



Social Media Apps Programming

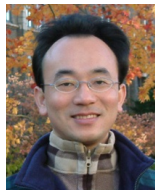
Google Cloud Platform

1061SMAP10

TLMXM1A (8648) (M2143) (Fall 2017)

(MIS MBA) (2 Credits, Elective) [Full English Course]

Fri 8,9 (15:10-17:00) B206



Min-Yuh Day, Ph.D.

Assistant Professor

Department of Information Management

Tamkang University

<http://mail.tku.edu.tw/myday>

2017-12-08



Course Schedule (1/2)



Tamkang
University

Week	Date	Subject/Topics
1	2017/09/22	Course Orientation and Introduction to Social Media and Mobile Apps Programming
2	2017/09/29	Introduction to Android / iOS Apps Programming
3	2017/10/06	Developing Android Native Apps with Java (Android Studio)
4	2017/10/13	Developing iPhone / iPad Native Apps with Swift (XCode)
5	2017/10/20	Mobile Apps using HTML5/CSS3/JavaScript
6	2017/10/27	jQuery Mobile
7	2017/11/03	Create Hybrid Apps with Phonegap
8	2017/11/10	jQuery Mobile/Phonegap
9	2017/11/17	jQuery Mobile/Phonegap

Course Schedule (2/2)



**Tamkang
University**

Week	Date	Subject/Topics
10	2017/11/24	Midterm Project Report
11	2017/12/01	Case Study on Social Media Apps Programming and Marketing in Google Play and App Store
12	2017/12/08	Google Cloud Platform
13	2017/12/15	Google App Engine
14	2017/12/22	Google Map API
15	2017/12/29	Facebook API (Facebook JavaScript SDK) (Integrate Facebook with iOS/Android Apps)
16	2018/01/05	Twitter API
17	2018/01/12	Final Project Presentation
18	2018/01/19	Final Exam Week (Final Project Presentation)

