used to process genomic data bear a strong resemblance to a series of Extract Transform and Load ("ETL") steps which convert raw data from a DNA sequencer ultimately into a list of variants for a single or set of individuals. Each step in the pipeline can extract from a set of input files or the output of a prior step, perform compute-intensive processing of the extracted data (i.e. transform) and then load the output into another location for longer term storage or subsequent analysis.

Any of the pipeline steps used to Transform, such as the assembly of raw data into a larger set of sequences or the alignment of DNA sequences to a reference genome, can be considered as a discrete compute operation which is able to be packaged along with all of its dependencies. On AWS, customers often implement these pipelines as batch workloads using one or both of the following architectures:

- **Amazon Machine Image (AMI)-based:** Some analytical modules or processes within your genomics workflow, such as those with a high memory requirement or processes that present additional complexity that inhibits containerization, such as a requirement to use FPGAs, may be better suited to utilize the complete and direct resources available to an entire AWS Instance. When such needs are present, all software and environment information can be bundled into a single AMI-based image which can be loaded onto an appropriate instance. Tightly coupled workloads are also best executed using the AMI-based approach as multiple instances on EC2 can be associated into a placement group to facilitate low latency communication between related member instances.

- **Container-based:** Containerization using Docker or similar services have become a common way to provide effective application-level virtualization and encapsulate individual processes. As the elements of a typical genomics workflow (pipeline) accept input, process and then produce output, containers present an excellent option for processing in genomics. Because they are lightweight and only require the software, immediate dependencies and environment for the bundled software, containers have become a common choice in the genomics community for sharing analysis methods between

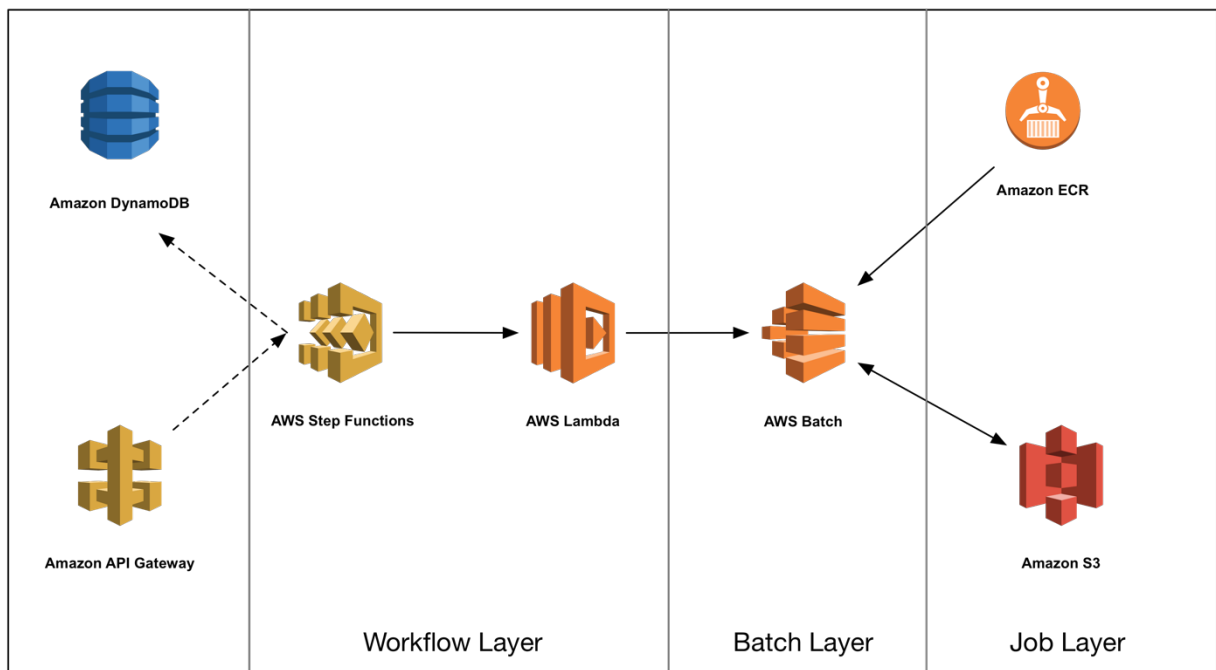
collaborators. Containers can also provide a more complete path to reproducible analyses by encapsulating all or portions of an analysis workflow in a single, fixed representation. When processes do not consume the full capacity or capability of compute resources available to a single instance, containerization can be used to fill that instance to capacity by packing multiple containers onto the same instance. AWS provides two services which enable you to deploy containers across a cluster of EC2 instances and orchestrate their activity. [AWS EC2 Container Service \(ECS\)](#) and [AWS Batch](#) work together to deploy containers and orchestrate execution of defined processes (jobs) against your data in either a managed or un-managed compute environment.

