gcp top 40 services in details that are most used

**ChatGPT said:**

Google Cloud Platform (GCP) offers a wide array of services for developers, data scientists, and enterprises looking to leverage cloud computing. Below are the top 40 GCP services, listed along with details about their uses and key features:

**1. Compute Engine**

* **Purpose**: Infrastructure-as-a-Service (IaaS) for running virtual machines (VMs).
* **Use Cases**: Hosting scalable applications, running custom workloads, or running virtualized environments.
* **Key Features**: Customizable VMs, various OS support, persistent disks, and autoscaling.

**2. Kubernetes Engine**

* **Purpose**: Managed Kubernetes clusters for **orchestrating containers**.
* **Use Cases**: Running containerized applications with high availability and scalability.
* **Key Features**: Automated deployments, scaling, and monitoring of containerized apps.

**3. App Engine**

* **Purpose**: Platform-as-a-Service (PaaS) for building and deploying web apps.
* **Use Cases**: **Serverless app deployment**, automatic scaling, and monitoring.
* **Key Features**: Fully managed, supports popular programming languages, automatic scaling, and integrated with other GCP services.
* **Just focus on appn logic and rest of the things taken care of google , autoscaling and every things taken care by google**

**4. Cloud Functions**

* **Purpose**: Serverless compute for executing event-driven functions.
* **Use Cases**: Real-time data processing, triggering actions based on events, or building microservices.
* **Key Features**: Event-driven, pay-per-use, auto-scaling.
* **When we click on button we want some code to exercute , event drive service , so we just write code and it will be execute based on some specific event**

**5. Cloud Storage**

* **Purpose**: Object storage for any amount of data.
* **Use Cases**: Storing files, backups, images, videos, or any **unstructured data.**
* **Key Features**: Scalable, highly available, and secure storage with global distribution.
* **Used to hold unlimited data , we store data in buckets, reliable like if we store 1lkh object then there is 100% reliality no obnejct will be lost, it can easily integrate with diff services , like using cloud function whenever we upload a file to the bucket then cloud funct get executed ,** unstructured data
* **Used for unstructured data , realiable service ,**

**6. BigQuery**

* **Purpose**: Fully managed data warehouse for running fast SQL analytics on large datasets.
* **Use Cases**: Business intelligence, data analysis, and processing large-scale datasets.
* **Key Features**: Serverless, highly scalable, supports real-time analytics.
* **Used to analyze the large dta set , it’s kind of data warehouse service(allow orgz to store data in the cloud) , serverless , scaleable service**
* **Data warehouse service , used for analysis of large data-set**

**7. Cloud SQL**

* **Purpose**: **Managed relational databases** with support for MySQL, PostgreSQL, and SQL Server.
* **Use Cases**: Web app backends, transactional applications, and managing relational data.
* **Key Features**: Automated backups, **replication, and scaling.**
* **It’s realtational db service , support mysql postgresql , scalable and replicate the data in mzone to ensure high avalalityu**
* **Fully managed**
* **Vertical scaling**
* **Lower latency for single region**
* **Compatibale with mysql , postgresql , and sql server .**
* **Automatically create replicas in multi-azones**

**8. Cloud Spanner**

* **Purpose**: Globally distributed **relational database servic**e **for mission-critical** applications.
* **Use Cases**: Large-scale applications, multi-region databases, and distributed transactions.
* **Key Features**: Horizontal scaling, ACID transactions, and high availability.
* **Horizontally scaling 4**
* **Higher latency due to global coordination**
* **Rdbms service for mission critical appn ,**

**9. Cloud Pub/Sub**

* **Purpose**: **Real-time messaging** service for event-driven systems.
* **Use Cases**: Decoupling components of microservices, real-time analytics, and event streaming.
* **Key Features**: Fully managed, message filtering, and auto-scaling
* **Pub**lisher- send a mesg.
* Topic – named channel wehere messg go
* Subscriber – receiver of msg
* Topic – the place where it posted
* Pub- person who send msg to the gp in whatspp is publisher
* Topic – the whtasapp gp
* Subscriber – everyone who in the gp read the msg
* Suppose there is whatasapp group we have , so whatsapp group is named channel or topic and publisher send msg to group and anyone who received the msg and reply is subscriber

