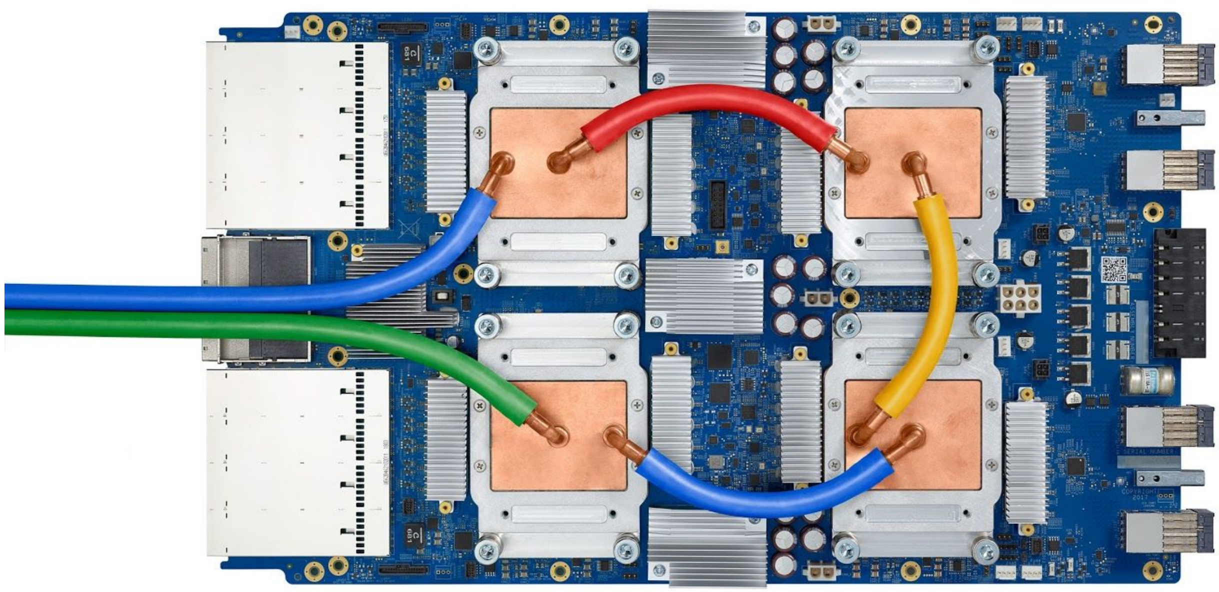
Google Cloud

Migrating Research Workloads and Data to GCP

Cloud training, resources, and reference architectures for NIH researchers and data scientists.



For more information visit **cloud.google.com/solutions/healthcare/**

Contents

Introduction

The National Institutes of Health (NIH) established The Science and Technology Research Infrastructure for Discovery, Experimentation, and Sustainability (STRIDES) initiative to provide biomedical researchers with access to advanced, cost-effective, cloud-based computational infrastructure, tools, and services. Through STRIDES, researchers can take advantage of emerging data management methodologies, technological expertise, computational platforms, and tools to support cutting-edge experimentation and innovation. NIH has partnered with Google Cloud to support the STRIDES initiative through cloud services. In support of STRIDES, we've developed sets of playbooks to help enable researchers to build healthcare and life sciences solutions on Google Cloud Platform (GCP).

The goal of this playbook is to aid researchers in migrating existing research workloads and data onto GCP. This playbook will provide NIH-specific processes, sample GCP architectures, and examples for migrating existing research projects and data to GCP. Additionally, this playbook will outline training and digital resources to help upskill and enable researchers to build on Google Cloud, while highlighting the appropriate products and services to use when architecting on GCP.

This document is a customized version of Google's "Migration to Google Cloud: Getting started." This document is part of a series of Google migration documents that include the following:

- Migration to Google Cloud: Getting started (this document)
- Migration to Google Cloud: Assessing and discovering your workloads
- Migration to Google Cloud: Building your foundation
- Migration to Google Cloud: Transferring your large datasets
- Migration to Google Cloud: Deploying your workloads
- Migration to Google Cloud: Migrating from manual deployments to automated, containerized deployments
- Migration to Google Cloud: Optimizing your environment

Learning

Generally, cloud adopters fall under one of three categories:

Cloud Novice	Cloud Ready	Cloud Native
Little to no understanding of the cloud	Familiar with the cloud, some experience	Lots of cloud experience, expert-level knowledge

Understanding this broad spectrum of experience levels, we've highlighted key training resources to help upskill researchers on Google Cloud. Additionally, Google offers on-site, instructor-led training to enable large groups of participants across your organization.

Cloud Novice

Video: Welcome to GCP

Video: Intro to GCP for Students

Video: GCP 101

Video: GCP Essentials

Documentation: GCP Conceptual Overview

Documentation: About GCP Services

Documentation: GCP Development & Admin Tools

Documentation: All GCP Products & Services

Virtual Course: GCP Fundamentals - Core Infrastructure

Virtual Lab: GCP Essentials

Cloud Ready

Documentation: All GCP Products & Services

Documentation: GCP for Data Center Professionals

Documentation: GCP for AWS Professionals

Documentation: GCP for Azure Professionals

Documentation: GCP for OpenStack Users

Video: The Future of Health

Video: Uncovering Clinical Insights

Video: Medical Imaging 2.0
Virtual Course: GCP Fundamentals - Core Infrastructure

Virtual Course: Essential Cloud Infrastructure - Foundation

Virtual Course: Essential Cloud Infrastructure - Core Services

Virtual Course: Elastic Cloud Infrastructure - Scaling and Automation

Virtual Course: Elastic Cloud Infrastructure - Containers and Services

Virtual Lab: Cloud Architecture

Virtual Lab: Cloud Engineering

VIrtual Lab: Cloud Development

Cloud Native

Documentation: All GCP Products & Services

Documentation: GCP Solutions

Documentation: Cloud Healthcare API

Video: Healthcare in the Cloud

Video: Predictions with Healthcare API

Video: Transform Healthcare with ML

Video: Agility in Healthcare

Video: Bringing Clinical Data to GCP

Video: Mining Clinical Notes with GCP

Video: Health Taxonomy - NLP for Health Research

Video: Genomic Analysis on GCP

Virtual Course: Architecting with Google Cloud Platform

Virtual Course: Big Data and Machine Learning Fundamentals

Virtual Course: Leveraging Unstructured Data with Cloud Dataproc on Google Cloud Platform

Virtual Course: Serverless Data Analysis with Google BigQuery and Cloud Dataflow

Virtual Course: Serverless Machine Learning with Tensorflow on Google Cloud Platform

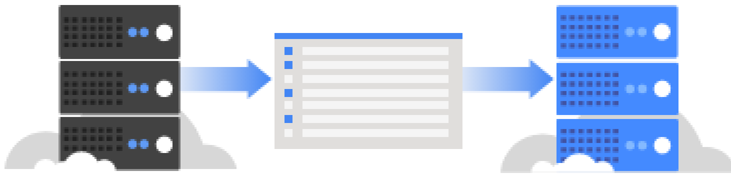
Virtual Course: Building Resilient Streaming Systems on Google Cloud Platform

Virtual Lab: Data Engineering

Virtual Lab: Data Science on GCP

Virtual Lab: Data Science on GCP - Machine Learning

Virtual Lab: Google Cloud Solutions II - Data and Machine Learning



Migrating to Google Cloud

Migrating research workloads to Google Cloud is a process that can be broken down into three steps. This guide provides instructions for completing each step. The steps are outlined below:

1. **Sign STRIDES agreement**
2. **Configure your Google Cloud Account**
3. **Migrate workloads to GCP using the Google Cloud Adoption Framework (GCAF)**

Overview: Migrating to Google Cloud

Google offers powerful, flexible infrastructure capable of supporting scalable workloads. This allows researchers to work with data at a larger scale and faster than before, decreasing time from hypothesis to insight. GCP can enable and accelerate time to results for tasks from analyzing genomic sequence data to simulating complex climate models.

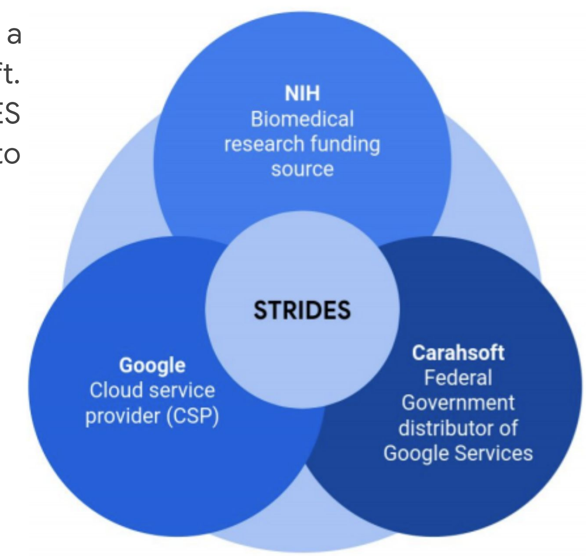
This document helps you plan, design, and implement the process of migrating your workloads onto the NIH.gov environment hosted on Google Cloud. The topics addressed in this document include:

- NIH STRIDES Agreement
- Summary of the NIH.gov environment
- Available NIH.gov resources
- Available GCP resources and services
- Types of GCP Migrations
- Google Cloud Adoption Framework
- Finding Help
- What's next

Moving applications and workloads from one environment to another is a challenging task, even for experienced teams, so you need to plan and execute your migration carefully.

Step 1: Sign STRIDES agreement

The STRIDES Initiative for Google Cloud Platform (GCP) is a partnership between NIH, Google, and Carahsoft. Migrating research and workloads to the NIH STRIDES environment on GCP provides your team with access to certain benefits including:



- **Discounts on STRIDES Initiative partner services**—Favorable pricing on computing, storage, and related cloud services for NIH Institutes, Centers, and Offices (ICOs) and NIH-funded institutions.
- **Professional services**—Access to professional service consultations and technical support from the STRIDES Initiative partners.
- **Training**—Access to training for researchers, data owners, and others to help ensure optimal use of available tools and technologies.
- **Best practices** in areas such as data storage, governance, and controlled access

Action 1 - Ask your authorized financial representative (e.g. Finance/Procurement/Contracting Officer) to review and sign the Customer Order Agreement (COA). This includes Google Cloud's General Terms of Service, with the option to honor Internet2 Net+ Terms of Service. This does not constitute a commitment - you only pay for what you use.

Action 2 - Once the COA has been signed, Principal Investigators (PIs) will be able to request a STRIDES billing account for their NIH Award.

Step 2: Configure Google Cloud Account

After requesting and receiving a STRIDES billing account, use the enterprise onboarding checklist to instantiate a Google Cloud Platform environment, if one has not already been setup. If a GCP environment already exists, skip to set up billing.

Enterprise onboarding checklist:

1. Set up a Cloud Identity account
2. Add users and groups to your Cloud Identity account
3. Set up administrator access to your organization
4. Set up billing

    Adding a STRIDES billing account ID

    Once you have a STRIDES billing account ID, the billing account administrator will need to log in to the GCP console and complete the following:

    1. Click the navigation menu, and select Billing to view your GCP billing account info
    2. In the left-side Billing panel, select Account management to view your current list of GCP projects
    3. Click the 3-dot option button beside your GCP project ID, and select Change billing. Note: Only an
    admin will have access to modify billing
    4. Apply the STRIDES billing account ID to the project, and click Set Account.
    5. Repeat steps 2 - 5 to add additional projects to the STRIDES billing account ID.

    ***Note:** As a billing account administrator, you'll be able to monitor how much compute power members of your organization are consuming.*
    ***Reference:** Modify a Project's Billing Settings*

5. Set up the resource hierarchy
6. Set up access control for your resource hierarchy
7. Set up support
8. Set up your networking configuration
9. Set up logging and monitoring
10. Configure security settings for apps and data

Follow Google's best practices for enterprise organizations when setting up your GCP environment and projects.