**The container-based batch workflows can be split into three separate components:**

1. **Analytical Module (Jobs):** Individual units of work. For genomics, these might include alignment, variant calling, quality control, or another module in your workflow. Amazon ECS orchestrates and runs these containers on top of Amazon EC2 instances. You might also choose to use other services for container orchestration, such as Docker Swarm or Kubernetes, running on top of Amazon EC2.
2. **Batch Engine:** This is a framework for submitting individual analytical modules with the appropriate requirements for computational resources, such as memory and number of CPUs. Each step of the analysis pipeline requires a definition of how to run a job: computational resources (disk, memory, CPU); the compute environment to run it in (Docker container, runtime parameters); information about the priority of different jobs; and any dependencies between jobs. You can leverage concepts such as container placement and bin packing to maximize performance of your genomic pipeline while concurrently optimizing for cost. The Task Placement Engine allows for fine grained job placement strategies based on instance type, memory utilization, or Availability Zone (AZ) preference. If you wish to use AWS services, AWS provides AWS Batch as a managed batch engine. AWS Batch is built on top of Amazon ECS, and you can use both managed compute environments for ease of use, as well as unmanaged compute environments for increased flexibility. You can also refer to the following reference architecture for batch processing that can be used to process genomics data sets with Amazon ECS.
3. **Workflow Orchestration:** This layer sits on top of the batch engine and allows you to decouple your workflow definition from the execution of

the individual jobs. You may envision this layer as a state machine where you define a series of steps and pass appropriate metadata between states. You can use AWS services such as AWS Step Functions and AWS Simple Workflow to define and execute these workflows. Alternatively, you can use workflow languages such as CWL and WDL to define the workflow as well as services such as Apache Airflow or Luigi to manage the entire workflow orchestration.

If you were to use native AWS services, an architecture that addresses these three layers might look like the following. Of course, each of these components may be substituted by your tool of choice, depending on your specific architectural and business considerations.



1. AWS Lambda, called manually or through Amazon API Gateway, invokes a state machine defined in AWS Step Functions.
2. The state machine orchestrates a series of Lambda Functions that both deploy the individual jobs to AWS Batch, such as alignment or variant calling, as well as track job status in Amazon DynamoDB
3. AWS Batch orchestrates each of these batch jobs on top of Amazon ECS.

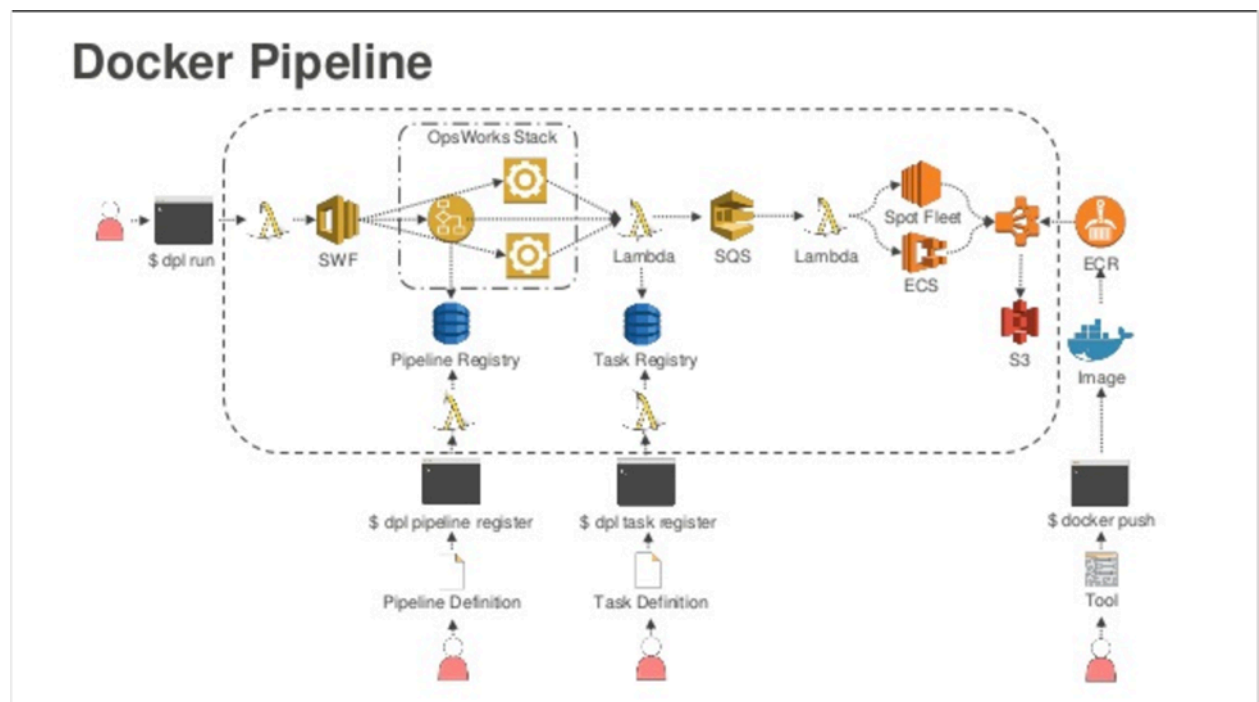
4. A Lambda Function polls AWS Batch and scales the ECS cluster as appropriate through Amazon CloudWatch Events and Auto Scaling. # Amazon S3 stores the output of each job, such as outputs from alignment (BAMs) and variant calling (VCFs/gVCFs). This check-pointing also enables workflow auditability.