**10. Cloud Datastore**

* **Purpose**: **NoSQL database service for storing non-relational data**.
* **Use Cases**: Storing unstructured or semi-structured data, such as user profiles, session data, etc.
* **Key Features**: High availability, automatic scaling, and ACID transactions.
* **Nosql service , used for holding non relational data , it is automatically scalable as the traffic grows , high avalible – it will automaitacally replicate data in mzones**
* It support key value pair , document db

**11. Cloud Firestore**

* **Purpose**: **Serverless NoSQL document database** for building **web, mobile, and server** apps.
* **Use Cases**: Real-time apps, mobile apps, and offline-first applications.
* **Key Features**: Real-time synchronization, offline support, and integrated security rules.

**12. Cloud Load Balancing**

* **Purpose**: Scalable and fully distributed load balancing for applications.
* **Use Cases**: Distributing traffic across multiple instances, improving performance, and scaling apps.
* **Key Features**: Global load balancing, auto-scaling, SSL termination.

**13. Cloud CDN**

* **Purpose**: Content Delivery Network to accelerate web content delivery.
* **Use Cases**: Faster website load times and reduced latency for static assets.
* **Key Features**: Integration with Cloud Storage, edge caching, and global distribution.

**14. Cloud Identity & Access Management (IAM)**

* **Purpose**: Identity and access management **service to secure access to resources**.
* **Use Cases**: Managing who can access GCP resources and at what level.
* **Key Features**: Fine-grained access control, policy enforcement, and audit logging.
* **We follow the least privlialge principle , don’t give more permission than user needs**
* To control access to the resource – we give persimission to specific users , like for tester
* we only give the permission for testing service – and so on
* using IAM we can control which user can access to which resoruces

**27. Cloud Deployment Manager - VIP**

* **Purpose**: **Infrastructure-as-code service for automating GCP resource provisioning**.
* **Use Cases**: Managing GCP infrastructure, automating resource deployment, and defining resources as code.
* **Key Features**: **Templates, YAML configuration**, and integration with GCP APIs.
* **Automate the creation of VM , network , WE WRITE yaml file and after applying all the resources has been successfully created**.

**Cloud Monitoring**

* **Purpose**: Managed monitoring solution for GCP infrastructure and applications.
* **Use Cases**: Monitoring app performance, resource usage, and availability.
* **Key Features**: Dashboards, **alerts,** and integrated with GCP services.

**Cloud Trace**

* Purpose: **Distributed tracing system for performance** monitoring.
* **Use Cases**: Identifying bottlenecks and improving app performance.
* **Key Features**: Integration with GCP services, detailed latency reports, and insights.

**34. Identity-Aware Proxy**

* **Purpose**: **Secure access to web apps based on user identity** and context.
* **Use Cases**: Protecting internal apps, conditional access, and secure access management.
* **Key Features**: Context-based access control and multi-factor authentication.

Service – account

* Used to access gcp services
* Can control each services using Sa

A service account is an account which is created to be **used by an application or services** that are running on the cloud platform.

Used to access gcp services – we use service accout

**To access google product we use service accounts**

**Cloud Scheduler**

* **Purpose**: Managed cron jobs for running scheduled tasks.
* **Use Cases**: **Periodic database backups**, task scheduling, and triggering batch jobs.
* **Key Features**: Simple **cron job management, scalable**, and integrates with Google Cloud services like Cloud Pub/Sub and Cloud Functions.

**Cloud Run**

* **Purpose**: Fully managed compute platform for running containerized applications without managing servers.
* **Use Cases**: Running stateless applications, microservices, and APIs.
* **Key Features**: Auto-scaling, serverless, and integration with GCP storage and networking services.

**17. Cloud Composer**

* **Purpose**: **Managed Apache Airflow service for orchestrating** workflows.
* **Use Cases**: Automating data pipelines, ETL jobs, and scheduling complex workflows.
* **Key Features**: **Integration with GCP services**, open-source support, and dynamic scheduling.
* **Automate and manage workflows , built on apache airflow**
* **Like loading data , clearning it , storaing it sending alerts, etc.**

**45. Cloud Identity Platform**

* **Purpose**: Identity management service that provides **authentication and authorization for users**.
* **Use Cases**: **Secure user authentication for mobile and web apps**, third-party integrations, and single sign-on (SSO) implementations.
* **Key Features**: Social logins, multi-factor authentication (MFA), and support for common protocols like OAuth 2.0 and OpenID Connect.

