

Chapter 1

Introduction to the Google Professional Cloud Architect Exam

PROFESSIONAL CLOUD ARCHITECT CERTIFICATION EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ Section 1: Designing and planning a cloud solution architecture

- 1.1 Designing a solution infrastructure that meets business requirements. Considerations include:
 - Business use cases and product strategy
 - Cost optimization
 - Supporting the application design
 - Integration with external systems
 - Movement of data
 - Design decision tradeoffs
 - Build, buy, modify, or deprecate
 - Success measurements (e.g., Key Performance Indicators (KPI), Return on Investment (ROI), metrics)
 - Compliance and observability



This Study Guide is designed to help you acquire the technical knowledge and analytical skills that you will need to pass the Google Cloud Professional Architect certification exam. This exam is designed to evaluate your skills for assessing business requirements, identifying technical requirements, and mapping those requirements to solutions using Google Cloud products, as well as monitoring and maintaining those solutions. This breadth of topics alone is enough to make this a challenging exam. Add to that the need for soft skills, such as working with colleagues in order to understand their business requirements, and you have an exam that is difficult to pass.

The Google Cloud Professional Architect exam is not a body of knowledge exam. You can know Google Cloud product documentation in detail, memorize most of what you read in this guide, and view multiple online courses, but that will not guarantee that you pass the exam. You will be required to exercise judgment. You will have to understand how business requirements constrain your options for choosing a technical solution. You will be asked the kinds of questions a business sponsor might ask about implementing their project.

This chapter will review the following:

- Exam objectives
- Scope of the exam
- Case studies written by Google and used as the basis for some exam questions
- Additional resources to help in your exam preparation

Exam Objectives

The Google Cloud Professional Cloud Architect exam will test your architect skills, including the following:

- Planning cloud solutions
- Managing and provisioning cloud solutions
- Securing systems and processes
- Analyzing and optimizing technical and business processes
- Managing implementations
- Ensuring solution and operations reliability

It is clear from the exam objectives that the test covers the full lifecycle of solution development from inception and planning through monitoring and maintenance.

Analyzing Business Requirements

An architect starts the planning phase by collecting information, starting with business requirements. You might be tempted to start with technical details about the current solution. You might want to ask technical questions so that you can start eliminating options. You may even think that you've solved this kind of problem before and you just have to pick the right architecture pattern. Resist those inclinations if you have them. All architecture design decisions must be made in the context of business requirements.

Business requirements define the operational landscape in which you will develop a solution. Example business requirements are as follows:

- The need to reduce *capital expenditures*
- Accelerating the pace of software development
- Reporting on *service-level objectives*
- Reducing time to recover from an incident
- Improving compliance with industry regulations

Business requirements may be about costs, customer experience, or operational improvements. A common trait of business requirements is that they are rarely satisfied by a single technical decision.

Reducing Operational Expenses

Reducing operational expenses may be satisfied by using managed services instead of operating services yourself, accepting different services commitments such as preemptible virtual machines and Pub/Sub Lite, and using services that automatically scale to load.

Managed services reduce the workload on systems administrators and DevOps engineers because they eliminate some of the work required when managing your own implementation of a platform. Note that while managed services can reduce costs, that is not always the case; if cost is a key driver for selecting a managed service, it is important to verify that managed services will actually cost less. A database administrator, for example, would not have to spend time performing backups or patching operating systems if they used Cloud SQL instead of running a database on Compute Engine instances or in their own data center. BigQuery is a widely used data warehouse and analytics managed service that can significantly reduce the cost of data warehousing by eliminating many database administrator tasks, such as managing storage infrastructure.

Some services have the option of trading some availability, scalability, or reliability features for lower costs. *Preemptible VMs*, for example, are low-cost instances that can be shut

down at any time but can run up to 24 hours before they will be preempted, that is, shut down and no longer available to you. They are a good option for batch processing and other tasks that are easily recovered and restarted. Pub/Sub Lite can be an order of magnitude less expensive than Pub/Sub but comes with lower availability and durability. Pub/Sub Lite is recommended only when the cost savings justify additional operational work to reserve and manage resource capacity.

Autoscaling enables engineers to deploy an adequate number of resources needed to meet the load on a system. In a Compute Engine Managed Instance Group, additional virtual machines are added to the group when demand is high; when demand is low, the number of instances is reduced. With autoscaling, organizations can stop pre-purchasing infrastructure to meet peak capacity and can instead scale their infrastructure to meet the immediate need. With Cloud Run, when a service is not receiving any traffic, the revision of that service is scaled to zero and no costs are incurred.

Accelerating the Pace of Development

Successful businesses are constantly innovating. Agile software development practices are designed to support rapid development, testing, deployment, and feedback.