### Customer Example

#### Human Longevity, Inc.

Human Longevity, Inc. (HLI) is at the forefront of genomics research and wants to build the world's largest database of human genomes along with related phenotype and clinical data, all in support of preventive healthcare. In order to easily get new analytics modules encapsulated in Docker containers into production, HLI has built a batch processing pipeline using Docker to process terabytes of genomic data every day.

HLI's Docker Pipeline uses many native AWS services. They use the AWS Flow Framework for Ruby and Amazon Simple Workflow to coordinate their bioinformatics workflows. A combination of AWS Lambda and Amazon SQS is used to execute individual jobs, and they use a combination of Amazon ECS with Spot Fleet to process their data at scale. All task data, as well as entire workflows, are stored in DynamoDB for easy auditability, versioning, and tracking.



**More Information**

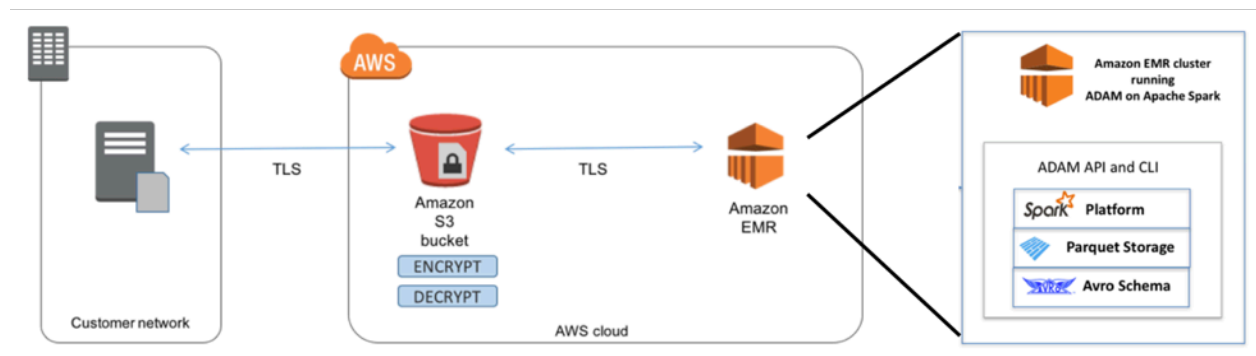
- [re:Invent talk](#)
- [TIMA](#)
- [HLI on AWS](#)