**Cloud Filestore**

* **Purpose**: **Managed NFS (Network File System) service for applications** requiring shared file storage.
* **Use Cases**: **Storing data for applications that require file systems**, such as web servers.
* **Key Features**: High performance, scalable, and easily integrated with other GCP services.

**Cloud Key Management**

* **Purpose**: **Managed encryption key storage and** management.
* **Use Cases**: Secure data encryption, key rotation, and compliance management.

**Key Features**: Integration with other GCP services, FIPS 140-2 compliant

**Cloud Build**

* **Purpose**: **Continuous integration and delivery service for building**, testing, and deploying applications.
* **Use Cases**: Automating the software delivery pipeline and improving developer productivity.
* **Key Features**: Support for custom build pipelines, integration with GCP services.

**Cloud Source Repositories**

* **Purpose**: Private Git repositories hosted on Google Cloud.
* **Use Cases**: Storing and version-controlling source code in the cloud.
* **Key Features**: Git-based, integrated with Google Cloud’s continuous integration tools like Cloud Build, and easy collaboration.

**Cloud Run for Anthos**

* **Purpose**: Managed Kubernetes and containerized service for applications running on Google Kubernetes Engine (GKE).
* **Use Cases**: Hybrid cloud setups, containerized applications, and microservices on Kubernetes.
* **Key Features**: Simplified Kubernetes deployment, integrates with Anthos for multi-cloud management.

**15. Cloud VPN**

* **Purpose**: Securely **connect your on-premises network to GCP.**
* **Use Cases**: Hybrid cloud architectures, secure remote access, and private networking.
* **Key Features**: IPsec VPN, high availability, and routing.
* **Secure connecrtion , access internal resources like db** – db in your office ,

**16. Cloud Interconnect**

* **Purpose**: **Direct connection between your on-premises infrastructure and GCP**.
* **Use Cases**: Low-latency connectivity, **large data transfers**, and secure private networking.
* **Key Features**: High throughput, low latency, and multiple connection options.

**17. Cloud Composer**

* **Purpose**: **Managed Apache Airflow service for orchestrating** workflows.
* **Use Cases**: Automating data pipelines, ETL jobs, and scheduling complex workflows.
* **Key Features**: **Integration with GCP services**, open-source support, and dynamic scheduling.
* **Automate and manage workflows , built on apache airflow**
* **Like loading data , clearning it , storaing it sending alerts, etc.**

**18. Cloud AI Platform**

* **Purpose**: Tools and services for **machine learning and AI model development**.
* **Use Cases**: Training, deploying, and managing ML models at scale.
* **Key Features**: AutoML, pre-trained models, and integration with TensorFlow and other frameworks.

**19. Cloud Vision AI**

* **Purpose**: Image recognition and analysis API powered by machine learning.
* **Use Cases**: Image labeling, facial recognition, object detection, and OCR.
* **Key Features**: Pre-trained models, real-time analysis, and multi-language support.

**20. Cloud Natural Language API**

* **Purpose**: Language processing tools for analyzing text.
* **Use Cases**: Sentiment analysis, entity recognition, and text classification.
* **Key Features**: Supports multiple languages and pre-trained models.

**21. Cloud Translation API**

* **Purpose**: Language translation services using machine learning.
* **Use Cases**: Real-time translation, global application support, and localization.
* **Key Features**: Dynamic translation, supports 100+ languages.

**22. Dialogflow**

* **Purpose**: AI-powered chatbot development and conversational interfaces.
* **Use Cases**: Building virtual assistants, chatbots, and automated customer service systems.
* **Key Features**: Natural language understanding, integrations with messaging platforms.

**23. Cloud Pub/Sub Lite**

* **Purpose**: Low-cost, simpler alternative to Cloud Pub/Sub.
* **Use Cases**: Lightweight **event-driven architectures and messaging applications**.
* **Key Features**: Simpler setup, reduced cost, and regional delivery.