A business that wants to accelerate the pace of development may turn to managed services to reduce the operational workload on their operations teams. Managed services also allow engineers to implement services, such as image processing and natural language processing, which they could not do on their own if they did not have domain expertise on the team.

Continuous integration and *continuous delivery* are additional practices within software development. The idea is that it's best to integrate small amounts of new code frequently so that it can be tested and deployed rather than trying to release many changes at one time. Small releases are easier to review and debug. They also allow developers to get feedback from colleagues and customers about features, performance, and other factors.

As an architect, you may have to work with monolithic applications that are difficult to update in small increments. In that case, there may be an implied business requirement to consider decomposing the monolithic application into a *microservice architecture*. If there is an interest in migrating to a microservice architecture, then you will need to decide if you should migrate the existing application into the cloud as is, known as *lift and shift*, or you should begin transforming the application during the cloud migration. Alternatively, you could also rebuild on the cloud using cloud-native design without migrating, which is known as *rip and replace*.

There is no way to decide about this without considering business requirements. If the business needs to move to the cloud as fast as possible to avoid a large capital expenditure on new equipment or to avoid committing to a long-term lease in a co-location data center or if the organization wants to minimize change during the migration, then lift and shift is the better choice. Most importantly, you must assess if the application can run in the cloud with minimal modification. Otherwise, you cannot perform a lift-and-shift migration.

If the monolithic application is dependent on deprecated components and written in a language that is no longer supported in your company, then rewriting the application or using a third-party application is a reasonable choice.

Reporting on Service-Level Objectives

The operational groups of a modern business depend on IT applications. A finance department needs access to accounting systems. A logistics analyst needs access to data about how well the fleet of delivery vehicles is performing. The sales team constantly queries and updates the customer management system. Different business units will have different business requirements around the availability of applications and services.

A finance department may only need access to accounting systems during business hours. In that case, upgrades and other maintenance can happen during off-hours and would not require the accounting system to be available during that time. The customer management system, however, is typically used 24 hours a day, every day. The sales team expects the application to be available all the time. This means that support engineers need to find ways to update and patch the customer management system while minimizing or even avoiding downtime.

Requirements about availability are formalized in *service-level objectives (SLOs)*. SLOs can be defined in terms of availability, such as being available 99.9 percent of the time. A database system may have SLOs around durability or the ability to retrieve data. For example, the human resources department may have to store personnel data reliably for seven years, and the storage system must guarantee that there is a less than 1 in 10 billion chances of an object being lost. Interactive systems have performance-related SLOs. A web application SLO may require a page loading average response time of 2 seconds with a 95th percentile of 4 seconds.

Logging and monitoring data are used to demonstrate compliance with SLOs. The *Cloud Logging* service collects information about significant events, such as a disk running out of space. *Cloud Monitoring* collects metrics from infrastructure, services, and applications such as average CPU utilization during a particular period of time or the number of bytes written to a network in a defined time span. Developers can create reports and dashboards using logging details and metrics to monitor compliance with SLOs. These metrics are known as *service-level indicators (SLIs)*.

Reducing Time to Recover from an Incident

Incidents, in the context of IT services, are a disruption that causes a service to be degraded or unavailable. An incident can be caused by single factors, such as an incorrect configuration. Often, there is no single root cause of an incident. Instead, a series of failures and errors contributes to a service failure.

For example, consider an engineer on call who receives a notification that customer data is not being processed correctly by an application. In this case, a database is failing to complete a transaction because a disk is out of space, which causes the application writing to the database to block while the application repeatedly retries the transaction in rapid succession. The application stops reading from a message queue, which causes messages to accumulate until the maximum size of the queue is reached, at which point the message queue starts to drop data.

Once an incident begins, systems engineers and system administrators need information about the state of components and services. To reduce the time to recover, it is best to collect metrics and log events and then make them available to engineers at any time, especially during an incident response.

The incident might have been avoided if database administrators created alerts on free disk space or if the application developer chose to handle retries using *exponential backoff* instead of simply retrying as fast as possible until it succeeds. Alerting on the size of the message queue could have notified the operations team of a potential problem in time to make adjustments before data was dropped.

Improving Compliance with Industry Regulations

Many businesses are subject to government and industry regulations. Regulations range from protecting the privacy of customer data to ensuring the integrity of business transactions and financial reporting. Major regulations include the following:

- *Health Insurance Portability and Accountability Act (HIPAA)*, a healthcare regulation
- *Children’s Online Privacy Protection Act (COPPA)*, a privacy regulation
- *Sarbanes–Oxley Act (SOX)*, a financial reporting regulation
- *Payment Card Industry Data Standard (PCI)*, a data protection regulation for credit card processing
- *General Data Protection Regulation (GDPR)*, a European Union privacy protection regulation

Complying with privacy regulations usually requires controls on who can access and change protected data, where it is stored, and under what conditions data may be retained by a business. As an architect, you will have to develop schemes for controls that meet regulations. Fine-grained access controls may be used to control further who can update data. When granting access, follow security best practices, such as granting only the permissions needed to perform one’s job and separating high-risk duties across multiple roles. For more on security best practices, see Chapter 7, “Designing for Security and Legal Compliance.”