**Distributed Computing using Amazon EMR**

Amazon Elastic MapReduce (EMR) is a managed cluster platform supporting data framework applications commonly used to process and analyze high-throughput genomics datasets. A family of framework associated resources is native to the platform and includes: Apache Hadoop, Spark, Hive, Pig and Presto. Amazon EMR is designed to allow for scalability and agility, and has the added advantage of integrating well with other AWS services (ie. Databases, Amazon S3) to enable data persistence outside of the cluster lifecycle. Many features of Amazon EMR enable seamless data flow in analytical pipelines. For example, Presto capabilities in EMR enable fluid dataset import from Presto-based Amazon Athena queries on S3 data objects. Users can also take advantage of the EMR File System (EMRFS) to use S3 as a data layer for running EMR cluster applications that require data persistence. EMRFS can also auto scale worker nodes to save costs. For processing data on the master and core nodes of your cluster, EMR also supports the Hadoop Distributed File System (HDFS) although it should be noted that this provides ephemeral (cluster-dependent-temporary) storage and therefore, any application outputs should be backed up to S3 prior to cluster termination (even if using EBS volumes with EMR). As previously mentioned, Event Notifications set on an S3 Bucket can launch a lambda function wrapped Data Pipeline in order to run parallelized Hadoop EMR jobs. Features such as 'HadoopActivity' enable selection of a scheduler for workflow submission to the cluster. Job level monitoring is available and jobs can be assigned to specific queues using scheduling pools. Although widely used for computationally intensive workloads, population-scale genomics variant analysis does not fit well within a Hadoop MapReduce architecture. Alternatively, EMR SPARK and the associated machine learning library (MLlib) provides in-memory caching and increased flexibility for designing applications not restricted to the MapReduce paradigm and leveraging stage-oriented scheduling via a DAGScheduler. As a result, several genomics based resources



leveraging SPARK capabilities have been developed including SPARKSEQ<sup>1</sup> for sequence data analysis, VariantSPARK<sup>2</sup> for machine learning enabled variant clustering, and the ADAM<sup>3</sup> resource that provides a set of formats and APIs enabling processing stage implementations for genomics datasets. Variant calling for these resources as well as other customized implementations can be done using HellBender, a GATK<sup>4</sup>/Picard engine that can be run using SPARK<sup>4</sup>. For cost optimization, instances available in the spot market can also be used in EMR compute clusters. In addition, Amazon EMR provides easy to implement end-to-end encryption options to ensure HIPAA compliant data security for at-rest data stored in S3, as well as during the in-transit and processing phases involved in data analytics (Figure below).



**End-to-End Encryption and HIPAA Compliance 5,6.**

## AWS Serverless Architectures for Genomics Research

With the development of virtual machines, the machine became the unit of scale and the hardware was an abstracted entity no longer needing to be managed by a customer. In the previous section we introduced containers, where the application is the unit of scale and the operating system (OS) is the abstracted entity. When discussing ‘serverless architectures’, the AWS lambda functions of code become the unit of scale and the language runtime becomes the abstracted entity. Because users do not need to actively manage the underlying instances running in container services, lambda architectures are referred to as “serverless”. Many genomics researchers are leveraging AWS serverless architectures for large-scale computational tasks that are reiterative, for example genome-wide sequence searches that are a necessary step in identifying CRISPR gene-editing target sites. In these cases, lambda execution of code

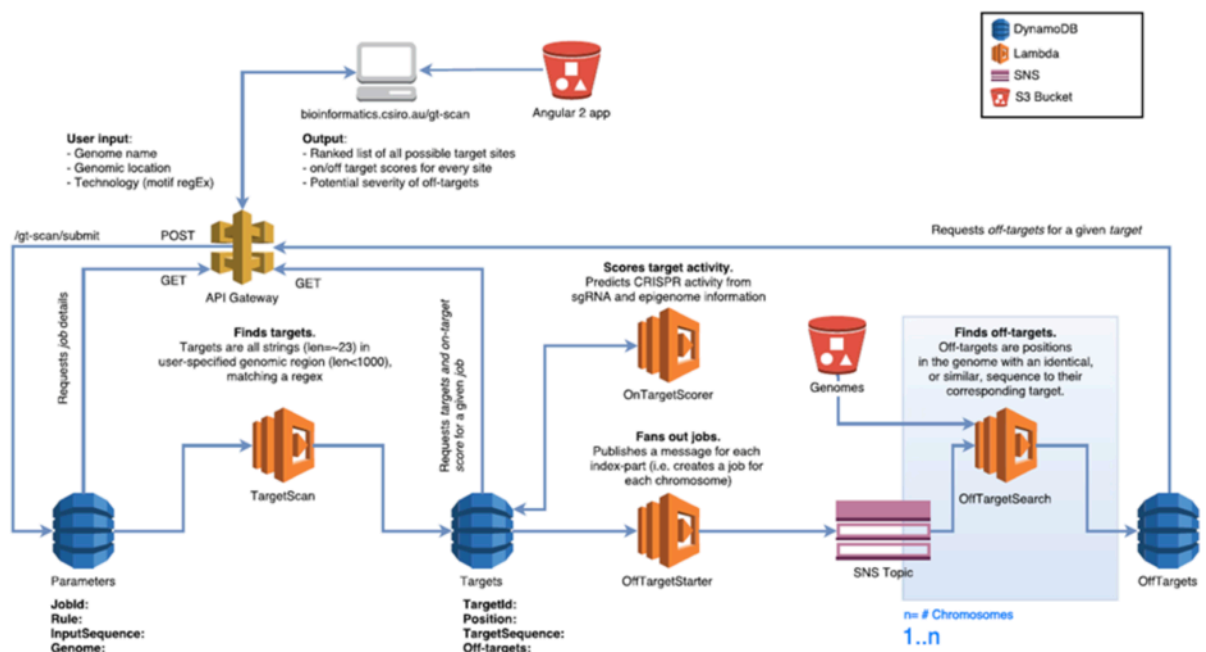
keeps the runtime stable in comparison to having to manage repeated start-up of EC2