Outline

- **Google Cloud Platform**

- Google App Engine



- Google Cloud Datastore



- Google Cloud Endpoints



- **Firebase**



Firebase

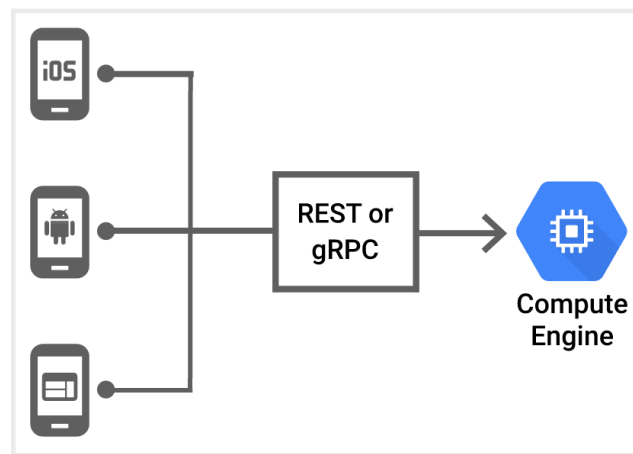
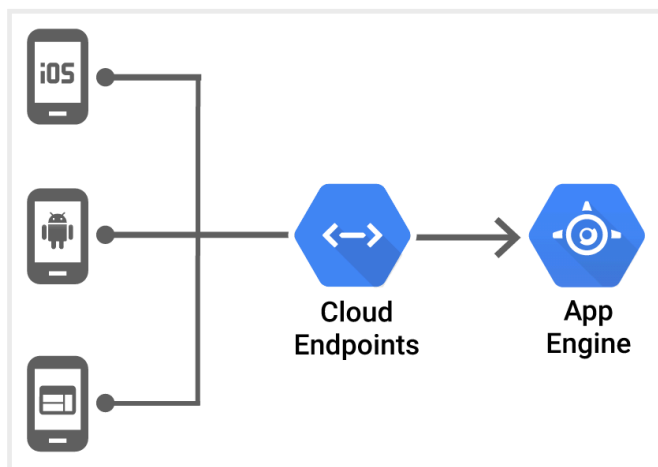
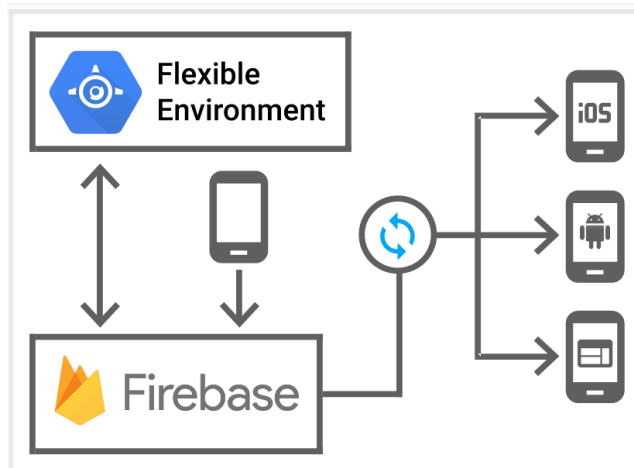
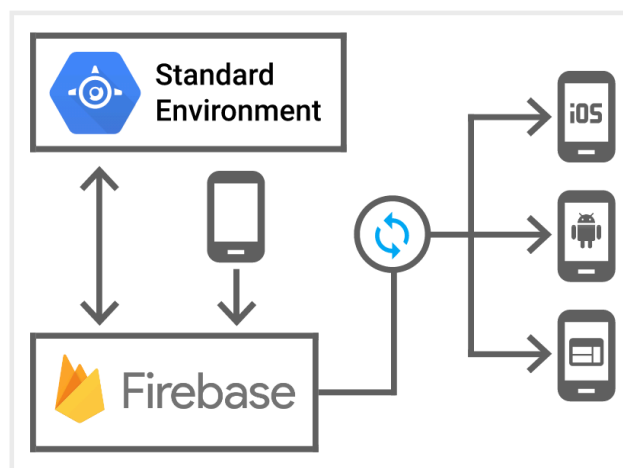
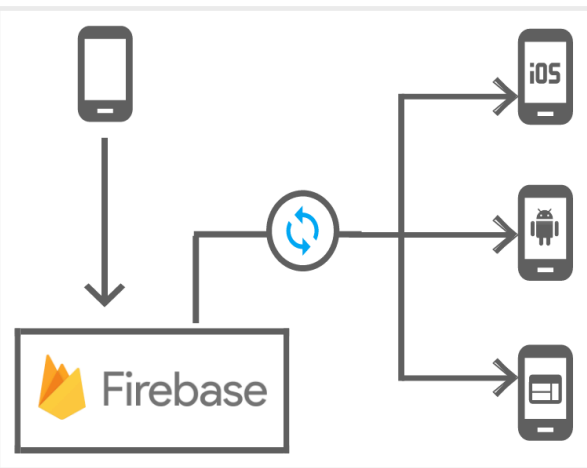
- **Mobile App with Google Cloud Platform**