Business requirements define the context in which architects make design decisions. On the Google Cloud Professional Architect exam, you must understand business requirements and how they constrain technical options and specify characteristics required in a technical solution.

Business Terms to Know

Capital Expenditure (Capex) Funds spent to acquire assets, such as computer equipment, vehicles, and land. Capital expenditures are used to purchase assets that will have a useful life of at least a few years. The other major type of expenditure is

operational expenditures. Capital expenses are spread over multiple years, with only a portion of the capital expense impacting the bottom line for each of the years.

Compliance Implementing controls and practices to meet the requirements of regulations, including security, monitoring, and verification that controls meet requirements.

Digital Transformation Major changes in businesses as they adopt information technologies to develop new products, improve customer service, optimize operations, and make other major improvements enabled by technology. Brick-and-mortar retailers using mobile technologies to promote products and engage with customers is an example of digital transformation. Digital transformations usually include some cloud component.

Governance Procedures and practices used to ensure that policies and principles of organizational operations are followed. Governance is the responsibility of directors and executives within an organization.

Key Performance Indicator (KPI) A measure that provides information about how well a business or organization is achieving an important or key objective. For example, an online gaming company may have KPIs related to the number of new players acquired per week, total number of player hours, and operational costs per player.

Line of Business The parts of a business that deliver a particular class of products and services. For example, a bank may have consumer banking and business banking lines, while an equipment manufacturer may have industrial as well as agricultural lines of business. Different lines of business within a company will have some business and technical requirements in common as well as their own distinct needs.

Operational Expenditures (Opex) An expense paid for from the operating budget, not the capital budget.

Operating Budget A budget allocating funds to meet the costs of labor, supplies, and other expenses related to performing the day-to-day operations of a business. Contrast this to capital expenditure budgets, which are used for longer-term investments.

Service-Level Agreement (SLA) An agreement between a provider of a service and a customer using the service. SLAs define responsibilities for delivering a service and consequences when responsibilities are not met.

Service-Level Indicator (SLI) A metric that reflects how well a service-level objective is being met. Examples include latency, throughput, and error rate.

Service-Level Objective (SLO) An agreed-upon target for a measurable attribute of a service that is specified in a service-level agreement.

Analyzing Technical Requirements

Technical requirements specify features of a system that relate to functional and nonfunctional performance. Functional features include providing Atomicity, Consistency, Reliability, and Durability (ACID) transactions in a database, which guarantees that transactions are atomic, consistent, isolated, and durable; ensuring *at least once* delivery in a messaging system; and encrypting data at rest. Nonfunctional features are the general features of a system, including scalability, reliability, observability, and maintainability.

Functional Requirements

The exam will require you to understand functional requirements related to computing, storage, and networking. The following are some examples of the kinds of issues you will be asked about on the exam.

Understanding Compute Requirements

Google Cloud has a variety of computing services, including *Compute Engine*, *App Engine*, *Cloud Functions*, *Cloud Run*, and *Kubernetes Engine*. As an architect, you should be able to determine when each of these platforms is the best option for a use case. For example, if there is a technical requirement to use a virtual machine running a particular hardened version of Linux, then Compute Engine is the best option. Sometimes, though, the choice is not so obvious.

If you want to run containers in a managed service on Google Cloud Platform (GCP), you could choose from App Engine Flexible, Cloud Run, or Kubernetes Engine. If you already have application code running in App Engine and you intend to run a small number of containers, then *App Engine Flexible* is a good option. If you plan to deploy and manage a large number of containers and want to use a service mesh like *Anthos Service Mesh* to secure and monitor microservices, *Kubernetes Engine* is a better option. If you are running stateless containers that do not require Kubernetes features such as namespaces or node allocation and management features, then *Cloud Run* is a good option.

Understanding Storage Requirements

There are even more options when it comes to storage. There are several factors to consider when choosing a storage option, including how the data is structured, how it will be accessed and updated, and for how long it will be stored.

Let's look at how you might decide which data storage service to use given a set of requirements. Structured data fits well with both relational and *NoSQL* databases. If SQL is required, then your choices are Cloud SQL, Spanner, BigQuery, or running a relational database yourself in Compute Engine. If you require a global, strongly consistent transactional data store, then Spanner is the best choice, while Cloud SQL is a good choice for regional-scale databases. If the application using the database requires a flexible schema, then you should consider NoSQL options. Cloud Firestore is a good option when a document store is needed, while Bigtable is well suited for ingesting large volumes of data at low latency.