instances, and therefore significantly expedites repetitive search processes. Example architectures leveraging AWS Lambda and additional AWS services and features in CRISPR research have been provided by the CSIRO in Australia and a start-up company, Benchling.

### Customer Example CSIRO and Benchling

Using the AWS platform, a transformational bioinformatics team headed by Dr. Denis Bauer at the eHealth program of the Commonwealth Scientific and Industrial Research Organization (CSIRO) in Australia has developed GT-Scan2. The GT-Scan2 application is a novel software tool used in designing efficient CRISPR-Cas9 studies for time-sensitive clinical care use cases (Figure below).

**GT-Scan2 Microservice-based target-finder for genome editing technologies**

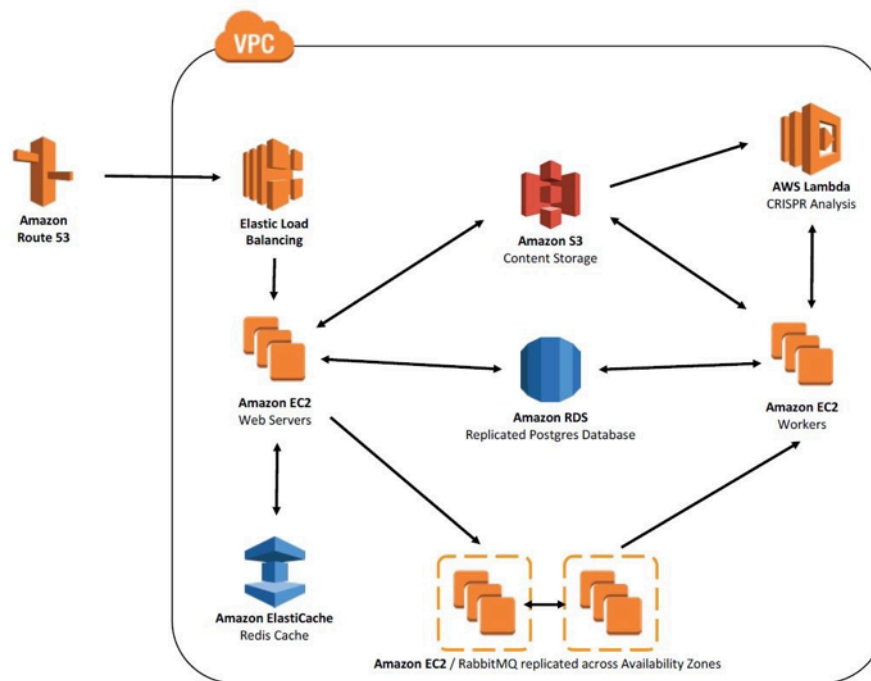


**GT-Scan2 application.**

GT-Scan2 is served directly from S3 making it a static web app without server-side processing. It retrieves the dynamic content (such as job results and

parameters) via API calls using API Gateway from a database (DynamoDB) using JavaScript.

[Benchling](#) is a start-up life sciences company based in San Francisco that provides a R&D platform enabling scientists to design, plan and collaborate at every stage of their experiment. As part of their platform resources for analytics, they provide a web server based CRISPR analytical tool (Figure below).



In the serverless architecture developed by Benchling, the web server receives a CRISPR search request which triggers a split of the specific target genome for smaller batch, parallelized lambda functions. After the lambda function performs the query on genome data stored in S3, the results are concatenated and returned to the user.