Step 3: Migrate workloads to GCP using the Google Cloud Adoption Framework

Beginning the journey

When planning your migration to Google Cloud, you start by defining the environments that are involved in the migration. This migration guide covers three migration starting points:

1. **On-premises environment** - An environment where you have full ownership and responsibility. You retain full control over every aspect of the environment, such as cooling, physical security, and hardware maintenance.
2. **Private hosting environment** - An environment where you outsource part of the physical infrastructure and its management to an external party. This infrastructure is typically shared between customers. In a private hosting environment, you don't have to manage the physical security and safety services.
3. **Public cloud environment** - An environment where you don't manage the whole resource stack by yourself. You focus on the aspect of the stack that is most valuable to you, and the rest is managed by the cloud provider. Like in a private hosting environment, you don't have to manage the underlying physical infrastructure. Additionally, you don't have to manage the resource virtualization hypervisor. You can build a virtualized infrastructure and can deploy your workloads in this new infrastructure. You can also buy fully managed services, where you care only about your workloads, handing off the operational burden of managing runtime environments.

After you define your starting and target environments, you define the workload types and the related operational processes that are in scope for the migration. This document considers two types of workloads and operations: legacy and cloud-native.

Legacy workloads and operations - Developed without any consideration for cloud environments. These workloads and operations can be difficult to modify and expensive to run and maintain because they usually don't support any type of scalability.

Cloud-native workloads and operations - Natively scalable, portable, available, and secure. These workloads and operations can help increase developer productivity and agility, because developers can focus on the actual workloads, rather than spending effort to manage development and runtime environments, or dealing with manual and cumbersome deployment processes. Google Cloud also has a shared responsibility model for security. Google Cloud is responsible for the physical security and the security of the infrastructure, while you're responsible for the security of the workloads you deploy to the infrastructure.

Considering these environment and workload types, your starting situation is one of the following:

- On-premises or private hosting environment with legacy workloads and operations.
- On-premises or private hosting environment with cloud-native workloads and operations.
- Public cloud or private hosting environment with legacy workloads and operations.
- Public cloud or private hosting environment with cloud-native workloads and operations.

The migration process depends on your starting point.

Migrating a workload from a legacy on-premises environment or private hosting environment to a cloud-native environment, such as a public cloud, can be challenging and risky. Successful migrations change the workload to migrate as little as possible during the migration operations. Moving legacy on-premises apps to the cloud often requires multiple migration steps.

Types of migrations

There are three major types of migrations:

- Lift and shift
- Improve and move
- Rip and replace

In the following sections, each type of migration is defined with examples of when to use each type.

Lift and shift

In a lift and shift migration, you move workloads from a source environment to a target environment with minor or no modifications or refactoring. The modifications you apply to the workloads to migrate are only the minimum changes you need to make in order for the workloads to operate in the target environment.

A lift and shift migration is ideal when a workload can operate as-is in the target environment, or when there is little or no business need for change. This migration is the type that requires the least amount of time because the amount of refactoring is kept to a minimum.

There might be technical issues that force a lift and shift migration. If you cannot refactor a workload to migrate and cannot decommission the workload, you must use a lift and shift migration. For example, it can be difficult or impossible to modify the source code of the workload, or the build process isn't straightforward so producing new artifacts after refactoring the source code might not be possible.

Lift and shift migrations are the easiest to perform because there is no need for new expertise and your team can use the same set of tools and skills. These migrations also support off-the-shelf software. Because you migrate existing workloads with minimal refactoring, lift and shift migrations tend to be the quickest, compared to improve and move or rip and replace migrations.

On the other hand, the results of a lift and shift migration are non-cloud-native workloads running in the target environment. These workloads don't take full advantage of cloud platform features, such as horizontal scalability, fine-grained pricing, and highly managed services.

Improve and move

In an improve and move migration, you modernize the workload while migrating it. In this type of migration, you modify the workloads to take advantage of cloud-native capabilities, and not just to make them work in the new environment. You can improve each workload for performance, features, cost, or user experience.

If the current architecture or infrastructure of an app isn't supported in the target environment as it is, a certain amount of refactoring is necessary to overcome these limits.

Another reason to choose the improve and move approach is when a major update to the workload is necessary in addition to the updates you need to make to migrate.

Improve and move migrations let your app leverage features of a cloud platform, such as scalability and high availability. You can also architect the improvement to increase the portability of the app.

On the other hand, improve and move migrations take longer than lift and shift migrations, because they must be refactored in order for the app to migrate. You need to evaluate the extra time and effort as part of the life cycle of the app.

An improve and move migration also requires that you master new skills.

Rip and replace

In a rip and replace migration, you decommission an existing app and completely redesign and rewrite it as a cloud-native app.

If the current app isn't meeting your goals—for example, you don't want to maintain it, it's too costly to migrate using one of the previously mentioned approaches, or it's not supported on Google Cloud—you can do a rip and replace migration.

Rip and replace migrations let your app take full advantage of Google Cloud features, such as horizontal scalability, highly managed services, and high availability. Because you're rewriting the app from scratch, you also remove the technical debt of the existing, legacy version.

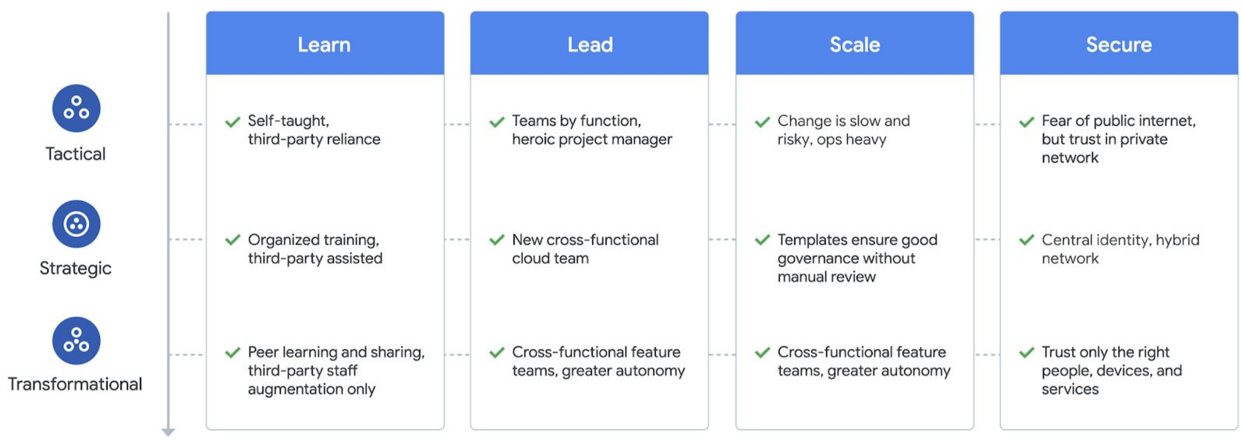
However, rip and replace migrations can take longer than lift and shift or improve and move migrations. Moreover, this type of migration isn't suitable for off-the-shelf apps because it requires rewriting the app. You need to evaluate the extra time and effort to redesign and rewrite the app as part of its lifecycle.

A rip and replace migration also requires new skills. You need to use new toolchains to provision and configure the new environment and to deploy the app in that environment.

Google Cloud Adoption Framework

Before starting your migration, you should evaluate the maturity of your organization in adopting cloud technologies. The Google Cloud Adoption Framework serves both as a map for determining where your information technology capabilities are now, and as a guide to where you want to be.

You can use this framework to assess your organization's readiness for Google Cloud and what you need to do to fill in the gaps and develop new competencies, as illustrated in the following diagram.



The framework assesses four themes:

- **Learn.** The quality and scale of your learning programs.
- **Lead.** The extent to which your IT departments are supported by a mandate from leadership to migrate to Google Cloud.
- **Scale.** The extent to which you use cloud-native services, and how much operational automation you currently have in place.
- **Secure.** The capability to protect your current environment from unauthorized and inappropriate access.

For each theme, you should be in one of the following three phases, according to the framework:

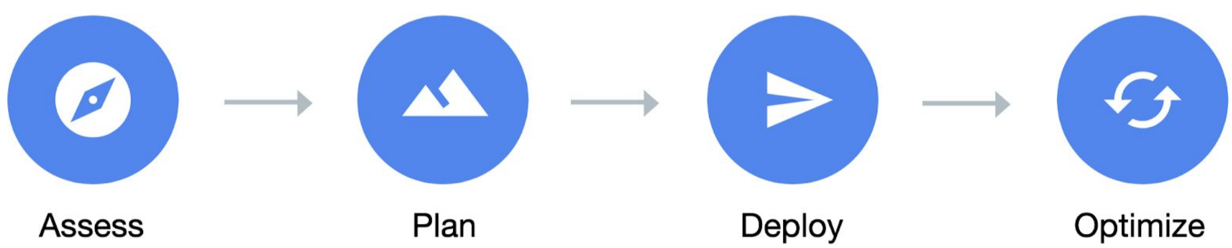
- **Tactical.** There are no coherent plans covering all the individual workloads you have in place. You're mostly interested in a quick return on investments and little disruption to your IT organization.
- **Strategic.** There is a plan in place to develop individual workloads with an eye to future scaling needs. You're interested in the mid-term goal to streamline operations to be more efficient than they are today.
- **Transformational.** Cloud operations work smoothly, and you use data that you gather from those operations to improve your IT strategy. You're interested in the long-term goal of making the IT department one of the engines of innovation in your organization.

When you evaluate the four topics in terms of the three phases, you get the Cloud Maturity Scale. In each theme, you can see what happens when you move from adopting new technologies when needed, to working with them more strategically across the organization—which naturally means deeper, more comprehensive, and more consistent training for your teams.

The migration path

It's important to remember that a migration is a journey. You are at point A with your existing infrastructure and environments, and you want to reach point B. To get from A to B, you can choose any of the options previously described.

The following diagram illustrates the path of this journey.



There are four phases of your migration:

- **Assess.** In this phase, you perform a thorough assessment and discovery of your existing environment in order to understand your app and environment inventory, identify app dependencies and requirements, perform total cost of ownership calculations, and establish app performance benchmarks.
- **Plan.** In this phase, you create the basic cloud infrastructure for your workloads to live in and plan how you will move apps. This planning includes identity management,

organization and project structure, networking, sorting your apps, and developing a prioritized migration strategy.
- **Deploy.** In this phase, you design, implement and execute a deployment process to move workloads to Google Cloud. You might also have to refine your cloud infrastructure to deal with new needs.
- **Optimize.** In this phase, you begin to take full advantage of cloud-native technologies and capabilities to expand your organization's potential to things such as performance, scalability, disaster recovery, costs, training, as well as opening the doors to machine learning and artificial intelligence integrations for your app.

Migration phase 1: Assess

In the assessment phase, you gather information about the workloads you want to migrate and their current runtime environment.

Take inventory

A key to a successful migration is understanding the data, apps, and workloads that exist in your current environment–what databases, message brokers, data warehouses, and network appliances exist–and the dependencies for each of them. You need to list all of your machines, hardware specifications, operating systems, licenses, and which of the apps and services are used by each of them.

Catalog apps

After you take your inventory, you can build your catalog matrix to help you organize your migration plans into categories based on their complexity and risk in moving to Google Cloud.

Educate your organization about Google Cloud

As part of the assess phase, your organization needs to start learning about Google Cloud. You need to train and certify your software and network engineers on how the cloud works and what Google Cloud products they can leverage as well as what kind of frameworks, APIs, and libraries they can use to deploy workloads on Google Cloud.

Experiment and design proofs of concept

Another important part of the assessment phase is choosing a proof of concept (PoC) and implementing it, or experimenting with Google Cloud products to validate use cases or any areas of uncertainty.

Calculate total cost of ownership

Building a total cost of ownership model lets you compare your costs on Google Cloud with the costs you have today. There are tools that can help you, such as the Google Cloud price calculator, and you can also leverage some of our partner offerings. Don't forget the operational costs of running on-premises or in your own data center–power, cooling, maintenance, and other support services impact the total cost of ownership.

Choose which workloads to migrate first

In order to prepare for your migration, you identify workloads with features that make them likely first-movers. You can pick just one, or include many workloads in your first-mover list. These first-movers let your teams run and test the cloud environment, where they can focus on the migration instead of on the complexity of the workloads. Starting with a less complex workload lowers your initial risk because later you can apply your team's new knowledge to harder to migrate workloads.