Of course, you could run a NoSQL database in Compute Engine. If a service needs to ingest time-series data at low latency and one of the business requirements is to maximize the use of managed services, then Bigtable should be used. If there is no requirement to use managed services, you might consider deploying Cassandra to a cluster in Compute Engine. This would be a better choice, for example, if you are planning a lift-and-shift migration to the cloud and are currently running Cassandra in an on-premises data center.

When long-term archival storage is required, then Cloud Storage is the best option. Since Cloud Storage has several classes to choose from, you will have to consider access patterns and reliability requirements when choosing a storage class. If the data is frequently accessed, Standard Storage class storage is appropriate. If high availability of access to the data is a concern or if data will be accessed from different areas of the world, you should consider multiregional or dual-region storage. If data will be infrequently accessed, then Nearline, Coldline, or Archive storage is a good choice. Nearline storage is designed for data that won't be accessed more than once a month and will be stored at least 30 days. Coldline storage is used for data that is stored at least 90 days and accessed no more than once every three months. Archive storage is well suited for data that will be accessed not more than once a year. Nearline, Coldline, and Archive storage have slightly lower availability than Standard Storage.

Understanding Network Requirements

Networking topics that require an architect tend to fall into two categories: structuring virtual private clouds and supporting hybrid cloud computing.

Virtual private clouds (VPCs) isolate a Google Cloud Platform customer's resource. Architects should know how to configure VPCs to meet requirements about who can access specific resources, the kinds of traffic allowed in or out of the network, and communications between VPCs. To develop solutions to these high-level requirements, architects need to understand basic networking components such as the following:

- Firewalls and firewall rules
- Domain name services (DNS)
- *CIDR* blocks and IP addressing
- Autogenerated and custom subnets
- *VPC peering*

Many companies and organizations adopting cloud computing also have their own data centers. Architects need to understand options for networking between on-premises data centers and the Google Cloud Platform network. Options include using a virtual private network (VPN), Dedicated Interconnect, and Partner Interconnects.

Virtual private networks are a good choice when bandwidth demands are not high and data is allowed to traverse the public Internet.

Dedicated Interconnects are used when a 10 Gbps connection is needed and both your on-premises point of presence and a Google point of presence are in the same physical location.

If you do not have point of presence co-located with a Google point of presence, a Partner Interconnect can be used. In that case, you would provision a connection between your point-of-presence location and a Google point of presence using the telecommunications partner's equipment.

Nonfunctional Requirements

Nonfunctional requirements often follow from business requirements. They include the following:

- Availability
- Reliability
- Scalability
- Durability
- Observability

Availability is a measure of the time that services are functioning correctly and accessible to users. Availability requirements are typically stated in terms of percent of time a service should be up and running, such as 99.99 percent. Fully supported Google Cloud services have SLAs for availability so that you can use them to help guide your architectural decisions. Note, alpha and beta products typically do not have SLAs.

Reliability is a closely related concept to availability. *Reliability* is a measure of the probability that a service will continue to function under some load for a period of time. The level of reliability that a service can achieve is highly dependent on the availability of infrastructure upon which it depends.

Scalability is the ability of a service to adapt its infrastructure to the load on the system. When load decreases, some resources may be shut down. When load increases, resources can be added. Autoscalers and managed instance groups are often used to ensure scalability when using Compute Engine. One of the advantages of services like Cloud Storage and App Engine is that scalability is managed by GCP, which reduces the operational overhead on DevOps teams.

Durability is used to measure the likelihood that a stored object will be retrievable in the future. Cloud Storage has 99.99999999 percent (eleven 9s) durability guarantees, which means it is extremely unlikely that you will lose an object stored in Cloud Storage. Because of the math, as the number of objects increases, the likelihood that one of them is lost will increase.

Observability is the ability to determine the internal state of a system by examining outputs of the system. Metrics and logs improve observability by providing information about the state of a system over time.

The Google Cloud Professional Cloud Architect exam tests your ability to understand both business requirements and technical requirements, which is reasonable since those skills are required to function as a cloud architect. Security is another common type of non-functional requirement, but that domain is large enough and complex enough to call for an entire chapter. See Chapter 7, “Designing for Security and Legal Compliance.”

Exam Case Studies

The Google Cloud Professional Cloud Architect certification exam uses case studies as the basis for some questions on the exam. Become familiar with the case studies before the exam to save time while taking the test.

Each case study includes a company overview, solution concept, description of existing technical environment, business requirements, and an executive statement. As you read each case study, be sure that you understand the driving business considerations and the solution concept. These provide constraints on the possible solutions.