## Accelerating Genomics Algorithms on AWS

Many Precision Medicine based initiatives need to prioritize a reduced time-to-insight for workloads that require rapid analyses, such as newborn-screening or oncology. AWS enables you to accelerate these algorithms using CPU, GPU, or FPGA based approaches. For example, you can vertically scale your CPU compute to reduce latency for genomic algorithms that demand tightly-coupled processing such as in the case of de-novo alignment studies or Tumor

heterogeneity reconstructions. The AWS GPU P-class instance types are also available for massively parallel computation of large datasets that you might be using to train or run deep learning algorithms. P2 instances are equipped with GPUDirect™ (peer-to-peer GPU communication) capabilities for up to 16 GPUs, so that multiple GPUs can work together within a single host, and ENA-based Enhanced Networking for cluster P2 instances.

The recent release of the first compute instance type ("F1") with customer programmable FPGA hardware for application acceleration has already disrupted the way data flow applications are run, including genomics workflows. For example, Edico Genome has already developed DRAGEN™, the world's first bioinformatics processor that uses a field-programmable gate array (FPGA) to provide hardware-accelerated implementations of genome pipeline algorithms and data compression. Their FPGA enabled GATK variant calling workflow reportedly reduces whole genome data analysis from hours to ~20 minutes while providing levels of accuracy [comparable to BWA-MEM+GATK methodology](#), at a decreased cost to the customer. In addition to whole genome and exome variant focused pipelines, Edico Genome also has developed FPGA implementations of other pipelines used for studies of gene expression, epigenomic modifications, the microbiome, and cancer.

Amazon F1 instances include Hardware Developer Kit (HDK) and a developer AMI for creating Amazon FPGA Images (AFIs) that are reusable across F1 instances. The Vivado Design Suite is also supported for ultra-high productivity with next generation C/C++ and IP-based design. Thus, researchers interested in an approximate speed up of ~30X for their applications now have easy access to FPGA enabled resources for creating their own service or product.

## Analysis

Numerous analytical resources that are commonly used for 'tertiary level' or comparative genomic data analytics are accessible on the AWS platform, many of which are readily available as fully managed services. For any 3rd party or custom solutions that users would like to develop, install and/or manage independently, the flexibility and breadth of the AWS platform also provides the infrastructure to enable these types of integration projects. Outputs from data processing runs described in the previous section such as collections of sequence variants are commonly aggregated and analyzed by implementing a caching solution (ie. Amazon ElastiCache) together with a relational database management resource such as Redshift.

In addition to relational database based solutions, AWS also provides NoSQL based resources such as DynamoDB which provides scalable single-digit millisecond latency and the Amazon Elasticsearch managed cluster for NoSQL-type queries. These resources are highly valuable for scalable and fast fine-grained filtering or selection based analytics on large collections of raw variant calls. Multi-approach solutions implementing caching, database, and/or managed data frame compute (ie Athena, EMR-based) analytics also facilitates computationally intensive joining of data points with associated clinical metadata, as is the case in many genotype-phenotype association studies.

In addition to providing a powerful framework for running genomics algorithms, Spark on Amazon EMR is also equipped for building customized machine learning solutions using the Spark MLlib. Other programs commonly used by genomics researchers for statistical analysis, deep learning modelling and real-time analytics are also easily run on Amazon EMR and include: R/RStudio, IPython/Jupyter notebooks, Apache Zeppelin, Kibana, and more. In addition, Jupyter and deep learning programs such as MXNet, CNTK, Caffe, Theano, Torch, and TensorFlow are also available as AMIs in the AWS Marketplace to be used in conjunction with our GPU P-class instances. The P2 instances are equipped with GPUDirect™ (peer-to-peer GPU communication) capabilities for up to 16 GPUs, so that multiple GPUs can work together within a single host, and ENA-based Enhanced Networking for cluster P2 instances.

Researchers can also use the Amazon Machine Learning (ML) service to create a custom predictive model from input training datasets. A user-friendly dashboard provides researchers with a menu to specify source data, ML models, generate evaluations, and run batch predictions. ML models include regression, binary classification, and multiclass classification. For training each model, the Amazon ML service provides industry standard learning algorithms. The Amazon ML service is also REST interface accessible, making it easy to programmatically build customized ML solutions.