Identifying a first-mover can be complex, but good candidates usually satisfy many of the following workload criteria:

- Not critical, so the main line of work or research isn't impacted by the migration, because your teams don't have yet a significant experience with cloud technologies.
- Not an edge case because it's easy to apply the same pattern to other workloads that you want to migrate.
- Can be used to build a knowledge base.
- Supported by a team that is highly motivated and eager to run on Google Cloud.
- Moved by a central team that moves other workloads. Moving the first workload leads to more experience in that team, which can prove useful in future workload migrations.
- A dependency-light workload, for example, a stateless one is easier to move because they can move without impacting other workloads or with minimal configuration changes.
- Requires minimal changes or refactoring.
- Doesn't need large quantities of data moved.
- Doesn't have strict compliance requirements.
- Doesn't require third-party proprietary licenses because some providers don't license their products for the cloud or might require a change in license type.
- Not impacted by downtime caused by a cutover window. For example, you can export data from your current database and then import it to a database instance on Google Cloud during a planned maintenance window. Synchronizing two database instances to achieve a zero downtime migration is more complicated.

Migration phase 2: Plan

In this phase, you provision and configure the cloud infrastructure and services that will support your workloads on Google Cloud. Building a foundation of critical configurations and services is an evolving process. When you establish your rules, governance, and settings, make sure you allow room for changes later. Avoid making decisions that lock you in to a way of doing things. If you need to change things later on, you want to have options to support those changes.

To plan for your migration, you need to do the following:

- Establish user and service identities.
- Design your resource organization.
- Define groups and roles for resource access.
- Design your network topology and establish connectivity.

Establish identities

In Google Cloud, you have identity types to choose from:

- **Google Accounts.** An account that usually belongs to an individual user that interacts with Google Cloud.
- **Service accounts.** An account that usually belongs to an app or a service, rather than to a user.
- **Google groups.** A named collection of Google accounts.
- **G Suite domains.** A virtual group of all the Google accounts that have been created in an organization's G Suite account.
- **Cloud Identity domains.** These domains are like G Suite domains, but they don't have access to G Suite applications.

For more information, read about each identity type.

For example, you can federate Google Cloud with Active Directory to establish consistent authentication and authorization mechanisms in a hybrid environment.

Design resource organization

After establishing the identities you need for your app, you grant them permissions on resources, such as project, folders, or buckets, that your app uses. You can do this by assigning roles to each identity. A role is a collection of permissions. A permission is a collection of operations that are allowed on a resource.

To avoid repeating the same configuration steps, you can organize your resources in different types of structures. These structures are organized in a hierarchy:

- Organizations are the root of a resource hierarchy and represent a real organization, such as a company. An organization can contain folders and projects. An organization admin can grant permissions on all the resources contained in that organization.
- Folders are an additional layer of isolation between projects and can be seen as sub-organizations in the organization. A folder can contain other folders and projects. An admin can use the folder to delegate admin rights.

- Projects are the base-level organization entities and must be used to access other Google Cloud resources. Every resource instance you deploy and use is contained in a project.

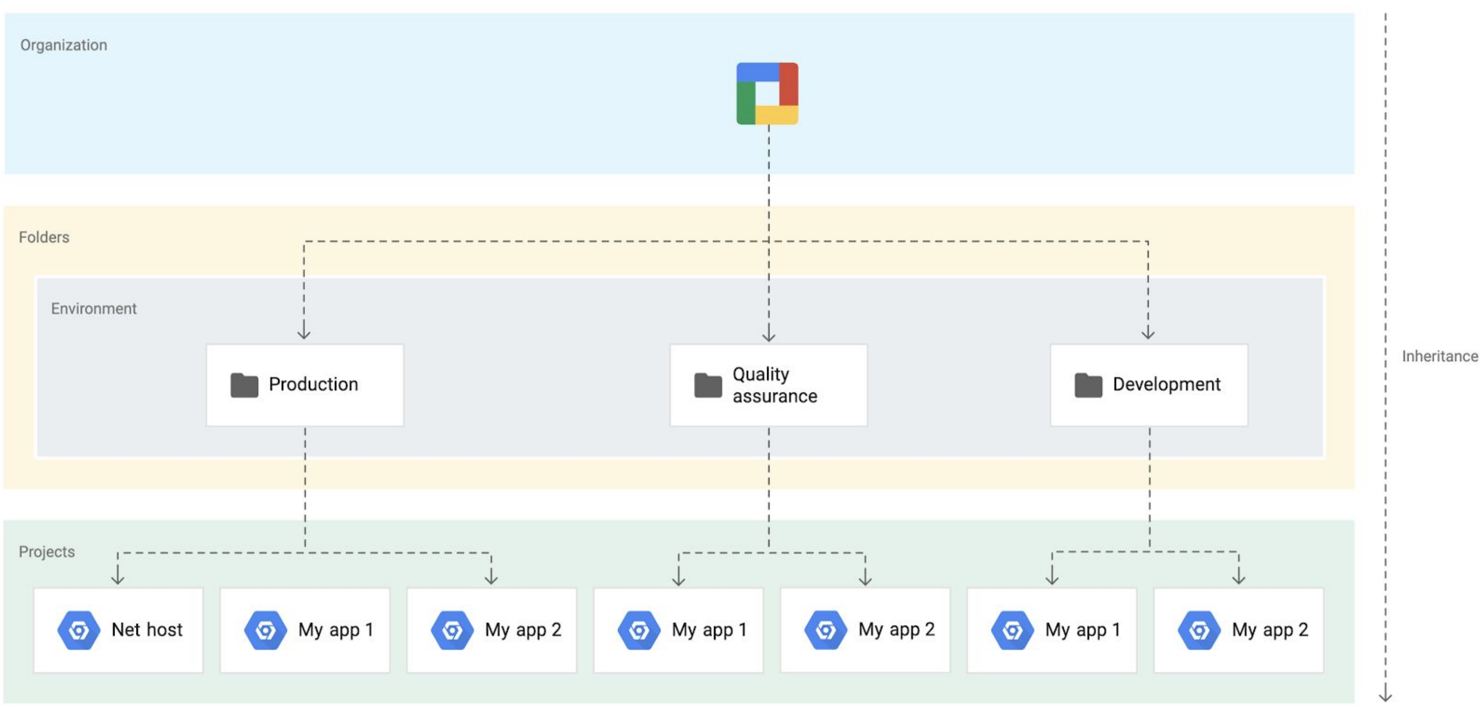
Because resources inherit permissions from the parent node, you can avoid repeating the same configuration steps for resources with the same parent. You can find more details about the Identity and Access Management (IAM) inheritance mechanism in the policy inheritance section of the Resource Manager documentation.

Organizations, folders and projects are resources and support a set of operations like all other Google Cloud resources. You can interact with these resources like you would any other Google Cloud resource. For example, you can automate the creation of your hierarchy by using the Resource Manager API. You can organize the resource hierarchy according to your needs. The root node of each hierarchy is always an organization. In the following sections, there are types of hierarchies that you can implement in your organization. Each hierarchy type is characterized by its implementation complexity and its flexibility.

Environment-oriented hierarchy

In an environment-oriented hierarchy, you have one organization that contains one folder per environment.

The following diagram shows an example of an environment-oriented hierarchy.



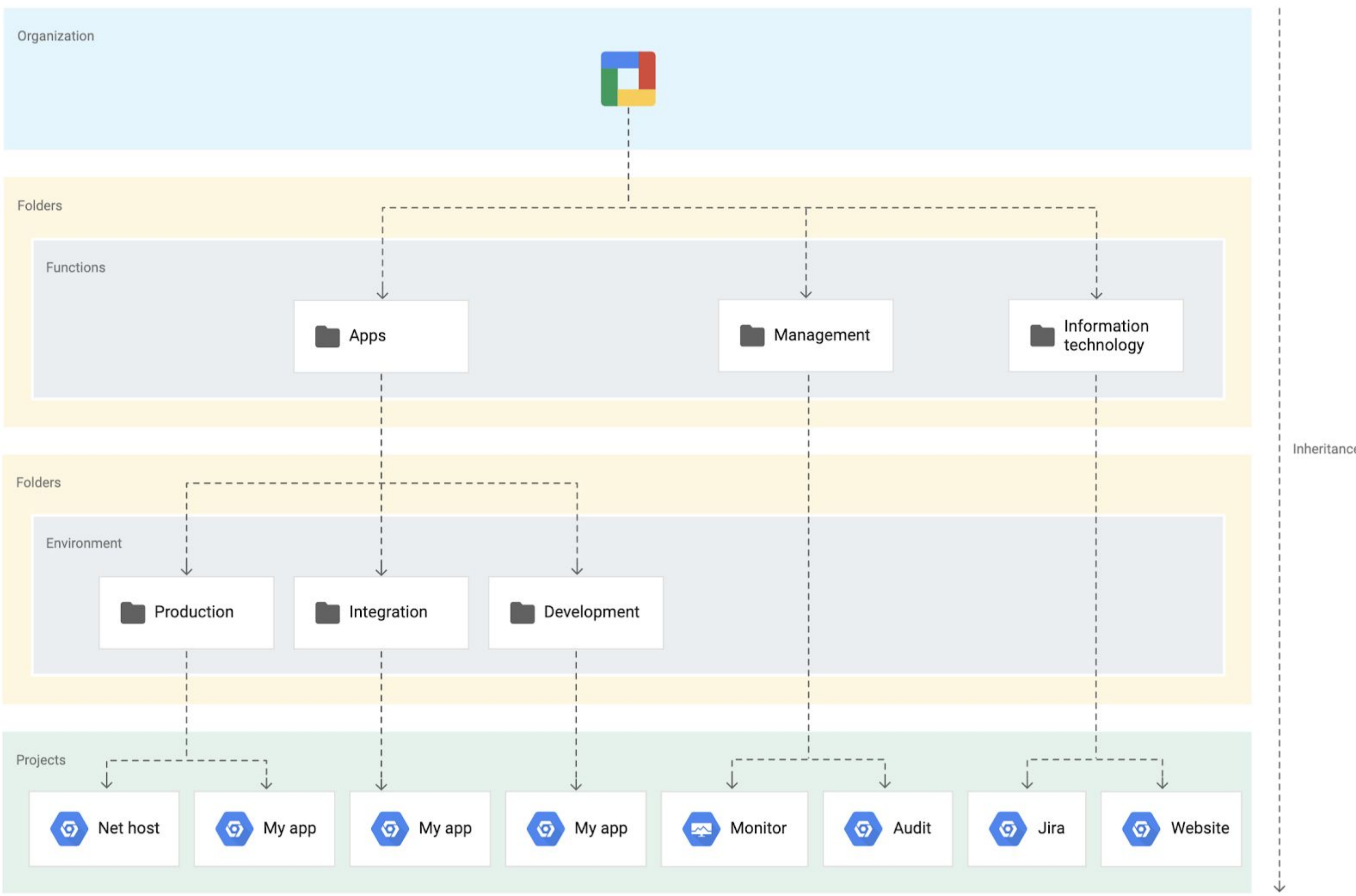
The multiple environments are development, quality assurance, and production. In each environment, there are multiple instances deployed of the same two apps, My app 1 and My app 2.

This hierarchy is simple to implement because it has only three levels, but it can pose challenges if you have to deploy services that are shared by multiple environments.

Function-oriented hierarch

In a function-oriented hierarchy, you have one organization that contains one folder per function, such as information technology and management. Each function folder can contain multiple environment folders.

The following diagram shows an example of a function-oriented hierarchy.