When existing infrastructure is described, think of what GCP services could be used as a replacement if needed. For example, Cloud SQL can be used to replace an on-premises MySQL server, Cloud Dataproc can replace self-managed Spark and Hadoop clusters, and Cloud Pub/Sub can be used instead of RabbitMQ.

Read for the technical implications of the business statements—they may not be stated explicitly. Business statements may imply additional requirements that the architect needs to identify without being explicitly told of a requirement.

Also, think ahead. What might be needed a year or two from now? If a business is using batch uploads to ingest data now, what would change if they started to stream data to GCP-based services? Can you accommodate batch processing now and readily adapt to stream processing in the future? Two obvious options are Cloud Dataflow and Cloud Dataproc.

Cloud Dataproc is a managed Spark and Hadoop service that is well suited for batch processing. Spark has support for stream processing, and if you are migrating a Spark-based batch processing system, then using Cloud Dataproc may be the fastest way to support stream processing.

Cloud Dataflow supports both batch and stream processing by implementing an Apache Beam runner, which is an open source model for implementing data workflows. Cloud Dataflow has several key features that facilitate building data pipelines, such as supporting commonly used languages like Python, Java, and SQL; providing native support for exactly one processing and event time; and implementing periodic checkpoints.

Choosing between the two will depend on details such as how the current batch processing is implemented and other implementation requirements, but typically for new development, Cloud Dataflow is the preferred option.

The case studies are available online here:

EHR Healthcare services.google.com/fh/files/blogs/master_case_study_ehr_healthcare.pdf

Helicopter Racing League services.google.com/fh/files/blogs/master_case_study_helicopter_racing_league.pdf

Mountkirk Games `services.google.com/fh/files/blogs/master_case_study_mountkirk_games.pdf`

TerramEarth `services.google.com/fh/files/blogs/master_case_study_terramearth.pdf`

The case studies are summarized in the following sections.

EHR Healthcare

In the EHR Healthcare cases study, you will have to assess the needs of an electronic health records software company. The company has customers in multiple countries, and the business is growing. The company wants to scale to meet the needs of new business, provide for disaster recovery, and adapt agile software practices, such as frequent deployments.

Business and Technical Considerations

EHR Healthcare uses multiple colocation facilities, and the lease on one of those facilities is expiring soon.

Customers use applications that are containerized and running in Kubernetes. Both relational and NoSQL databases are in use. Users are managed with Microsoft Active Directory. Open source tools are used for monitoring, and although there are alerts in place, email notifications about alerts are often ignored.

Business requirements include onboarding new clients as soon as possible, maintaining a minimum of 99.9 percent availability for applications used by customers, improving observability into system performance, ensuring compliance with relevant regulations, and reducing administration costs.

Technical requirements include maintaining legacy interfaces, standardizing on how to manage containerized applications, providing for high-performance networking between on-premises systems and GCP, providing consistent logging, provisioning and scaling new environments, creating interfaces for ingesting data from new clients, and reducing latency in customer applications.

The company has experienced outages and struggles to manage multiple environments.

Architecture Considerations

From the details provided in the case study, we can quickly see several factors that will influence architecture decisions.

The company has customers in multiple countries, and reducing latency to customers is a priority. This calls for a multiregional deployment of services, which will also help address disaster recovery requirements. Depending on storage requirements, multiregional Cloud Storage may be needed. If a relational database is required to span regions, then Cloud Spanner may become part of the solution.

EHR Healthcare is already using Kubernetes, so Kubernetes Engine will likely be used. Depending on the level of control they need over Kubernetes, they may be able to reduce operations costs by using Autopilot mode of Kubernetes instead of Standard mode.

The company uses Microsoft Active Directory to manage identities, so you may want to use Cloud Identity with Active Directory as an identity provider (IdP) for federating identities.

To improve deployments of multiple environments, you should treat infrastructure as code using Cloud Deployment Manager or Terraform. Cloud Build, Cloud Source Repository, and Artifact Registry are key to supporting an agile continuous integration/continuous delivery.

Current logging and monitoring are insufficient given the problems with outages and ignored alert messages. Engineers may be experiencing alert fatigue caused by too many alerts that either are false positives or provide insufficient information to help resolve the incident. Cloud Monitoring and Cloud Logging will likely be included in a solution.

Helicopter Racing League

The Helicopter Racing League case study describes a global sports provider specializing in helicopter racing at regional and worldwide scales. The company streams races around the world. In addition, it provides race predictions throughout the race.

Business and Technical Considerations

The company wants to increase its use of managed artificial intelligence (AI) and machine learning (ML) services as well as serving content closer to racing fans.

The Helicopter Racing League runs its services in a public cloud provider, and initial video recording and editing is performed in the field and then uploaded to the cloud for additional processing on virtual machines. The company has truck-mounted mobile data centers deployed to race sites. An object storage system is used to store content. The deep learning platform TensorFlow is used for predictions, and it runs on VMs in the cloud.