Finally, third party applications (ie. Tableau) as well as additional AWS services such as Amazon Quicksight are also readily available for visualization and analytical summaries, plots or graphs of data elements as well as comparative analysis with other existing datasets. Although presently more commonly used for genomics associated phenotype/clinical data, the AWS Cloud also provides streaming data analytics capabilities with managed resources such as Amazon

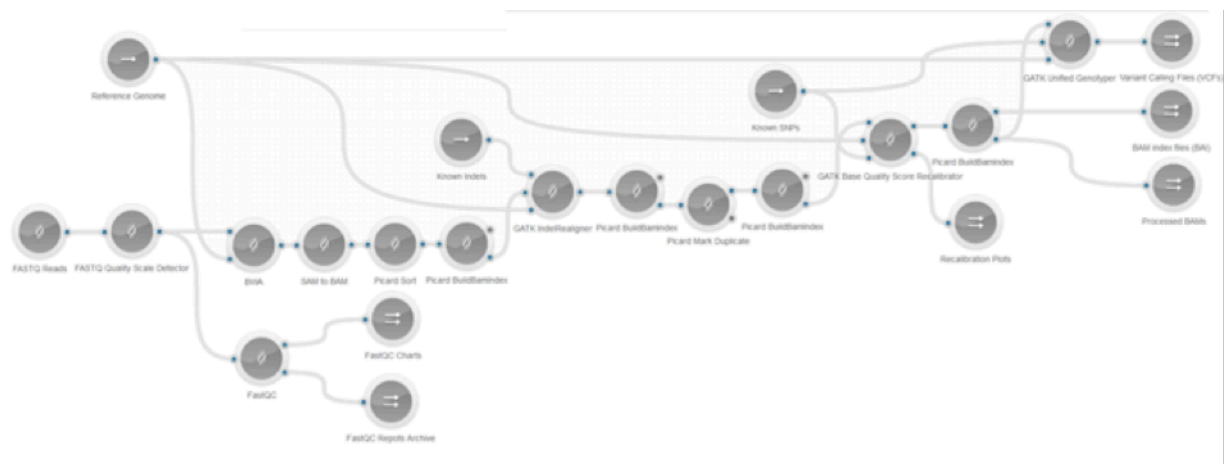
Kinesis, Kinesis Client Library Applications or by easy integration of Kinesis with additional resources such as Apache Storm. Streaming analytics is also supported by DynamoDB Streams, SPARK Streaming, and AWS Lambda.

## Managed Services for Genomics on AWS

### Seven Bridges Genomics

[Seven Bridges Genomics](#), an AWS Life Sciences Competency Partner focusing on genomics, offers researchers and labs a cloud platform for analyzing genetic data generated through next-generation sequencing (NGS) technologies. Through its IGOR platform, Seven Bridges provides a one-stop solution for managing NGS projects and enables customers to create and run complex data analysis pipelines easily using a drag-and-drop interface. The AWS Cloud provides highly scalable computation and the means to easily consume, share and reproduce results.

To overcome the challenges of processing data at scale, Seven Bridges Genomics developed IGOR, a cloud-based genetic data analysis and discovery platform. Seven Bridges' IGOR platform provides customers with a graphical user interface to set up complex data processing pipelines, either by using an existing pipeline as a model or by creating a new one. They also have deep integration with CWL, which allows you to easily write and share your workflow-based analyses. An example of this can be seen below:



### DNANexus

[DNANexus](#) provides an API-based platform for the storage, processing, and comparative analytics of complex and diverse genomics and healthcare associated datasets. Using the DNANexus platform, clinical and life sciences

researchers can scale customized analytical workloads using the latest cloud-enabled services and features in a secure environment tailored for compliance to standards such as HIPAA, ISO27001, SOC 1/2/3, ISO9001, FedRamp, and FISMA. The DNAnexus Platform enables scientists and clinicians worldwide to accelerate medical advances, improve patient care, and advance research and development in areas such as cancer, heart disease, Alzheimer's disease, noninvasive prenatal testing, and enhanced agricultural production.