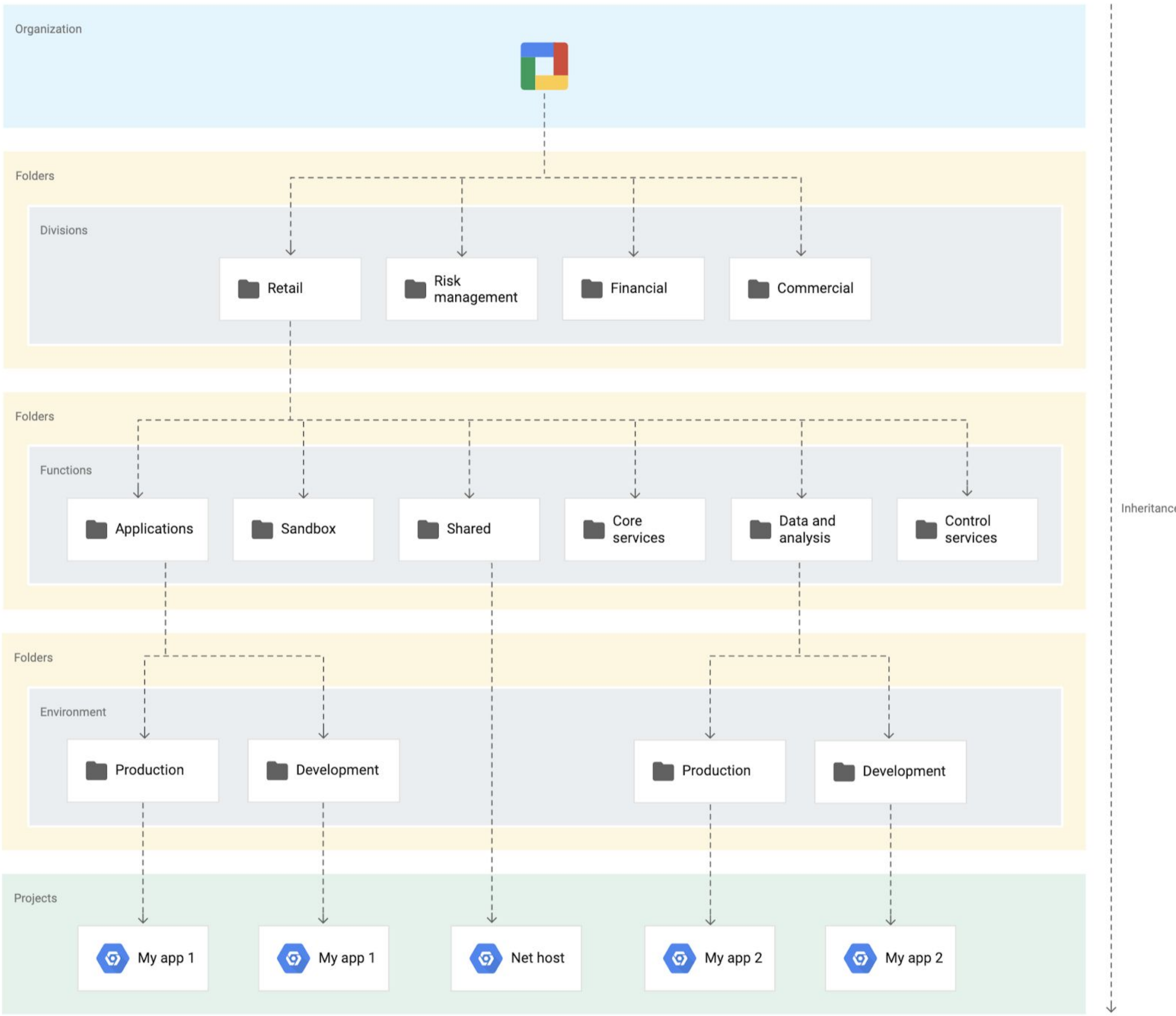
In this hierarchy, the multiple functions are apps, management, and information technology. You can deploy multiple instances of My app, plus shared services, such as Jira and website.

This option is more flexible compared to environment-oriented hierarchies because it gives you the same environment separation, plus it allows you to deploy shared services. On the other hand, a function-oriented hierarchy is more complex to manage than an environment-oriented one, and it doesn't separate access by business unit, such as retail or financial.

Granular access-oriented hierarchy

In a granular access-oriented hierarchy, you have one organization that contains one folder per research unit. Each research unit folder can contain one folder per function. Each function folder can contain one folder per environment.

The following diagram shows an example of a granular access-oriented hierarchy.



In this hierarchy, there are multiple research units, multiple research functions, and environments. You can deploy multiple instances of the My app 1 and My app 2 apps and a shared service, Net host.

This hierarchy is the most flexible and extensible option. On the other hand, you need to spend a greater effort to manage the structure, roles, and permissions. The network topology can also be significantly more complex because the number of projects is higher compared to the other options.

Define groups and roles for resource access

You need to set up the groups and roles to grant the necessary access to resources. In Google Cloud, you can delegate admin access to resources in your organization. At minimum, you need the following roles:

- An organization admin, who defines IAM policies and the hierarchy of the organization and its resources.
- A network admin, who creates and configures networks, subnetworks, and network devices, such as Cloud Router, Cloud VPN and Cloud Load Balancing. An additional responsibility is to maintain firewall rules in collaboration with the security admin.
- A security admin, who establishes policies and constraints for the organization and its resources, configures new IAM roles for projects, and maintains visibility on logs and resources.
- A billing admin, who configures billing accounts and monitors resource usage and spending across the whole organization.

Note: The admin roles grant full rights over the resources in your organization. Grant them only to trusted users, and consider Multi-factor authentication to protect such identities.

Design network topology and establish connectivity

The last step of the plan phase is to set up the network topology and connectivity from your existing environment to Google Cloud.

After creating your projects and establishing identities, you should create at least one Virtual Private Cloud (VPC) network. VPCs let you have a private global addressing space, spanning multiple regions. Inter-regional communication doesn't use the public internet. You can create VPCs to segregate parts of your apps, or have a shared VPC spanning multiple projects. After setting up VPCs, you should also configure network flow logging and firewall rules logging by using Cloud Logging. For more information about VPCs and how to set them up, see Best practices and reference architectures for VPC design.

Google Cloud offers many hybrid connectivity options to connect your existing environment to your Google Cloud projects:

- Public internet
- Cloud VPN
- Peering
- Cloud Interconnect

Connecting through the public internet is a simple and inexpensive connection option because it's backed by a resilient infrastructure that uses Google's existing edge network. On the other hand, this infrastructure isn't private or dedicated. The security of this option depends on the apps that exchange data on each connection. For this reason, we don't recommend using this type of connection to send unencrypted traffic.

Cloud VPN extends your existing network to Google Cloud by using an IPSec tunnel. Traffic is encrypted and travels between the two networks over the public internet. While Cloud VPN requires additional configuration and can impact the throughput of your connection, it is often the best choice if you don't encrypt traffic at the app level and if you need to access private Google Cloud resources.

Peering lets you establish a connection to Google's network over a private channel. There are two peering types:

- Direct peering lets you establish a direct peering connection between your network and Google's edge network. If you don't need to access private resources on Google Cloud and if you meet Google's peering requirements, this is a good option. It doesn't have any Service Level Agreements (SLA), but this option lets you cut your egress fees over public internet access of Cloud VPN.
- Carrier peering lets you connect to Google's network by using enterprise-grade network services managed by a service provider. Although Google doesn't offer any SLA on this connectivity option, it might be covered by a service provider's SLA. When evaluating the pricing of this option, you should consider both Google Cloud egress fees and service provider fees.

Cloud Interconnect extends your existing network to Google's network through a highly available connection. It doesn't provide any encrypted channel by default, so if you want to use this option, we recommend that you encrypt sensitive traffic at the app level. You can choose between two Cloud Interconnect options:

- Dedicated Interconnect gives you high bandwidth private connections with a minimum of 10 Gbps, but requires routing equipment in a colocation facility. In other words, you have to meet Google at one of the points of presence (PoPs). Google provides an end-to-end SLA for Dedicated Interconnect connections, and you're charged based on the dedicated bandwidth and the number of attachments.
- Partner Interconnect lets you use dedicated high-bandwidth private connections managed by a service provider, without requiring you to configure routing equipment in a Google colocation facility. Google provides an SLA for the connection between Google and the service provider. The service provider might offer an SLA for the connection between you and them. Partner Interconnect is charged based on the connection capacity and the amount of egress traffic through an interconnect. Additionally, you might be charged by the service provider for their service.

Migration phase 3: deploy

After building a foundation for your Google Cloud environment, you can begin to deploy your workloads. You can implement a deployment process and refine it during the migration. You might need to revisit the foundation of your environment as you progress with the migration. New needs can arise as you become more proficient with the new cloud environment, platforms, services, and tools.

When designing the deployment process for your workloads, you should take into account how much automation and flexibility you need. There are multiple deployment process types for you to choose, ranging from a fully manual process to a streamlined, fully automated one.

Fully manual deployments

A fully manual provisioning, configuration, and deployment lets you quickly experiment with the platform and the tools, but it's also error prone, often not documented, and not repeatable. For these reasons, we recommend that you avoid a fully manual deployment unless you have no other option. For example, you can manually create resources using the Cloud Console such as a Compute Engine instance and manually run the commands to deploy your workload.

Configuration management tools

A configuration management (CM) tool lets you configure an environment in an automated, repeatable, and controlled way. You can use a CM tool to configure the environment and to deploy your workloads. While this is a better process compared to a fully manual deployment, it typically lacks the features to implement an elaborate deployment, like a deployment with no downtime or a blue-green deployment. Some CM tools let you implement your own deployment logic and can be used to mimic those missing features. However, using a CM tool as a deployment tool can add complexity to your deploy process, and can be more difficult to manage and maintain than a dedicated deployment toolchain. Designing, building, and maintaining a customized deployment solution can be a large additional burden for your operations team.

Container orchestration

If you have already invested in containerization, you can go a step further and use a service such as Google Kubernetes Engine (GKE) to orchestrate your workloads. By using Kubernetes to orchestrate your containers, you don't have to worry about the underlying infrastructure and the deployment logic.

Deployment automation

By implementing an automated artifact production and deployment process, such as a continuous integration and continuous delivery (CI/CD) pipeline, you can automate the creation and deployment of artifacts. You can fully automate this process, and you can even insert manual approval steps, if needed.

For an example implementation of this process, see Continuous Delivery Pipelines with Spinnaker and Google Kubernetes Engine.

Infrastructure as code

While you can automate the deployment process by implementing a CI/CD pipeline, you can adopt a similar process for your infrastructure. By defining your infrastructure as code, you can automatically provision all the necessary resources to run your workloads. With this type of process, you make your infrastructure more observable and repeatable. You could also apply a test-driven development approach to your infrastructure. On the other hand, you need to invest time and effort to implement an infrastructure as code process, so take this into account when planning your migration.

Cloud Deployment Manager is a tool that can help implement your infrastructure as code processes on Google Cloud.

Migrating Data (Overview)

For more detail, see Migration to Google Cloud: Transferring your large datasets

In most enterprise environments, the transfer process involves many factors including the following:

- Devising a transfer plan that accounts for administrative time, including time to decide on a transfer option, get approvals, and deal with unanticipated issues.
- Coordinating people in your organization, such as the team that executes the transfer, personnel who approve the tools and architecture, and stakeholders who are concerned with the value and disruptions that moving data can bring.
- Choosing the right transfer tool based on your resources, cost, time, and other project considerations.
- Overcoming data transfer challenges, including "speed of light" issues (insufficient bandwidth), moving datasets that are in active use, protecting and monitoring the data while it's in flight, and ensuring the data is transferred successfully.

Planning a transfer typically requires personnel with the following roles and responsibilities:

- **Enabling resources needed for a transfer:** Storage, IT, and network admins, an executive sponsor, and other advisors (for example, a Google Account team or integration partners)

- **Approving the transfer decision:** Data owners or governors (for internal policies on who is allowed to transfer what data), legal advisors (for data-related regulations), and a security admin (for internal policies on how data access is protected)
- **Executing the transfer:** A team lead, a project manager (for executing and tracking the project), an engineering team, and on-site receiving and shipping (to receive appliance hardware)

It's crucial to identify who owns the preceding responsibilities for your transfer project and to include them in planning and decision meetings when appropriate. Poor organizational planning is often the cause of failed transfer initiatives.

Gathering project requirements and input from these stakeholders can be challenging, but making a plan and establishing clear roles and responsibilities pays off. You can't be expected to know all the details of your data. Assembling a team gives you greater insight into the needs of the organization. It's a best practice to identify potential issues before you invest time, money, and resources to complete the transfers.