The company is focused on expanding the use of predictive analytics and reducing latency to those watching the race. They are particularly interested in predictions about race results, mechanical failures, and crowd sentiment. They would also like to increase the telemetry data collected during races. Operational complexity should be minimized while still ensuring compliance with relevant regulations.

Specific technical requirements include increasing prediction accuracy, reducing latency for viewers, increasing post-editing video processing performance, and providing additional analytics and data mart services.

Architecture Considerations

The emphasis on AI and ML makes the Helicopter Racing League a candidate for Vertex AI services. Since they are using TensorFlow, performance may be improved using GPUs or TPUs to build machine learning models.

Improving the accuracy of predictive models will likely require additional data or larger ML models, possibly both. Cloud Pub/Sub is ideal for ingesting large volumes of telemetry data. Services can run in Kubernetes Engine with appropriate scaling configurations and

using a Google Cloud global load balancer. The Helicopter Racing League should consider adopting MLOps practices, including automated CI/CD for ML pipelines, such as Vertex Pipelines.

The league has racing fans across the globe, and latency is a key consideration, so Premium Tier network services should be used over the lower-performance Standard Network Tier. Cloud CDN can be used for high-performance edge caching of recorded content to meet latency requirements.

BigQuery would be a good option for deploying data marts and supporting analytics since it scales well and is fully managed.

Mountkirk Games

The Mountkirk Games case study is about a developer of online, multiplayer games for mobile devices. It has migrated on-premises workloads to Google Cloud. It is creating a game that will enable hundreds of players to play in geospecific digital arenas. The game will include a real-time leader board.

Business and Technical Considerations

The game will be deployed on Google Kubernetes Engine (GKE) using a global load balancer along with a multiregion Cloud Spanner cluster. Some existing games that were migrated to Google Cloud are running on virtual machines although they will be eventually migrated to GKE. Popular legacy games are isolated in their own projects in the resource hierarchy while those with less traffic have been consolidated into one project.

Business sponsors of the game want to support multiple gaming devices in multiple geographic regions in a way that scales to meet demand. Server-side GPU processing will be used to render graphics that can be used on multiple platforms. Latency and costs should be minimized, and the company prefers to use managed services and pooled resources.

Structured game activity logs should be stored for analysis in the future. Mountkirk Games will be making frequent changes and want to be able to rapidly deploy new features and bug fixes.

Architecture Considerations

Mountkirk Games has completed a migration to Google Cloud using a lift-and-shift approach. Legacy games will eventually be migrated from VMs to GKE, but the new game is a higher priority.

The new game will support multiple device platforms, so some processing, like rendering graphics, will be done on the server side to ensure consistency in graphics processing and minimizing the load on players' devices. To minimize latency, plan for global load balancing and multiregion deployment of services in GKE.

Cloud Logging can ingest custom log data, so it should be used to collect game activity logs. Since Cloud Logging stores logs for only 30 days, you will likely need to create a log sink to store the data in Cloud Storage or BigQuery. Since the logs are structured and

you will be analyzing the logs, storing them in BigQuery is a good option. At the time of writing, in North America the cost of active storage in BigQuery is about the same as the cost of Standard Storage in Cloud Storage. The cost of BigQuery's Long-term Storage is also about the same as Nearline Storage in Cloud Storage. Prices vary by region and may vary over time.

TerramEarth

The TerramEarth case study describes a heavy equipment manufacturer for the agriculture and mining industries. The company has hundreds of dealers in 100 countries with more than 2 million vehicles in operation. The company is growing at 20 percent annually.

Business and Technical Considerations

Vehicles generate telemetry data from sensors. Most of the data collected is compressed and uploaded after the vehicle returns to its home base. A small amount of data is transmitted in real time. Each vehicle generates from 200 to 500 MB of data per day.

Data aggregation and analysis is performed in Google Cloud. Significant amounts of sensor data from manufacturing plants are stored in legacy inventory and logistics management applications running in private data centers. Those data centers have multiple network interconnects to GCP.

Business sponsors want to predict and detect vehicle malfunctions and ship replacement parts just in time for repairs. They also want to reduce operational costs, increase development speed, support remote work, and provide custom API services for partners.

An HTTP API access layer for legacy systems will be developed to minimize disruptions when moving those services to the cloud.

Developers will use a modern CI/CD platform as well as a self-service platform for creating new projects.

Cloud-native solutions for key management will be used along with identity-based access management.

Architecture Considerations

For data that is transmitted in real time, Cloud Pub/Sub can be used for ingestion. If there is additional processing to be done on that data, Cloud Dataflow could be used to read the data from a Pub/Sub topic, process the data, and then write the results to persistent storage. BigQuery would be a good option for additional analytics.