Google Cloud Platform

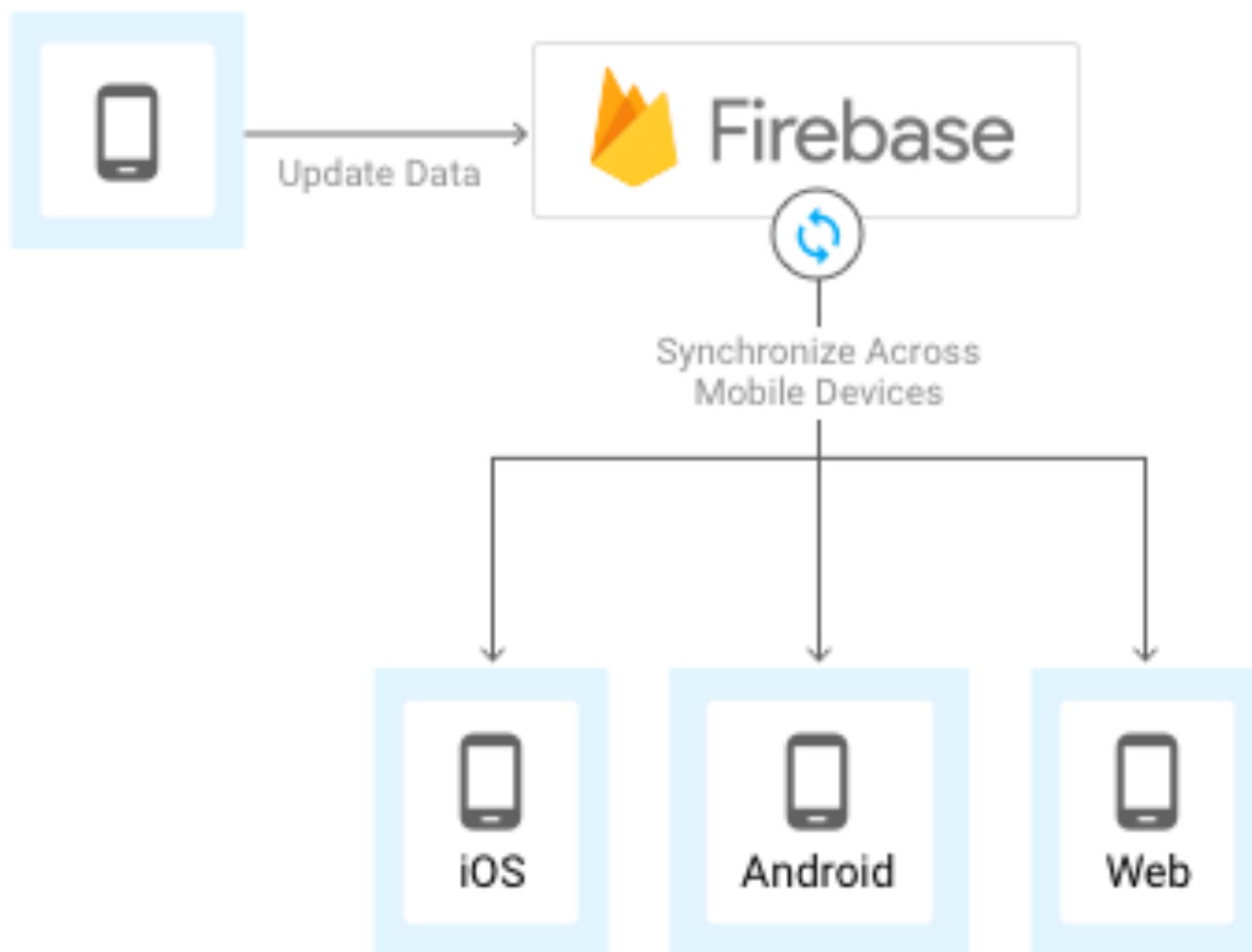
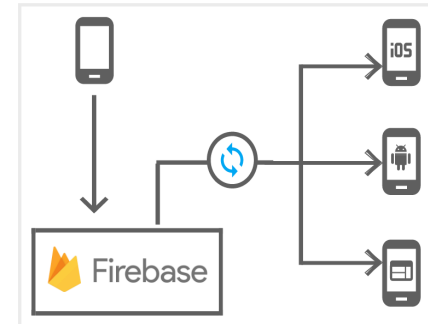