When you design a transfer plan, we recommend that you first collect requirements for your data transfer and then decide on a transfer option. To collect requirements, you can use the following process:

1. Identify what datasets you need to move.
2. Identify what datasets you *can* move.
3. For datasets that are movable, determine where to transfer each dataset.
4. For datasets that are movable, determine what resources are available to move them.
    - Time: When does the transfer need to be completed?
    - Cost: What is the budget available for the team and transfer costs?
    - People: Who is available to execute the transfer?
    - Bandwidth (for online transfers): How much of your currently available bandwidth for Google Cloud can be allocated for a transfer, and for what period of time?

Before you evaluate and select transfer options in the next phase of planning, we recommend that you assess whether any part of your IT model can be improved, such as data governance, organization, and security.

Migration phase 4: Optimize

After deploying your workloads, you can start optimizing your target environment. In this optimization phase, the following cross-area activities can help you optimize this environment:

- Build and train your team.
- Monitor everything.
- Automate everything.
- Codify everything.
- Use managed services instead of self-managed ones.
- Optimize for performance and scalability.
- Reduce costs.

Build and train your team

When you plan your migration, you can train your development and operation teams to take full advantage of the new cloud environment. Not only can those teams be more efficient with effective training, but they can also choose the best cloud-native tools and services for the job. Training opportunities help to retain technical talent and empower engineers to leverage all of the advantages of Google Cloud.

During this phase, you can also review the business processes that govern those teams. If you find any inefficiency or unnecessary burden in those processes, you have the possibility to refine them and improve them with training.

Monitor everything

Monitoring is the key to ensure that everything in your environment is working as expected, and to improve your environments, practices, and processes.

Before you expose your environment to production traffic, we recommend that you design and implement a monitoring system where you define metrics that are important to assess the correct operation of the environment and its components, including your workloads. For example, if you are deploying a containerized infrastructure, you can implement a white-box monitoring system with Prometheus. Or, you can monitor your IoT Core devices with Cloud Logging and Cloud Functions.

We also recommend that you set an alerting system like Cloud Monitoring alerting, that lets you be proactive, not just reactive. You need to set up alerts for critical errors and conditions, but you also need to set up warnings that give you time to correct a potentially disruptive situation before it affects your users.

You can then export Cloud Monitoring metrics logs for long-term storage because Cloud Logging logs have a limited retention period, or run data analytics against the metrics extracted from such logs to gain insights on how your environment is performing and start planning improvements.

Automate everything

Manual operations are exposed to a high error risk and are also time consuming. In most cases, you can automate critical activities such as deployments, secrets exchanges, and configuration updates. Automation leads to cost and time savings, and reduces risk. Teams also become more efficient, because they don't have to spend effort on repetitive tasks. Automating infrastructure with Cloud Composer and Automating Canary Analysis on Google Kubernetes Engine with Spinnaker are examples of automation on Google Cloud.

Codify everything

When provisioning the target environment on Google Cloud, you should aim to capture as many aspects as you can in code. By implementing processes such as infrastructure-as-code and policy-as-code, you can make your environment fully auditable and repeatable. You can also apply a test-driven development approach to aspects other than code, to have immediate feedback on the modifications you intend to apply to your environment.

Use managed services instead of self-managed ones

Google Cloud has a portfolio of services and products that you can use without having to manage any underlying servers or infrastructure. In the optimization phase, you could either expand your workloads to use such services, or replace some of your existing workloads with these services.

A few examples of managed services are as follows:

- Using Cloud SQL for MySQL instead of managing your own MySQL cluster.
- Using AutoML to tag and classify images instead of deploying and maintaining your own machine learning models.
- Deploying your workloads on GKE instead of using your own self-managed Kubernetes cluster, or even migrating your VMs to containers and running them on GKE.
- Using App Engine for serverless web hosting.

Optimize for performance and scalability

One of the advantages of migrating to the cloud is access to resources. You can bolster existing resources, add more when you need them, and also remove unneeded resources in a scalable way.

You have more options to optimize performance compared to on-premises deployments:

- **Horizontal scaling.** You can elastically add or remove virtual machines, cluster nodes, and database instances. You can use services such as Compute Engine autoscaling groups, GKE cluster autoscaler.
- **Vertical scaling.** Adding more resources to your existing instances is easier in a cloud environment because you don't have to provision any additional physical infrastructure. For example, you can easily change the machine type of Compute Engine instances.

Reduce costs

Google Cloud offers a wide range of tools and pricing options to help you reduce your costs.

For example, if you provisioned Compute Engine instances, you can apply sizing recommendations for those instances.

To reduce your billing, you can analyze your billing reports to study your spending trends and determine which Google Cloud products you are using most frequently. You can even export your billing data to BigQuery or to a file to analyze.

To further reduce your costs, Google Cloud has features, such as sustained use discounts that apply automatic discounts to your Compute Engine billing. You can also purchase committed use contracts in return for discounted prices for Compute Engine instances. For BigQuery, you can also enroll in flat-rate pricing. Google Cloud autoscaling features also help you reduce your billing by scaling down your resources according to client requests. You can reduce monitoring and logging costs by optimizing your usage of Cloud Monitoring and Cloud Logging.

Appendix

Finding help

Google Cloud offers various options and resources for you to find the necessary help and support to best leverage Google Cloud services.

Self-service resources

If you don't need dedicated support, you can use these self-service resources:

- Documentation. Google Cloud provides documentation for each of its products and services, as well as for APIs. For more information about migrations, check out the Google Cloud migration center.
- Tools. Google Cloud provides several products and services to help you migrate. For example:
    - Migrate for Compute Engine is a product for migrating physical servers and virtual machines from on-premises and cloud environments to Google Cloud. Migrate for Compute Engine lets you migrate a virtual machine to Google Cloud in a few minutes, where the data is copied in the background but the virtual machines are completely operational.
    - Online transfer to Cloud Storage by using the gsutil command-line tool, Cloud Storage JSON API or the Cloud Console.
    - Storage Transfer Service lets you bring data to Cloud Storage from other cloud providers, online resources, or local data.
    - Transfer Appliance is a hardware appliance you can use to migrate large volumes of data (from hundreds of terabytes up to 1 petabyte) to Google Cloud without disrupting operations.
    - BigQuery Data Transfer Service automates data movement from software-as-a-service apps to BigQuery on a scheduled, managed basis.
- Whitepapers. These papers include reference architectures, case studies, best practices, and advanced tutorials.
- Media content. You can listen to the Google Cloud podcast or watch any of the videos on the Google Cloud YouTube channel. These resources discuss a wide range of topics from product explanations to development strategies.
- Online courses and hands-on labs. Google Cloud has several courses on Coursera that include video content, reading materials, and hands-on labs. You can also take hands-on labs using Qwiklabs or participate in live online classes.

Technology partners

Google Cloud has partnered with multiple companies to enable you to use their products. Some of the offerings might be free to use so ask the company and your Google Cloud account manager.

These products include the following:

- **Assessment and discovery phase:** StratoZone, CloudPhysics, Risc Networks, and Cloudamize
- **Provisioning and configuration phases:** Terraform, Chef, Ansible, SaltStack, and Puppet

System integrators

Google Cloud partners not just with product and technology companies, but with system integrators that can provide hands-on-keyboard assistance. In the partners list, you can find a list of system integrators that specialize in cloud migrations.

Google Cloud Professional Services

Our Professional Services team is here to help you get the most out of your investment in Google Cloud.

Google's Professional Services can help you plan your migration and deploy your workloads in production with our Cloud Plan and Foundations offering. These experts provide your team with guidance through each phase of migrating your workload into production, from setting up Google Cloud foundations to optimize the platform for your unique workload needs and deploying the workload.

The objectives of Cloud Plan and Foundations are:

- Set up the Google Cloud foundation.
- Create design documentation.
- Plan deployment and migration activities.
- Deploy workloads into production.
- Track issues and risks.

Professional Services guides your team through the following activities and deliverables:

1. Conducting technical kickoff workshops.
2. Building a technical design document.
3. Creating a migration plan.
4. Creating a program charter.
5. Providing project management.

6. Providing technical expertise.

Cloud Sprint: accelerate your migration to Google Cloud

Cloud Sprint is an intensive hands-on workshop that accelerates your app migration to Google Cloud. In this workshop, Google Cloud Professional Services leads one of your teams through interactive discussions, whiteboarding sessions, and reviewing target apps to migrate to Google Cloud. During the Cloud Sprint, Professional Services works alongside your team members to help you gain first-hand experience with cloud solutions with required deployment activities to help you understand your next steps for future Google Cloud migrations.

Training: Develop your team's skills

Google Cloud Professional Services can provide training in fields based on your team's needs.

What's next

- Read about migrating a monolithic application to microservices on Google Kubernetes Engine.
- Learn about Anthos and Migrate for Anthos.
- Try out other Google Cloud features for yourself. Have a look at our tutorials.

The journey onto the NIH.gov STRIDES environment begins by creating a formal relationship with the NIH STRIDES organization. The process for creating this relationship is described in the NIH STRIDES Onboarding Guide.

After the formal relationship is established, the technical process for migrating to GCP begins. Google utilizes the Google Cloud Adoption Framework to govern the process of migration. The framework contains four themes:

- **Learn.** The quality and scale of your learning programs.
- **Lead.** The extent to which your IT departments are supported by a mandate from leadership to migrate to Google Cloud.
- **Scale.** The extent to which you use cloud-native services, and how much operational automation you currently have in place.
- **Secure.** The capability to protect your current environment from unauthorized and inappropriate access.

This document will provide a guide for migrating to When planning your migration to Google Cloud, you start by defining the starting and target environments that are involved in the migration. Your starting point can be an on-premises environment, a private hosting environment, or another public cloud environment. In this document, the target environment is Google Cloud. After you define your starting and target environments, you define the workload types and the related operational processes that are in scope for the migration.

Google offers a number of healthcare and life sciences solutions to aid researchers in realizing the potential of data through holistic insights.

Google's Healthcare API can be used to accelerate the ingestion, storage, and integration of key data types, such as FHIR, HL7, and DICOM. The Healthcare API can also be used with the Apigee Healthcare APIx to help organizations easily develop healthcare and patient applications that are securely hosted, easy to use, and capable of analyzing patient data.

Tools such as BigQuery, Cloud Machine Learning Engine and Tensorflow enable researchers to apply analytics and artificial intelligence to data. With these tools, users can establish mechanisms to auto-detect patterns, predict clinical outcomes, and to quickly analyze large amounts of data.

Additionally, both G Suite and GCP are FedRAMP compliant and HITRUST CSF certified; prescriptive security controls are implemented throughout Google Cloud for processing, storing, and transmitting sensitive data. Both G Suite and GCP also support HIPAA compliance across dozens of products in all regions, zones, network paths, and points of presence for Google Cloud.

Learn more about Google Cloud's healthcare solutions and innovative capabilities.