The other data that is uploaded in batch may be stored in Cloud Storage where a Cloud Dataflow job could decompress the files, perform any needed processing, and write the data to BigQuery.

BigQuery has the advantages of being a fully managed, petabyte-scale analytical database that supports the creation of machine learning models without the need to export data. Also, the machine learning functionality is available through SQL functions, making it accessible to relational database users who may not be familiar with specialized machine learning tools.

TerraEarth is a good use case for Vertex AI. Assuming much of the sensor data is highly structured, that is, it is not images or videos, then AutoML Tables may be used for developing models. If deep learning models are used, then GPUs and TPUs may be used as well.

For workflows with more complex dependencies, Cloud Composer is a good option since it allows you to define workflows as directed acyclic graphs. Consider an MLOps workflow that includes training a machine learning model using the latest data, using the model to make predictions about data collected in real time, and initiating the shipment of replacement parts when a component failure is predicted. If the model is not successfully trained, then the existing prediction job should not be replaced. Instead, the training job should be executed again with an update to the prediction job to follow only if training is successful. This kind of workflow management is handled automatically in Cloud Composer.

Summary

The Google Cloud Professional Architect exam covers several broad areas, including the following:

- Planning a cloud solution
- Managing a cloud solution
- Securing systems and processes
- Complying with government and industry regulations
- Understanding technical requirements and business considerations
- Maintaining solutions deployed to production, including monitoring

These areas require business as well as technical skills. For example, since architects regularly work with nontechnical colleagues, it is important for architects to understand issues such as reducing operational expenses, accelerating the pace of development, maintaining and reporting on service-level agreements, and assisting with regulatory compliance. In the realm of technical knowledge, architects are expected to understand functional requirements around computing, storage, and networking as well as nonfunctional characteristics of services, such as availability and scalability.

The exam includes case studies, and some exam questions reference the case studies. Questions about the case studies may be business or technical questions.

Exam Essentials

Assume every word matters in case studies and exam questions. Some technical requirements are stated explicitly, but some are implied in business statements. Review the business requirements as carefully as the technical requirements in each case study. Similarly, when reading an exam question, pay attention to all the statements. What may look like extraneous background information at first may turn out to be information that you need to choose between two options.

Study and analyze case studies before taking the exam. Become familiar with the case studies before the exam to save time while taking the text. You don't need to memorize the case studies, as you'll have access to them during the test. Watch for numbers that indicate the scale of the problem. For example, if you need 10 Gbps, then you should consider a Cloud Interconnect solution over a VPN solution, which works up to about 3 Gbps for each VPN tunnel.

Understand what is needed in the near term and what may be needed in the future. For example, we don't have specific MLOps workflows in the TerramEarth case study. Initially, predictions may be based only on structured data, but some vehicles may have cameras to create images of machine components or the operating environment. In the future, there may be an opportunity to use images of operating environments to automatically detect a problem in the environment that could damage the vehicle. In that case, AutoML Vision Edge may be useful for performing image classification in real time. This requirement is not stated, and not even implied, but it is the kind of planning for the future that architects are expected to do.

Understand how to plan a migration. Migrations are high-risk operations. Data can be lost, and services may be unavailable. Know how to plan to run new and old systems in parallel so that you can compare results. Be able to identify lower-risk migration steps so that they can be scheduled first. Plan for incremental migrations.

Know agile software development practices. You won't have to write code for this exam, but you will need to understand continuous integration/continuous delivery and how to maintain development, test, staging, and production environments. Understand what is meant by an infrastructure-as-code service and how that helps accelerate development and deployment.





Keep in mind that solutions may involve non-Google services or applications. Google has many services, but sometimes the best solution involves a third-party solution. For example, Jenkins and Spinnaker are widely used tools to support continuous integration and deployment. Google Cloud has a code repository, but many developers use GitHub. Sometimes businesses are locked into existing solutions, such as a third-party database. The business may want to migrate to another database solution, but the cost may be too high for the foreseeable future.