Mobile App Backend Services



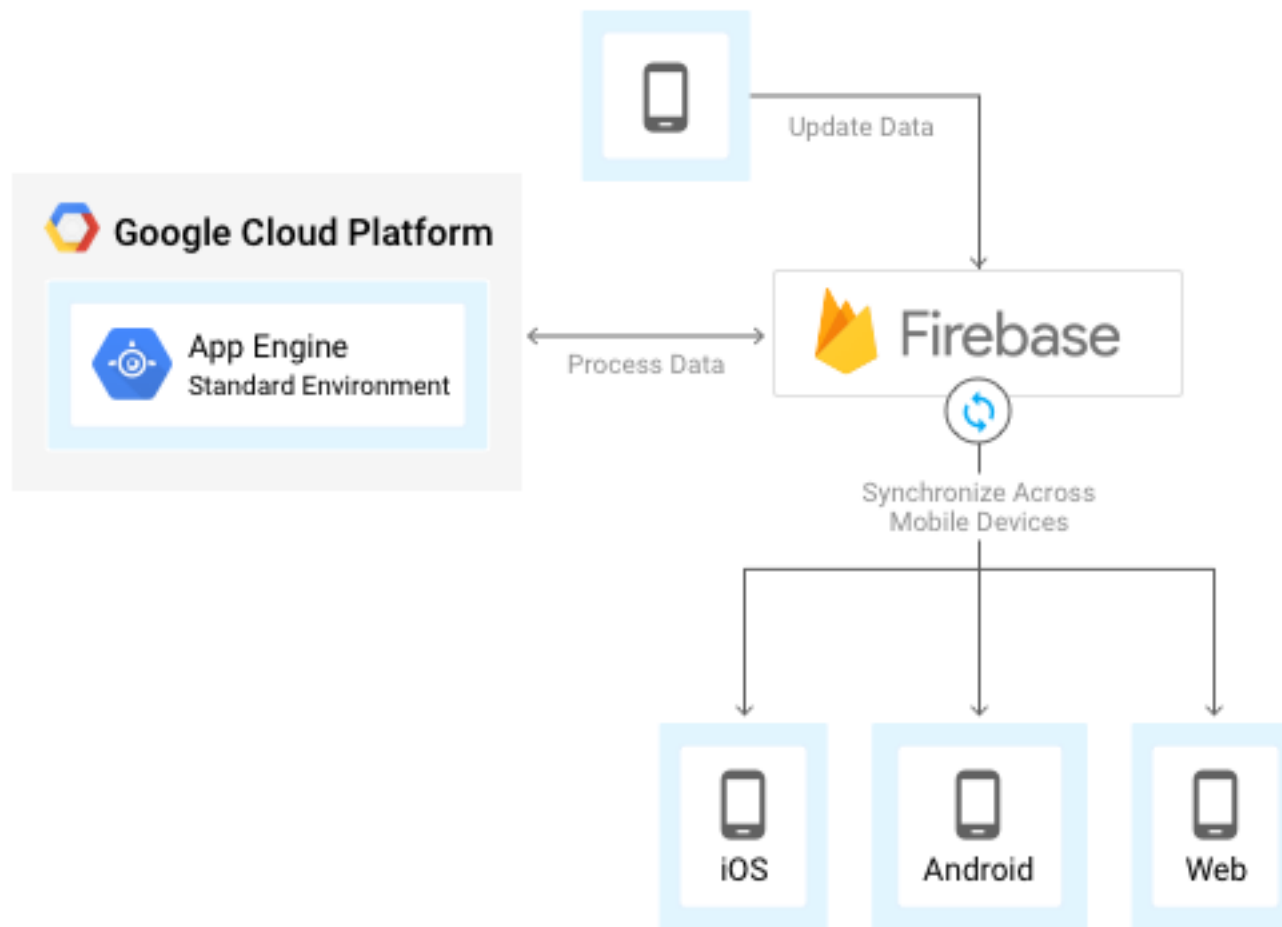
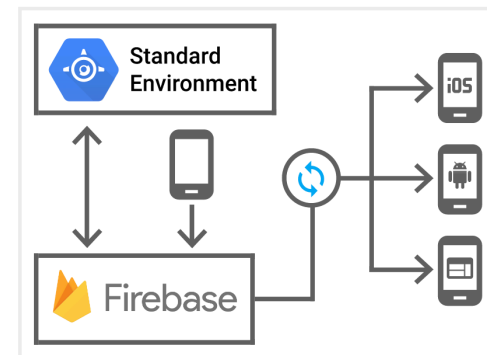