**24. Cloud Healthcare API**

* **Purpose**: Manage and analyze healthcare data.
* **Use Cases**: Interoperability in health systems, medical research, and patient data management.
* **Key Features**: Support for HL7, FHIR, and DICOM formats.

**25. Cloud Filestore**

* **Purpose**: **Managed NFS (Network File System) service for applications** requiring shared file storage.
* **Use Cases**: **Storing data for applications that require file systems**, such as web servers.
* **Key Features**: High performance, scalable, and easily integrated with other GCP services.

**26. Cloud Storage for Firebase**

* **Purpose**: Object storage for storing user-generated content.
* **Use Cases**: Web and mobile app development, storing images, videos, and files.
* **Key Features**: Secure file storage, real-time synchronization, and offline support.

**27. Cloud Deployment Manager - VIP**

* **Purpose**: **Infrastructure-as-code service for automating GCP resource provisioning**.
* **Use Cases**: Managing GCP infrastructure, automating resource deployment, and defining resources as code.
* **Key Features**: **Templates, YAML configuration**, and integration with GCP APIs.
* **Automate the creation of VM , network , WE WRITE yaml file and after applying all the resources has been successfully created**.

**28. Cloud Trace**

* Purpose: **Distributed tracing system for performance** monitoring.
* **Use Cases**: Identifying bottlenecks and improving app performance.
* **Key Features**: Integration with GCP services, detailed latency reports, and insights.

**29. Cloud Profiler**

* **Purpose**: Continuous profiling for optimizing application performance.
* **Use Cases**: Analyzing CPU and memory usage to reduce resource consumption.
* **Key Features**: Real-time performance analysis, supports various programming languages.

**30. Cloud Monitoring**

* **Purpose**: Managed monitoring solution for GCP infrastructure and applications.
* **Use Cases**: Monitoring app performance, resource usage, and availability.
* **Key Features**: Dashboards, alerts, and integrated with GCP services.

**31. Cloud Logging**

* **Purpose**: Centralized log management for applications and infrastructure.
* **Use Cases**: Log aggregation, debugging, and performance monitoring.
* **Key Features**: Integrated with GCP services, real-time log analysis.

**32. Cloud Security Command Center**

* **Purpose**: Security and risk management platform for GCP resources.
* **Use Cases**: Identifying vulnerabilities, threat detection, and risk assessment.
* **Key Features**: Real-time threat intelligence, unified security dashboard, and compliance reporting.

**33. Cloud Key Management**

* **Purpose**: **Managed encryption key storage and** management.
* **Use Cases**: Secure data encryption, key rotation, and compliance management.
* **Key Features**: Integration with other GCP services, FIPS 140-2 compliant.

**34. Identity-Aware Proxy**

* **Purpose**: Secure access to web apps based on user identity and context.
* **Use Cases**: Protecting internal apps, conditional access, and secure access management.
* **Key Features**: Context-based access control and multi-factor authentication.

**35. Cloud AI Building Blocks**

* **Purpose**: Set of pre-trained models for AI tasks like vision, language, and translation.
* **Use Cases**: Adding AI capabilities to applications without training models from scratch.
* **Key Features**: Pre-trained models, easy integration, and scalable.

**36. Cloud Build**

* **Purpose**: **Continuous integration and delivery service for building**, testing, and deploying applications.
* **Use Cases**: Automating the software delivery pipeline and improving developer productivity.
* **Key Features**: Support for custom build pipelines, integration with GCP services.

**37. Cloud Dataproc**

* **Purpose**: Managed Spark and Hadoop clusters for big data processing.
* **Use Cases**: Data processing, analytics, and machine learning tasks.
* **Key Features**: Fast deployment, scalability, and integration with other GCP tools.

**38. Cloud Dataprep**

* **Purpose**: Data preparation service for cleaning and transforming data before analysis.
* **Use Cases**: Data wrangling, preparing datasets for analytics.
* **Key Features**: Visual interface, automated data cleaning, and integration with BigQuery.

**39. Cloud Dataflow**

* **Purpose**: Fully managed service for stream and batch processing.
* **Use Cases**: Data pipelines, real-time analytics, and ETL jobs.
* **Key Features**: Unified batch and stream processing, auto-scaling.