Review Questions


1. You have been tasked with interviewing line-of-business owners about their needs for a new cloud application. Which of the following do you expect to find?
 - A. A comprehensive list of defined business and technical requirements
 - B. That their business requirements do not have a one-to-one correlation with technical requirements
 - C. Business and technical requirements in conflict
 - D. Clear consensus on all requirements
2. You have been asked by stakeholders to suggest ways to reduce operational expenses as part of a cloud migration project. Which of the following would you recommend?
 - A. Managed services, preemptible machines, access controls
 - B. Managed services, preemptible machines, autoscaling
 - C. NoSQL databases, preemptible machines, autoscaling
 - D. NoSQL databases, preemptible machines, access controls
3. Some executives are questioning your recommendation to employ continuous integration/continuous delivery (CI/CD). What reasons would you give to justify your recommendation?
 - A. CI/CD supports small releases, which are easier to debug and enable faster feedback.
 - B. CI/CD is used only with preemptible machines and therefore saves money.
 - C. CI/CD fits well with waterfall methodology but not agile methodologies.
 - D. CI/CD limits the number of times code is released.
4. The finance director has asked your advice about complying with a document retention regulation. What kind of service-level objective (SLO) would you recommend to ensure that the finance director will be able to retrieve sensitive documents for at least the next seven years? When a document is needed, the finance director will have up to seven days to retrieve it. The total storage required will be approximately 100 TB.
 - A. High availability SLO
 - B. Durability SLO
 - C. Reliability SLO
 - D. Scalability SLO
5. You are facilitating a meeting of business and technical managers to solicit requirements for a cloud migration project. The term *incident* comes up several times. Some of the business managers are unfamiliar with this term in the context of IT. How would you describe an incident?
 - A. A disruption in the ability of a DevOps team to complete work on time
 - B. A disruption in the ability of the business managers to approve a project plan on schedule





- C. A disruption that causes a service to be degraded or unavailable
 - D. A personnel problem on the DevOps team
- 6. You have been asked to consult on a cloud migration project that includes moving private medical information to a storage system in the cloud. The project is for a company in the United States. What regulation would you suggest that the team review during the requirements-gathering stages?
 - A. General Data Protection Regulations (GDPR)
 - B. Sarbanes–Oxley (SOX)
 - C. Payment Card Industry Data Security Standard (PCI DSS)
 - D. Health Insurance Portability and Accountability Act (HIPAA) 
- 7. You are in the early stages of gathering business and technical requirements. You have noticed several references about needing up-to-date and consistent information regarding product inventory and support for SQL reporting tools. Inventory is managed on a global scale, and the warehouses storing inventory are located in North America, Africa, Europe, and Asia. Which managed database solution in Google Cloud would you include in your set of options for an inventory database?
 - A. Cloud Storage
 - B. BigQuery
 - C. Cloud Spanner 
 - D. Microsoft SQL Server
- 8. A developer at Mountkirk Games is interested in how architects decide which database to use. The developer describes a use case that requires a document store. The developer would rather not manage database servers or have to run backups. What managed service would you suggest the developer consider?
 - A. Cloud Firestore 
 - B. Cloud Spanner
 - C. Cloud Storage
 - D. BigQuery
- 9. Members of your company’s legal team are concerned about using a public cloud service because other companies, organizations, and individuals will be running their systems in the same cloud. You assure them that your company’s resources will be isolated and not network-accessible to others because of what networking resource in Google Cloud?
 - A. CIDR blocks
 - B. Direct connections
 - C. Virtual private clouds 
 - D. Cloud Pub/Sub

10. A startup has recently migrated to Google Cloud using a lift-and-shift migration. They are now considering replacing a self-managed MySQL database running in Compute Engine with a managed service. Which Google Cloud service would you recommend that they consider?
- A. Cloud Dataproc
 - B. Cloud Dataflow
 - C. Cloud SQL 
 - D. PostgreSQL
11. Which of the following requirements from a customer make you think the application should run in Compute Engine and not App Engine?
- A. Dynamically scale up or down based on workload
 - B. Connect to a database
 - C. Run a hardened Linux distro on a virtual machine 
 - D. Don't lose data
12. Mountkirk Games wants to store player game data in a time-series database. Which Google Cloud managed database would you recommend?
-  A. Bigtable
 - B. BigQuery
 - C. Cloud Storage
 - D. Cloud Dataproc
13. The original video captured during helicopter races by the Helicopter Racing League are transcoded and stored for frequent access. The original captured videos are not used for viewing but are stored in case they are needed for unanticipated reasons. The files require high durability but are not likely to be accessed more than once in a five-year period. What type of storage would you use for the original video files?
- A. BigQuery Long Term Storage
 - B. BigQuery Active Storage
 -  C. Cloud Storage Nearline class
 - D. Cloud Storage Archive class
14. The game analytics platform for Mountkirk Games requires analysts to be able to query up to 10 TB of data. What is the best managed database solution for this requirement?
- A. Cloud Spanner
 -  B. BigQuery
 - C. Cloud Storage
 - D. Cloud Dataprep

15. EHR Healthcare business requirements frequently discuss the need to improve observability in their systems. Which of the following Google Cloud Platform services could be used to help improve observability?
- A. Cloud Build and Artifact Registry
 - B. Cloud Pub/Sub and Cloud Dataflow
 - C. Cloud Monitoring and Cloud Logging 
 - D. Cloud Storage and Cloud Pub/Sub