Reference Architectures

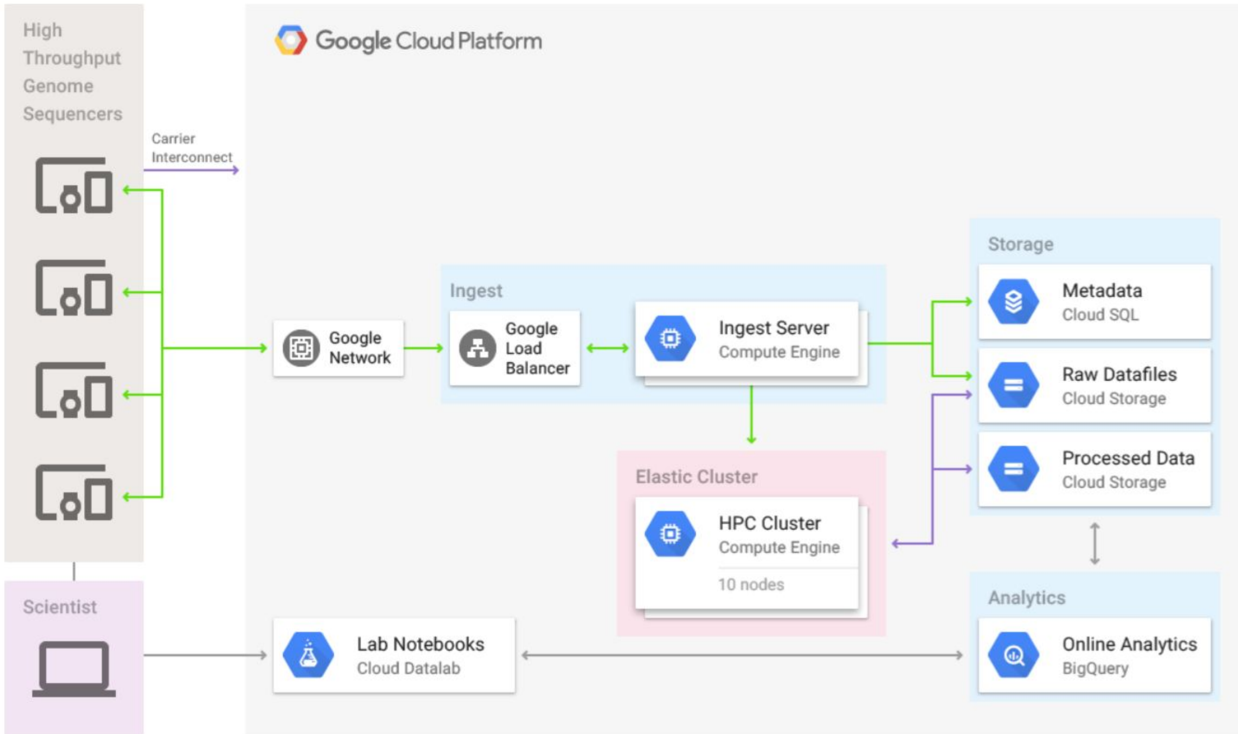
(link)

Google Cloud supports a wide variety of healthcare systems and workflows, from genomics and variant analysis to patient monitoring and machine learning on health records. Here's a few common healthcare use cases, and how to implement those solutions in Google Cloud.

Genomics and Secondary Analysis

(link)

Send sequence data to Google Cloud for processing, analytic analysis, and share results to interactive lab notebooks.



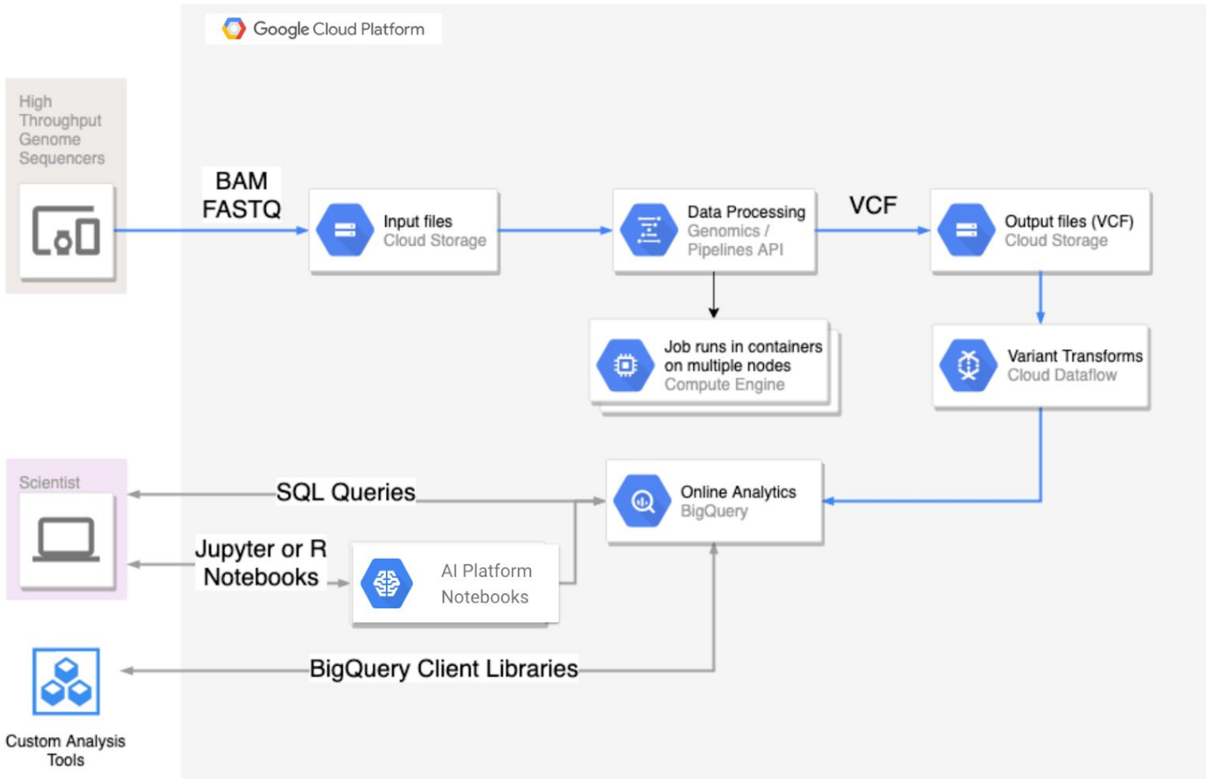
Solution Summary:
Use the cloud to process data from your existing, on-premise sequencers. Establish a high-bandwidth, Interconnect from on-premise to your Google Cloud VPC network. As data is sent from sequencers over Interconnect, it's load balanced to ingest server(s) that subsequently stores the raw data in Google Cloud Storage (GCS) and metadata about the raw data in Cloud SQL. After the raw data is stored in GCS, a job is triggered and submitted to an HPC cluster in the cloud. The HPC cluster processes the raw sequence reads into contigs or aligned BAM files. The processed data is then made accessible to scientists through interactive notebooks, and BigQuery is used to further analyze the sequence data.

Suggested GCP Products and Services:
- Cloud Interconnect - for high-capacity connectivity from on-premise to your GCP network VPC through a Dedicated connection or using a Partner/Carrier connection
- Cloud VPC Network - Virtual Private Cloud provides global, scalable, and flexible networking for your GCP resources and services without compromising integrity on public-facing connections
- Google Load Balancer - To distribute load-based resources while meeting high availability requirements
- Google Compute Engine (GCE) - VM instances for processing data
- Cloud Genomics: Pipelines API - Google's API for running workflows in containers on GCE instances.
- Google Datalab - Google's interactive lab notebook built on Jupyter

- AI Platform Notebooks - A **managed service** that offers an integrated JupyterLab environment; create instances running JupyterLab that come pre-installed with the latest data science and machine learning frameworks.
- Cloud SQL - Fully **managed** relational database to support MySQL or PostgreSQL data storage.
- Cloud Storage - Unified object storage for storing raw data files and processed data. Supports regional, multi-regional, archive, and infrequently accessed data.
- BigQuery - Analytics data warehouse for large-scale datasets.

High-level Setup Steps:

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project
2. In the GCP project, create a VPC network to logically isolate your project resources
3. Provision a Dedicated Interconnect or a Partner Interconnect to extend your on-premise environment to GCP over a high-bandwidth connection
4. In the VPC network, use GCE VMs to create a high performance compute (HPC) cluster for data processing, and spin up ingest servers to receive the data
5. Create a MySQL or PostgreSQL instance that'll store metadata from the raw sequence data that's sent to the ingest servers
6. Create a Cloud Storage bucket that will store the raw sequence data from the ingest servers
7. Configure the load balancer service to distribute incoming sequence data to the ingest servers
8. Use Cloud Functions to create a GCS trigger that kicks off a job on the HPC cluster to process the raw sequence data reads from GCS. The HPC job should process the raw sequence reads into contigs or aligned BAM files
9. Load the GCS data into BigQuery or use BigQuery to query the data directly from GCS, to run further analysis
10. Create Cloud Datalab instances to share the processed data via notebooks, or use AI Platform (recommended) to create notebooks with native Jupyter support.

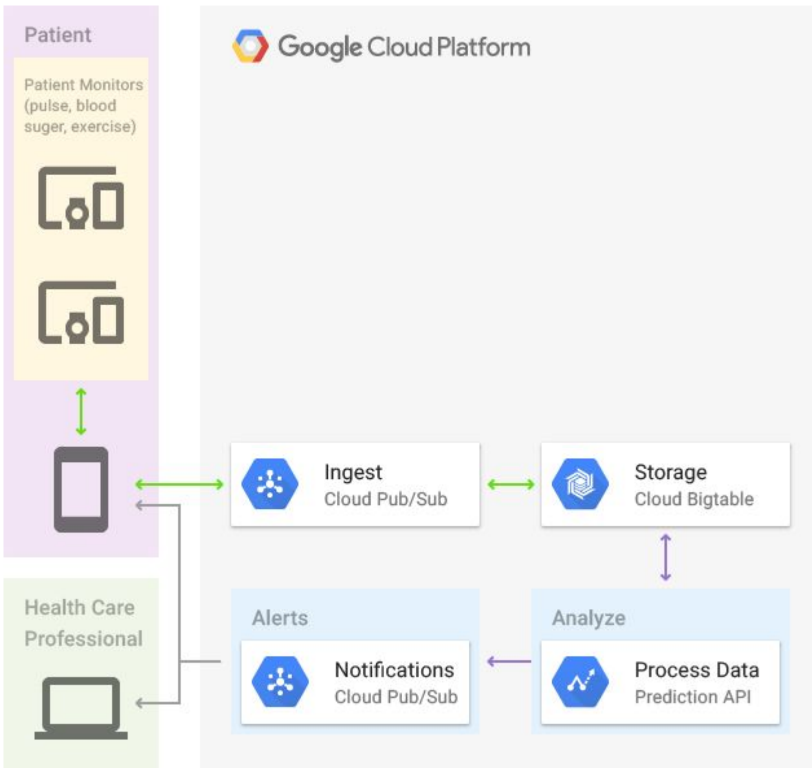


High-level Setup Steps (using Pipelines API):

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project
2. Provision a Dedicated Interconnect or a Partner Interconnect to extend your on-premise environment to GCP over a high-bandwidth connection
3. Create a Cloud Storage bucket that will store the raw sequence data from the ingest servers
4. Configure the load balancer service to distribute incoming sequence data to the ingest servers
5. Package the analytic workflow into containers and define the workflow in a WDL or CWL config file or use an existing workflow such as GATK (tutorial).
6. Use a workflow engine (such as Cromwell or dsub) to schedule and run the analytics workflow using Pipelines API.
7. Use Variant Transforms to load the variant files (VCF) to BigQuery for further analysis.
8. Use BigQuery to analyze the variants.
9. Create Cloud Datalab instances to share the processed data via notebooks, or use AI Platform (recommended) to create notebooks with native Jupyter support.

Patient Monitoring (link)

Securely send patient data to Google Cloud for processing, analytic analysis, and collaboration on results.



Solution Summary:
Information is ingested from patient monitoring devices (eg. blood sugar readings, heart rate, etc.) into connected cloud services for storage and analysis. As data is stored, advanced analytics using the prediction API or Tensorflow, are updated to determine patient risk. Notifications are triggered based on predetermined criteria, and can be sent to the patient, additional health care professionals, and researchers for further collaboration.