**40. Cloud AI Explanations**

* **Purpose**: AI service for explaining model predictions.
* **Use Cases**: Providing transparency in AI models, especially for regulated industries.
* **Key Features**: Model interpretability, fairness analysis, and transparency.

These services span multiple domains such as compute, storage, data analytics, machine learning, and networking. Depending on your needs, these services can be combined to build scalable, high-performance solutions.

4o

**You said:**

continue

**ChatGPT said:**

Certainly! Continuing from where we left off, here are additional Google Cloud Platform (GCP) services, building on the previously mentioned ones:

**41. Cloud AutoML**

* **Purpose**: A suite of machine learning services that allow developers to build custom models without requiring deep expertise in ML.
* **Use Cases**: Custom image classification, text analysis, video intelligence, and more.
* **Key Features**: Easy-to-use interface, pre-built models, automatic model tuning, and supports a variety of use cases like vision, language, and structured data.

**42. Cloud Video Intelligence API**

* **Purpose**: Extract insights from video content using AI-powered video analysis.
* **Use Cases**: Video tagging, content moderation, object recognition, and speech-to-text for video.
* **Key Features**: Real-time video analysis, speech transcription, label detection, and scene change detection.

**43. Cloud Speech-to-Text**

* **Purpose**: Convert audio to text using Google’s machine learning models.
* **Use Cases**: Transcription of spoken content, call center analytics, voice commands, and creating subtitles.
* **Key Features**: Real-time transcription, multi-language support, automatic punctuation, and speaker diarization.

**44. Cloud Text-to-Speech**

* **Purpose**: Convert text into natural-sounding speech using Google’s AI models.
* **Use Cases**: Voice applications, creating lifelike audio for accessibility, and generating automated voice responses.
* **Key Features**: Customizable voice, multiple languages and voices, and real-time streaming.

**45. Cloud Identity Platform**

* **Purpose**: Identity management service that provides authentication and authorization for users.
* **Use Cases**: Secure user authentication for mobile and web apps, third-party integrations, and single sign-on (SSO) implementations.
* **Key Features**: Social logins, multi-factor authentication (MFA), and support for common protocols like OAuth 2.0 and OpenID Connect.

**46. Cloud Workstations**

* **Purpose**: Managed virtual desktop environments for cloud-based development.
* **Use Cases**: Cloud-based development environments, remote workstations, and providing secure development environments.
* **Key Features**: Scalable cloud workstations, integrated with IDEs, and customizable environments.

**47. Cloud Scheduler**

* **Purpose**: Managed cron jobs for running scheduled tasks.
* **Use Cases**: Periodic database backups, task scheduling, and triggering batch jobs.
* **Key Features**: Simple cron job management, scalable, and integrates with Google Cloud services like Cloud Pub/Sub and Cloud Functions.

**48. Cloud Tasks**

* **Purpose**: Fully managed task queue for asynchronous task execution.
* **Use Cases**: Handling background tasks like image processing, sending email notifications, or data processing.
* **Key Features**: Scalable queues, retry policies, and cloud-native integration with other GCP services.

**49. Cloud Memorystore**

* **Purpose**: Fully managed in-memory store compatible with Redis and Memcached.
* **Use Cases**: Caching data, session management, and improving application performance by reducing database load.
* **Key Features**: Low-latency, high-throughput, and fully managed with auto-scaling.

**50. Cloud Data Fusion**

* **Purpose**: Fully managed data integration platform for building ETL pipelines.
* **Use Cases**: Data transformation, integration, and preparation before analysis or machine learning.
* **Key Features**: Visual interface for designing pipelines, integration with BigQuery, and support for real-time data processing.

**51. Cloud Functions for Firebase**

* **Purpose**: Serverless functions to trigger events in Firebase applications.
* **Use Cases**: Responding to Firebase database changes, user authentication, and more.
* **Key Features**: Simple integration with Firebase, serverless, and automatic scaling.

**52. Firebase Realtime Database**

* **Purpose**: NoSQL cloud database for syncing and storing data in real-time.
* **Use Cases**: Building real-time applications, chat apps, live collaboration tools.
* **Key Features**: Real-time synchronization, offline support, and automatic scaling.

**53. Firebase Hosting**

* **Purpose**: Fully managed hosting for static websites and dynamic web apps.
* **Use Cases**: Hosting web apps, landing pages, and static content.
* **Key Features**: SSL certificates, global CDN, and easy deployment.