## Sharing

So far, we have come to see that the AWS platform provides numerous tools to build highly scalable, cost efficient and elastic data processing pipelines. Be it getting data to the compute processing with ease at low cost, or innovations in pipeline and processing algorithm development, or just-in-time provisioning of data querying platforms that allow for quick insights into the processed data, all of these facets are useless without a means to share and collaborate on the findings produced by the previous phases of the data processing pipeline. Therefore data analysis and data distribution go hand-in-hand. Data distribution, identified here as the means of sharing data to multiple interested parties, is a critical phase in the genomic processing pipeline. The AWS cloud platform helps scientists focus on science and collaboration, not servers -- all with minimal effort and confidence that your data and budget are secure and optimized. The following describes some of the capabilities provided by the AWS platform to enable distribution of AWS Cloud hosted data. For more information on what AWS is doing in the Science and Research domain, visit the AWS research and technical computing site

## Global Data Egress Waiver

AWS measures 3 types of actions taken against data used by its cloud infrastructure; Ingress - measuring data into the cloud, Storage/Requests - measuring data hosted ON (or requested from) cloud storage services like S3, Egress - data transfer out of cloud storage to be delivered to a requesting entity using AWS network infrastructure. Ingress is currently no charge to the customer. Storage is dependent on data amounts stored/requested in the cloud, and the service used. Egress has historically been difficult to measure by the scientific community since it is often workload dependent and a challenge to plan for when arranging organizational budgeting exercises. To help offset this egress entropy and make cloud use budgeting more predictable, AWS waives data egress fees from the AWS Cloud, for qualified researchers and academic



customers. For more information on how to take advantage of the Global Data Egress Waiver, please see [here](#)

## Peering with Global Research Networks

Often times genomics research requires interaction with other research institutes that leverage common research networks, such as Internet2, ES.NET and others. By peering with Global Research Networks, AWS gives researchers robust network connections to the AWS cloud that are often orders of magnitude more performant than standard internet connections. These network connections allow for reliable movement of data between your home institution, distributed data collection sites, and AWS. For more information on the research networks AWS works with, please see [here](#)

## AMI Sharing

Not only is shared data of high value to researchers and scientists in the genomics domain, but sharing tools and environments is critical in assuring that common experiment environments can be leveraged in efficient ways by collaborators and publication validators. A shared AMI is an Amazon Machine Image that an innovator created and made available for other collaborators to use. Instead of trying to replicate a research environment from scratch with all the tools and configurations necessary to replicate experiments, one can use a shared AMI that has the components you need and then add custom content once the AMI instantiates as a live EC2 instance within a VPC. You can also share a specific AMI with a given collaborator's AWS Account, without making the AMI public.

## Public Datasets

AWS is committed to collaboration and offering more and more ways for our customers to gain valuable scientific insights to pivot and iterate off of. With this in mind, many of AWS' genomics customers and researchers have found high value in the [Public Datasets program](#). The value add comes from integrating these datasets into the development of the genomic processing pipeline as a testbed for innovative solutions. When organizations make data open on AWS, scientists can access and analyze it, delivering innovative solutions to big challenges. AWS makes a variety of Public Data Sets available to researchers and the public to copy for free, and share amongst their colleagues. Common AWS sponsored open datasets that genomics customers find of high value are the following:



- [The Cancer Genome Atlas](#): Raw and processed genomic, transcriptomic, and epigenomic data from The Cancer Genome Atlas (TCGA) available to qualified researchers via the Cancer Genomics Cloud
- [1000 Genomes Project](#): A detailed map of human variation
- [Genome in a Bottle \(GIAB\)](#): Several reference genomes to enable translation of whole human genome sequencing to clinical practice
- [3000 Rice Genome on AWS](#): Genome sequence of 3,024 rice varieties

The public datasets are hosted in two possible formats: Amazon Elastic Block Store (Amazon EBS) snapshots and/or Amazon Simple Storage Service (Amazon S3) buckets. To access a public dataset hosted in Amazon S3: You can make simple HTTP requests, use AWS Command Line Tools and SDKs (Ruby, Java, Python, .NET, PHP, etc.), download the data using Amazon EC2, or use Hadoop to process the data with Amazon EMR. To access a dataset hosted as an Amazon EBS snapshot: Sign up for an AWS account, launch an Amazon EC2 instance, and create an Amazon EBS volume using the Snapshot ID listed in one of the links above.

If you have any questions or want to participate in our Public Datasets community, please email us at [opendata@amazon.com](mailto:opendata@amazon.com).

## Conclusion

We hope this guide was informative and helpful. The production of valuable genomic data requires careful consideration over several stages ranging from acquisition to storage, onto compute and finally to distribution. AWS presents a wide range of capabilities that can be leveraged for each stage and help researchers unlock the potential inside of genomic data.

## Document Revisions

Date	Description
August 2017	Initial content made publication ready
September 2017	Updated formatting