Suggested GCP Products and Services:

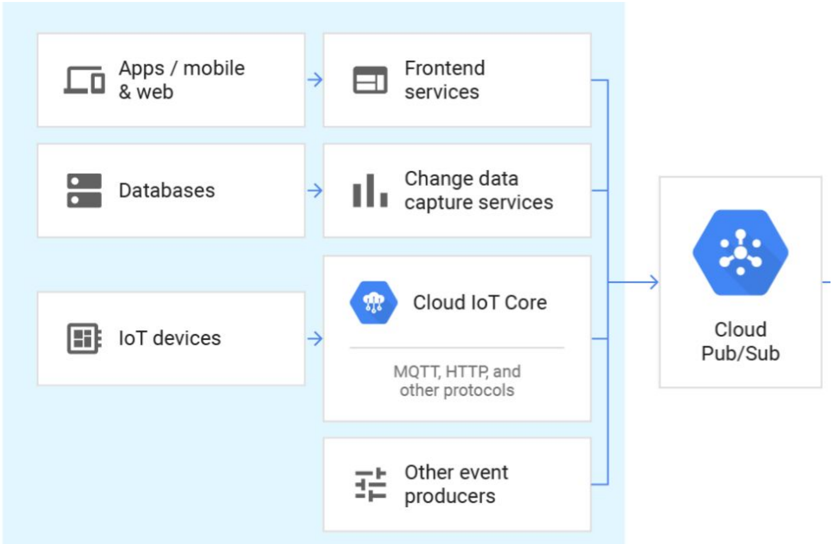
- Cloud VPC Network - Virtual Private Cloud provides global, scalable, and flexible networking for your GCP resources and services without compromising integrity on public-facing connections
- Cloud IoT Core - A fully managed service that provides a complete solution for collecting, processing, analyzing, and visualizing IoT (Internet of Things) data in real time
- Cloud Pub/Sub - a simple and reliable staging location for pushing (pub) and pulling (sub) event data on its journey towards processing, storage, and analysis

- Cloud Bigtable - A large-scale, fully managed NoSQL database service for operational workloads
- Cloud Machine Learning Engine (now part of the Google AI Platform) - The successor to Google Prediction API, MLE is a managed service that allows developers and data scientists to build and run production-level machine learning models with the purpose of optimizing data analysis and data flow.

High-level Setup Steps:

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project.
2. In the GCP project, create a VPC network to logically isolate your project resources.
3. Build out a Cloud Pub/Sub system to ingest events & messages data from patient monitoring devices.
4. (Optional) Use Cloud IoT Core to create registries of IoT devices, making it easy to securely connect, manage, and ingest data from millions of globally dispersed devices. Device data captured by Cloud IoT Core gets published to Cloud Pub/Sub for downstream analytics.
5. Create a Google Cloud Storage bucket that will store event and message notifications exported from Pub/Sub.
6. In AI Platform, create an ML training job against your patient device data stored in GCS, then create predictions from the trained models you developed.
7. Cloud Pub/Sub integrates with various Cloud APIs, including machine learning, big data, and storage APIs. Use this integration to send notifications to patients, other healthcare professionals, researchers, and end devices.

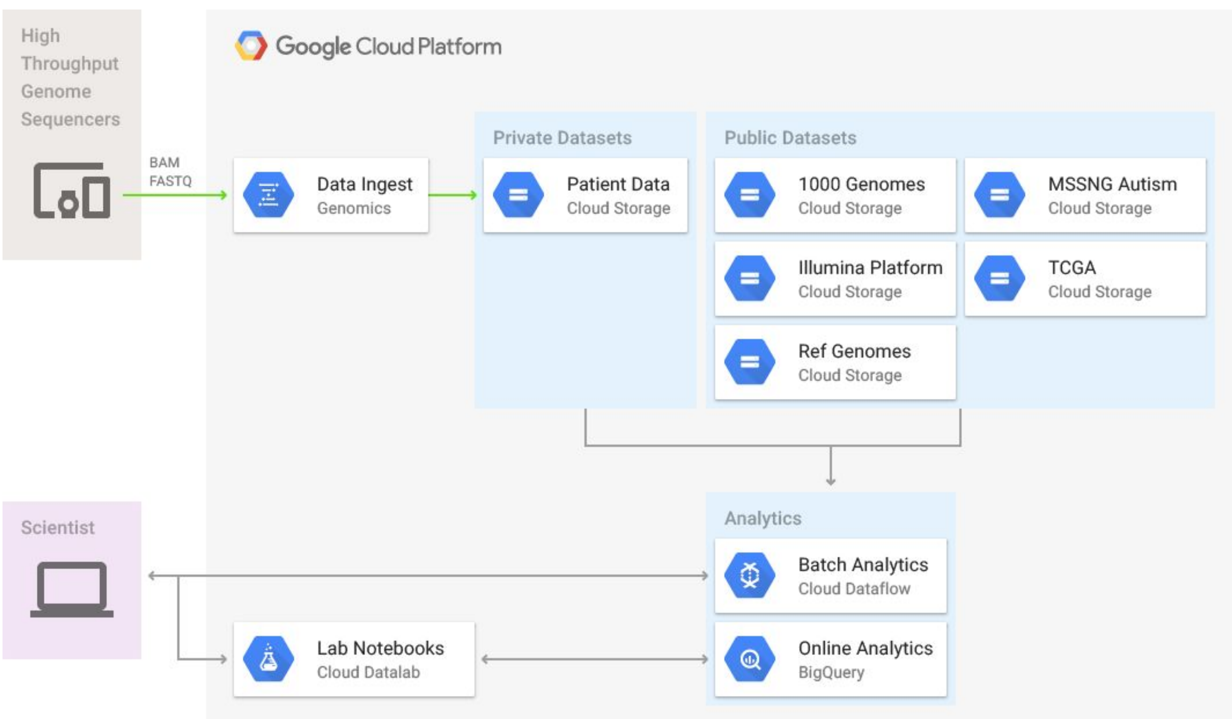
Optional Cloud IoT setup:



Variant Analysis

(link)

Efficiently process and analyze large genomic datasets using Google Genomics. Run experiments in parallel to get results sooner, sharing results and insights with collaborators and broader communities.



Solution Summary:

Cloud Genomics offers petabyte scale genomic data processing and analysis to meet the needs of bioinformatic researchers. Genomics API leverages Google's unique big data technology to store and manage large volumes of sequence data in FASTQ or BAM format. Using the API and command line tools, users can upload files into private and shared repositories.

In addition to private datasets, users can access a large corpus of public datasets available natively in GCS. Both batch analytics using Cloud Dataflow, or interactive analytics using BigQuery and DataLab are supported.

Suggested GCP Products and Services:

- Cloud VPC Network - Virtual Private Cloud provides global, scalable, and flexible networking for your GCP resources and services without compromising integrity on public-facing connections
- Google Genomics - Ask bigger questions by efficiently processing petabytes of genomic data
- BigQuery - Analytics data warehouse for large-scale datasets
- Cloud Storage - Unified object storage for developers and enterprises
- Cloud Datalab - An easy-to-use interactive tool for data exploration, analysis, visualization, and machine learning.
- AI Platform Notebooks - A **managed service** that offers an integrated JupyterLab environment; create instances running JupyterLab that come pre-installed with the latest data science and machine learning frameworks.
- Cloud Dataflow - Simplified stream and batch data processing, with equal reliability and expressiveness

High-level Setup Steps:

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project.
2. In the GCP project, create a VPC network to logically isolate your project resources.
3. Create Google Cloud Storage buckets to store public and private datasets in the cloud; select storage classes based on availability requirements and bucket locations that are close to users. Use IAM to control bucket access policies. Implement additional encryption for sensitive data, and consider a requester pays model for qualified datasets.
    a. Review strategies for transferring big datasets to GCP, and Google's various data transfer options including Online Transfer, GCS Transfer, Transfer Appliance, and BigQuery Transfer Service.
    b. Use Cloud Genomics Public Datasets, which have already been loaded into Cloud Genomics BigQuery and GCS buckets

4. Use Cloud Genomics to create an ingest pipeline of sequence data & metadata from BAM and FASTQ files. Store and load genomic variants in Google Cloud Storage buckets.
    a. Use Cloud Genomics Public Datasets, which have already been loaded into Cloud Genomics BigQuery and GCS buckets
5. Create dataset(s) in BigQuery, which will store the pipeline data for further analysis. Create BigQuery jobs to load, export, query, or copy data from GCS on your behalf.
6. Create an Apache Beam pipeline and use Cloud Dataflow to run the batch analytics pipeline, which will analyze data stored in GCS using BigQuery. Use Dataflow to read data from and write data to BigQuery.
7. Use Cloud Datalab or AI Platform (recommended) to create notebooks for visualizing BigQuery data. AI Platform is recommended for native Jupyter notebook integration.
8. Use notebooks and Dataflow to share analytics data, visualizations, and insights from BigQuery with additional teams and researchers.

Healthcare API for Machine Learning and Analytics

Solution Summary

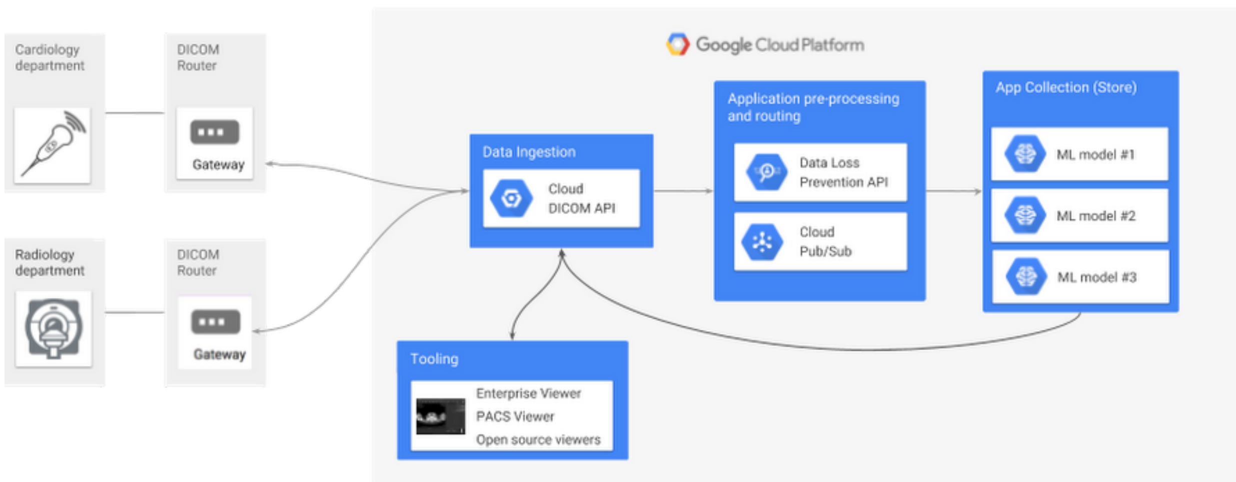
Use the Healthcare API to easily integrate machine learning's powerful analytical insights on top of new and existing medical data. With a focus on healthcare applications, the Google Healthcare API gives developers an efficient tool set to transform propositions into solutions at work. Google's machine learning has precise recognition that can be trained with high accuracy to serve an array of predictive functions.

Suggested GCP Products and Services:

- Cloud VPC Network - Virtual Private Cloud provides global, scalable, and flexible networking for your GCP resources and services without compromising integrity on public-facing connections.
- Cloud Healthcare API - Standards-based APIs powering actionable healthcare insights for security and compliance-focused environments.
- Cloud Pub/Sub - a simple and reliable staging location for pushing (pub) and pulling (sub) event data on its journey towards processing, storage, and analysis.
- Cloud Storage - Unified object storage for storing raw data files and processed data. Supports regional, multi-regional, archive, and infrequently accessed data.
- Cloud Dataflow - Simplified stream and batch data processing, with equal reliability and expressiveness.
- Cloud Dataproc - A fast, easy-to-use, fully managed cloud service for running Apache Spark and Apache Hadoop clusters in a simpler, more cost-efficient way.
- BigQuery - Analytics data warehouse for large-scale datasets.
- Cloud Datalab - An easy-to-use interactive tool for data exploration, analysis, visualization, and machine learning.

- Data Loss Prevention API - Understand and manage sensitive data. It provides fast, scalable classification and redaction for sensitive data elements like credit card numbers, names, social security numbers, US and selected international identifier numbers, phone numbers, and GCP credentials.
- Cloud AutoML - Suite of machine learning products that enables developers with limited machine learning expertise to train high-quality models specific to their business needs.