**54. Firebase Cloud Messaging**

* **Purpose**: Service to send notifications and messages to users on iOS, Android, and the web.
* **Use Cases**: Push notifications, messaging for mobile apps, and cross-platform communication.
* **Key Features**: Real-time messaging, message targeting, and integration with other Firebase services.

**55. Cloud Source Repositories**

* **Purpose**: Private Git repositories hosted on Google Cloud.
* **Use Cases**: Storing and version-controlling source code in the cloud.
* **Key Features**: Git-based, integrated with Google Cloud’s continuous integration tools like Cloud Build, and easy collaboration.

**56. Cloud AI Notebooks**

* **Purpose**: Managed Jupyter notebooks environment for data science and machine learning.
* **Use Cases**: Building machine learning models, exploring data, and creating interactive workflows.
* **Key Features**: Easy-to-use notebooks, GPU support, and integration with TensorFlow, Keras, and other frameworks.

**57. Cloud Build API**

* **Purpose**: API for automating build processes with Cloud Build.
* **Use Cases**: Automating CI/CD pipelines and software builds.
* **Key Features**: Integration with source repositories, multi-platform support, and webhook support for automation.

**58. Cloud Run**

* **Purpose**: Fully managed compute platform for running containerized applications without managing servers.
* **Use Cases**: Running stateless applications, microservices, and APIs.
* **Key Features**: Auto-scaling, serverless, and integration with GCP storage and networking services.

**59. Cloud Deployment Manager**

* **Purpose**: Infrastructure-as-code tool for managing GCP resources.
* **Use Cases**: Automating infrastructure provisioning, resource orchestration, and managing cloud resources as code.
* **Key Features**: YAML configuration files, supports multi-region deployments, and integrates with the Google Cloud SDK.

**60. Cloud Virtual Machines (VM)**

* **Purpose**: Compute instances for running various workloads, including web apps, databases, and batch jobs.
* **Use Cases**: Running virtualized environments on-demand, managing compute resources.
* **Key Features**: Customizable machine types, persistent disk storage, and integration with networking and security services.

**61. Cloud Identity-Aware Proxy (IAP)**

* **Purpose**: Securely manage access to applications by validating the identity of users.
* **Use Cases**: Protecting internal apps, managing who can access what based on identity and context.
* **Key Features**: Context-aware access, easy integration with other GCP resources, and multi-factor authentication support.

**62. Cloud Shell**

* **Purpose**: Interactive command-line environment in the browser for managing GCP resources.
* **Use Cases**: Administering GCP projects, managing resources without requiring local setup.
* **Key Features**: Pre-configured environment, includes Google Cloud SDK, and no need for local installations.

**63. Cloud Deployment Manager**

* **Purpose**: Service to automate deployment and configuration of GCP resources using templates.
* **Use Cases**: Automating infrastructure provisioning, creating standardized environments, and orchestrating complex deployments.
* **Key Features**: YAML-based configuration files, templating system, and multi-cloud resource orchestration.

**64. Cloud Cost Management**

* **Purpose**: Helps you manage and optimize your GCP spending.
* **Use Cases**: Budgeting, cost monitoring, and resource optimization.
* **Key Features**: Cost forecasts, billing reports, and resource optimization tools.

**65. Cloud Encryption**

* **Purpose**: Ensures that your data is encrypted in transit and at rest.
* **Use Cases**: Securing sensitive data, compliance with privacy regulations.
* **Key Features**: Integration with Cloud Key Management, AES-256 encryption, and customer-managed encryption keys.

**66. Cloud Run for Anthos**

* **Purpose**: Managed Kubernetes and containerized service for applications running on Google Kubernetes Engine (GKE).
* **Use Cases**: Hybrid cloud setups, containerized applications, and microservices on Kubernetes.
* **Key Features**: Simplified Kubernetes deployment, integrates with Anthos for multi-cloud management.

These additional services further extend the capabilities of Google Cloud Platform, especially in the realms of development, AI, machine learning, data management, and infrastructure automation. With GCP’s vast ecosystem of tools and services, you can create a scalable, secure, and efficient cloud-based infrastructure tailored to your business or project needs.