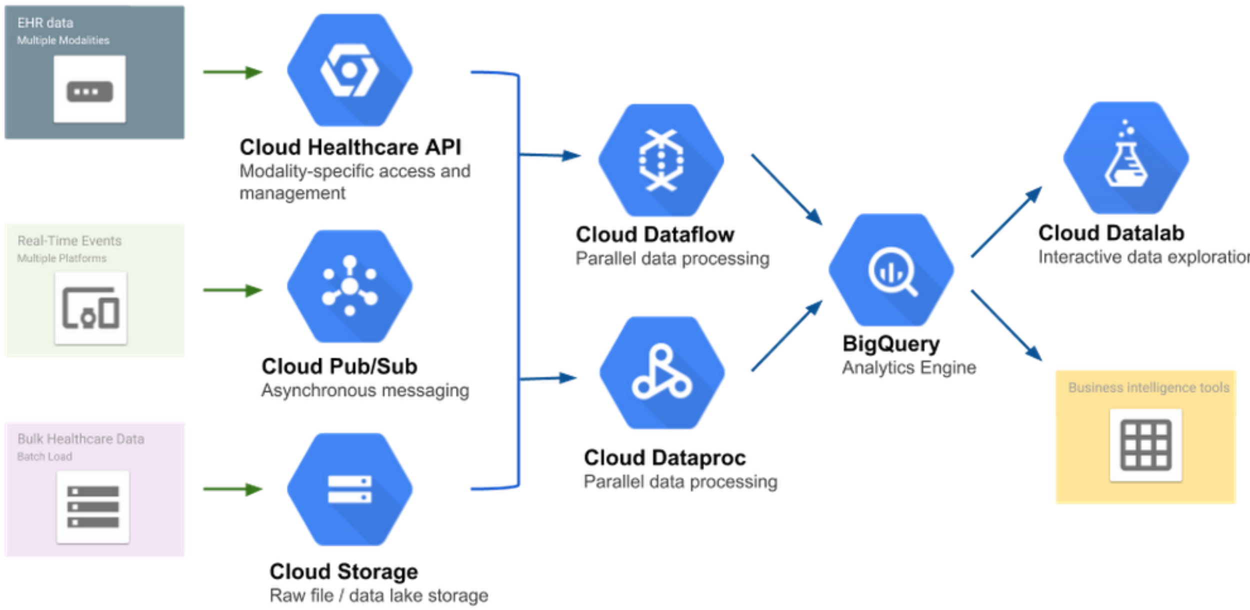
Machine Learning Reference (link)



High-level Setup Steps:

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project.
2. In the GCP project, create a VPC network to logically isolate your project resources.
3. Create service account(s) that will enable you to authenticate to the Healthcare API.
4. To use the Cloud Healthcare API for ingesting, storing, analyzing, and integrating healthcare data with cloud-based applications, create dataset(s) to support the healthcare data models you want to analyze.
    a. Overview of Healthcare API datasets and data stores.
    b. Enable dataset de-identification to remove PII or obscure sensitive information
5. Create DICOM, HL7, and/or FHIR stores that will house ingested medical imaging, clinical event messaging, and clinical resource data.
6. Cloud Pub/Sub integrates with various Cloud APIs, including machine learning, big data, and storage APIs. Use this integration to ingest notifications and messages from Healthcare API and send them to CloudAutoML for things like natural language processing, ML modeling, translation, video intelligence, and vision classification.

Analytics Reference [(link)]



High-level Setup Steps:

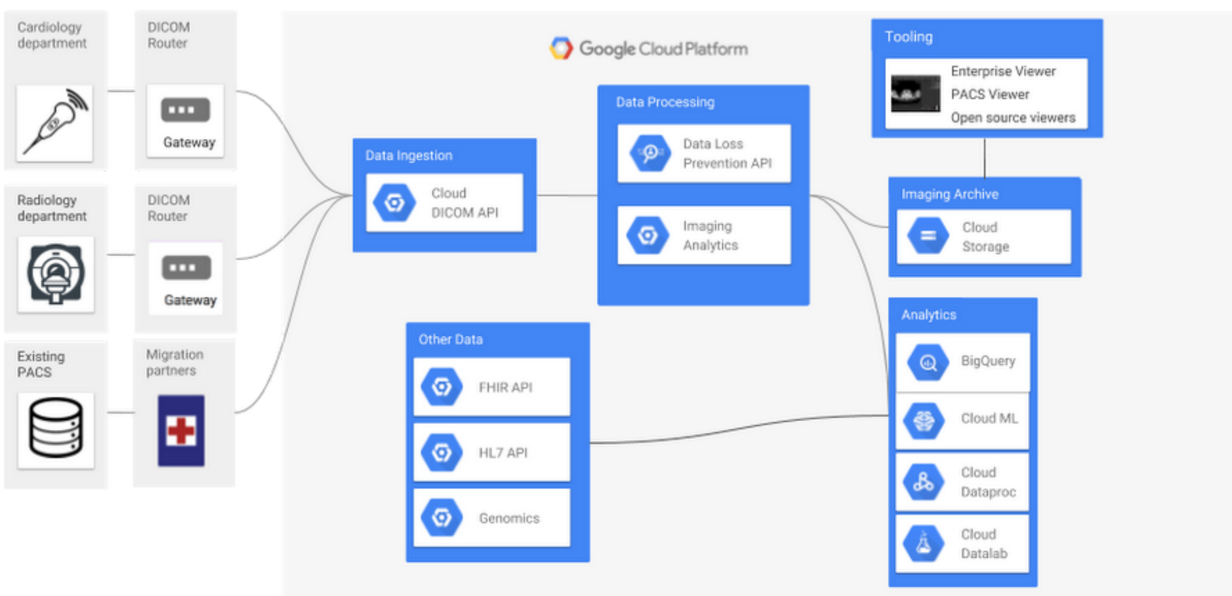
1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project.
2. In the GCP project, create a VPC network to logically isolate your project resources.
3. Create service account(s) that will enable you to authenticate to the Healthcare API.
4. To use the Cloud Healthcare API for ingesting, storing, analyzing, and integrating healthcare data with cloud-based applications, create dataset(s) to support the healthcare data models you want to analyze.
    a. Overview of Healthcare API datasets and data stores.
    b. Enable dataset de-identification to remove PII or obscure sensitive information
5. Create DICOM, HL7, and/or FHIR stores that will house ingested medical imaging, clinical event messaging, and clinical resource data.
6. Create Google Cloud Storage buckets to store raw electronic health records and medical files. Select storage classes based on availability requirements and bucket locations that are close to users. Use IAM to control bucket access policies. Implement additional encryption for sensitive data, and consider a requester pays model for qualified datasets.
    a. Review strategies for transferring big datasets to GCP, and Google's various data transfer options including Online Transfer, GCS Transfer, Transfer Appliance, and BigQuery Transfer Service.
    b. Create a Google Cloud Storage bucket that will store event and message notifications exported from Pub/Sub.

7. Cloud Pub/Sub integrates with various Cloud APIs, including machine learning, big data, and storage APIs. Use Pub/Sub to great an ingestion pipeline of messaging streams for Dataproc or Dataflow
8. Create dataset(s) in BigQuery, which will store the pipeline data for further analysis. Create BigQuery jobs to load, export, query, or copy data from GCS on your behalf.
9. Create an Apache Beam pipeline and use Cloud Dataflow to run the batch analytics pipeline, which will analyze data stored in GCS using BigQuery. Use Dataflow to read data from and write data to BigQuery.
10. Create a Dataproc cluster and get the BigQuery Connector for Dataproc, to submit a Dataproc job that further processes the ingested medical data.
11. Use Cloud Datalab or AI Platform (recommended) to create notebooks for visualizing BigQuery data. AI Platform is recommended for native Jupyter notebook integration.
12. Use notebooks and Dataflow to share analytics data, visualizations, and insights from BigQuery with additional teams and researchers.

Radiological Image Extraction

(link)

Use Google Cloud to gain insights from devices and equipment. Ingest common medical data formats for interoperable processing and analytics.



Solution Summary:

Google Cloud is home to a collection of products and services that can be customized to fit unique customer needs. In the case of Radiological Image Extraction, healthcare professionals and their associates can connect to the Google Cloud Platform to upload, analyze and share findings in near-real time.

With trainable machine learning, GCP becomes a powerful tool in analyzing large datasets of images to produce meaningful results. On a further step, one can use Google's Healthcare API's to establish unique systems to easily parse discrete information from analytic results and compare this current data against historical data.

Suggested GCP Products and Services:

- Cloud VPC Network - Virtual Private Cloud provides global, scalable, and flexible networking for your GCP resources and services without compromising integrity on public-facing connections.
- Cloud Healthcare API (supports/replaced DICOM, FHIR, HL7 APIs) - Supports existing healthcare data standards, formats, and protocols to bridge the gap between traditional on-premise care applications & systems, enabling them in the cloud.
    - DICOM - Digital Imaging and Communications in Medicine (DICOM) is an international standard used for medical images such as X-rays, MRIs, ultrasounds, and other medical imaging modalities. The Cloud Healthcare API provides operations for reading DICOM instances, studies, and series that are consistent with the DICOMweb standard, and supports the DICOM DIMSE C-STORE operation using an open-source adapter.
    - FHIR - Fast Healthcare Interoperability Resources (FHIR) is a healthcare standard for representing and exchanging electronic medical information.
    - HL7v2 - Health Level Seven International Version 2 (HL7v2) is a clinical messaging format that provides data about events that occur inside an organization.
- Data Loss Prevention API - Understand and manage sensitive data. It provides fast, scalable classification and redaction for sensitive data elements like credit card numbers, names, social security numbers, US and selected international identifier numbers, phone numbers, and GCP credentials.
- AI Platform - Collection of machine learning and data analytics tools for developers, data scientists, and data engineers.
    - Cloud Machine Learning Engine - The successor to Google Prediction API, MLE is a managed service that allows developers and data scientists to build and run production-level machine learning models with the purpose of optimizing data analysis and data flow.
- BigQuery - Analytics data warehouse for large-scale datasets.
- BigQuery ML (use instead of Cloud ML) - Enables users to create and execute machine learning models in BigQuery using standard SQL queries.
- Cloud Storage - Unified object storage for developers and enterprises.
- Cloud Dataproc - A fast, easy-to-use, fully managed cloud service for running Apache Spark and Apache Hadoop clusters in a simpler, more cost-efficient way.
- Cloud Datalab - An easy-to-use interactive tool for data exploration, analysis, visualization, and machine learning.

High-level Setup Steps:

1. Create a Google Cloud Project and use Cloud IAM to manage who has access to the resources within the GCP project.
2. In the GCP project, create a VPC network to logically isolate your project resources.
3. Create service account(s) that will enable you to authenticate to the Healthcare API.
4. To use the Cloud Healthcare API for ingesting, storing, analyzing, and integrating healthcare data with cloud-based applications, create dataset(s) to support the healthcare data models you want to analyze.
    a. Overview of Healthcare API datasets and data stores.
    b. Enable dataset de-identification to remove PII or obscure sensitive information
5. Create DICOM, HL7, and/or FHIR stores that will house ingested medical imaging, clinical event messaging, and clinical resource data.
6. Create Google Cloud Storage buckets to store imaging and clinical data archives in the cloud. Select storage classes based on availability requirements and bucket locations that are close to users. Use IAM to control bucket access policies. Implement additional encryption for sensitive data, and consider a requester pays model for qualified datasets.
7. Create dataset(s) in BigQuery, which will store the analyzed data.
8. (Optional) Create a Dataproc cluster and get the BigQuery Connector for Dataproc, to submit a Dataproc job that further processes the ingested medical data.
9. Using the Healthcare API, ingest and manage medical imaging, clinical events and clinical messaging data. Configure DICOM routers and/or use GCP's data transfer options to migrate medical data to the cloud.
10. Export DICOM, HL7, and/or FHIR data to GCS for backup or archive storage. Additionally, send DICOM, HL7, and/or FHIR data to BigQuery for analysis.
11. (Optional) Use BigQuery ML to create and train ML models based on the medical data collected and analyzed in BigQuery
12. Use Cloud Datalab or AI Platform (recommended) to create notebooks for visualizing BigQuery data; AI Platform is recommended for native Jupyter notebook integration.

Resource Links

Healthcare Solutions Reference Architectures (link)

Reference Architecture for Creating a HIPAA-Aligned Project in GCP (link)

Importing FHIR clinical data into the cloud using the Cloud Healthcare API (link)

Getting to know the Google Cloud Healthcare API - Part 1 (link)

Getting to know the Google Cloud Healthcare API - Part 2 (link)

Getting to know the Google Cloud Healthcare API - Part 3 (link)