Firestore



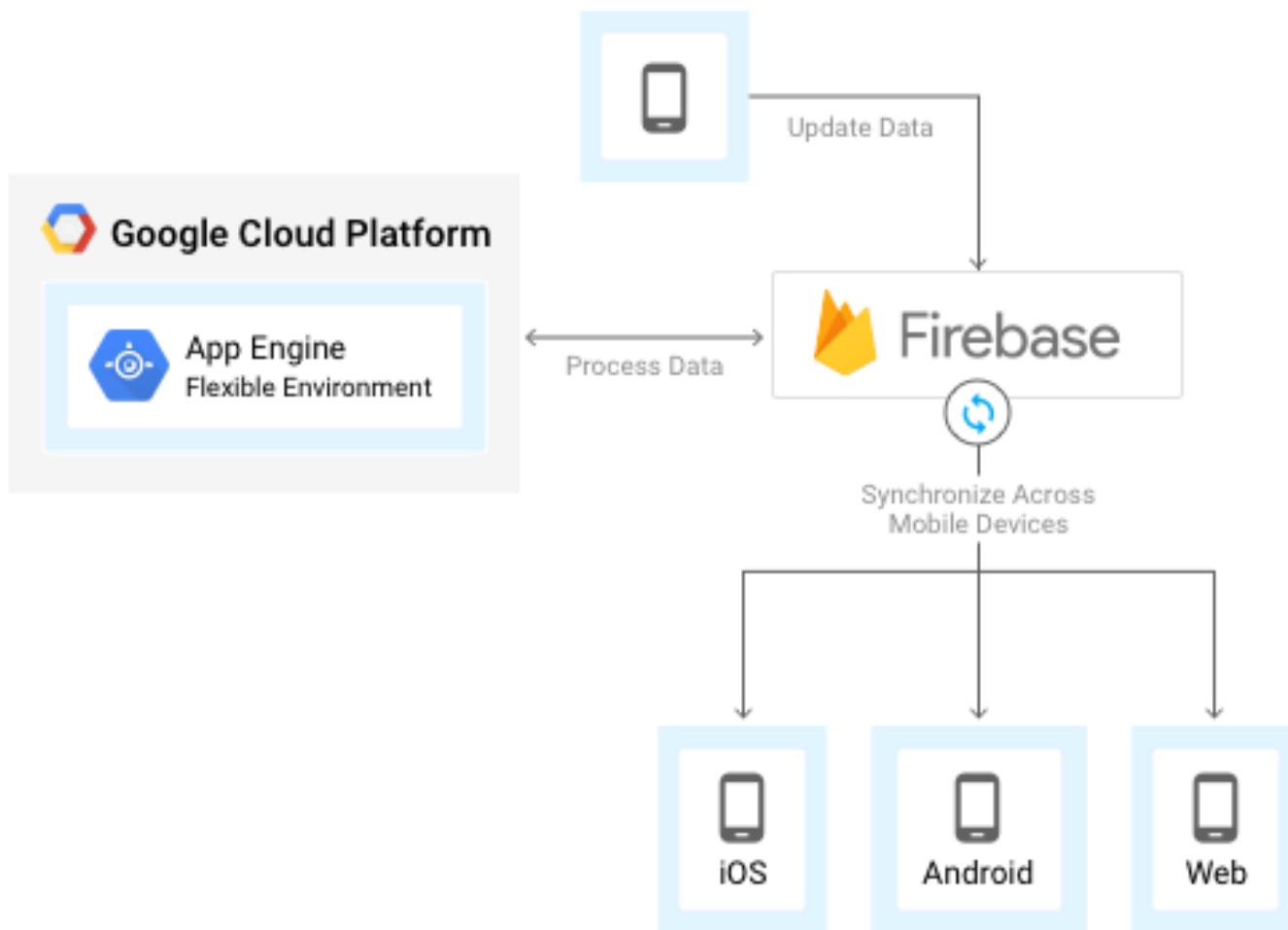
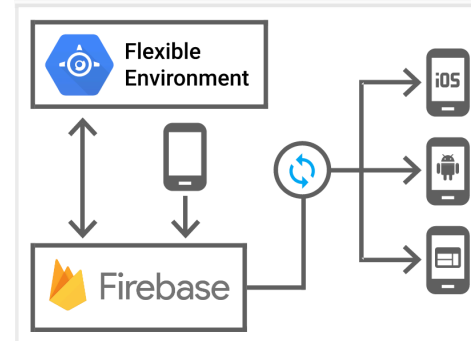


Firestore and Google App Engine standard environment



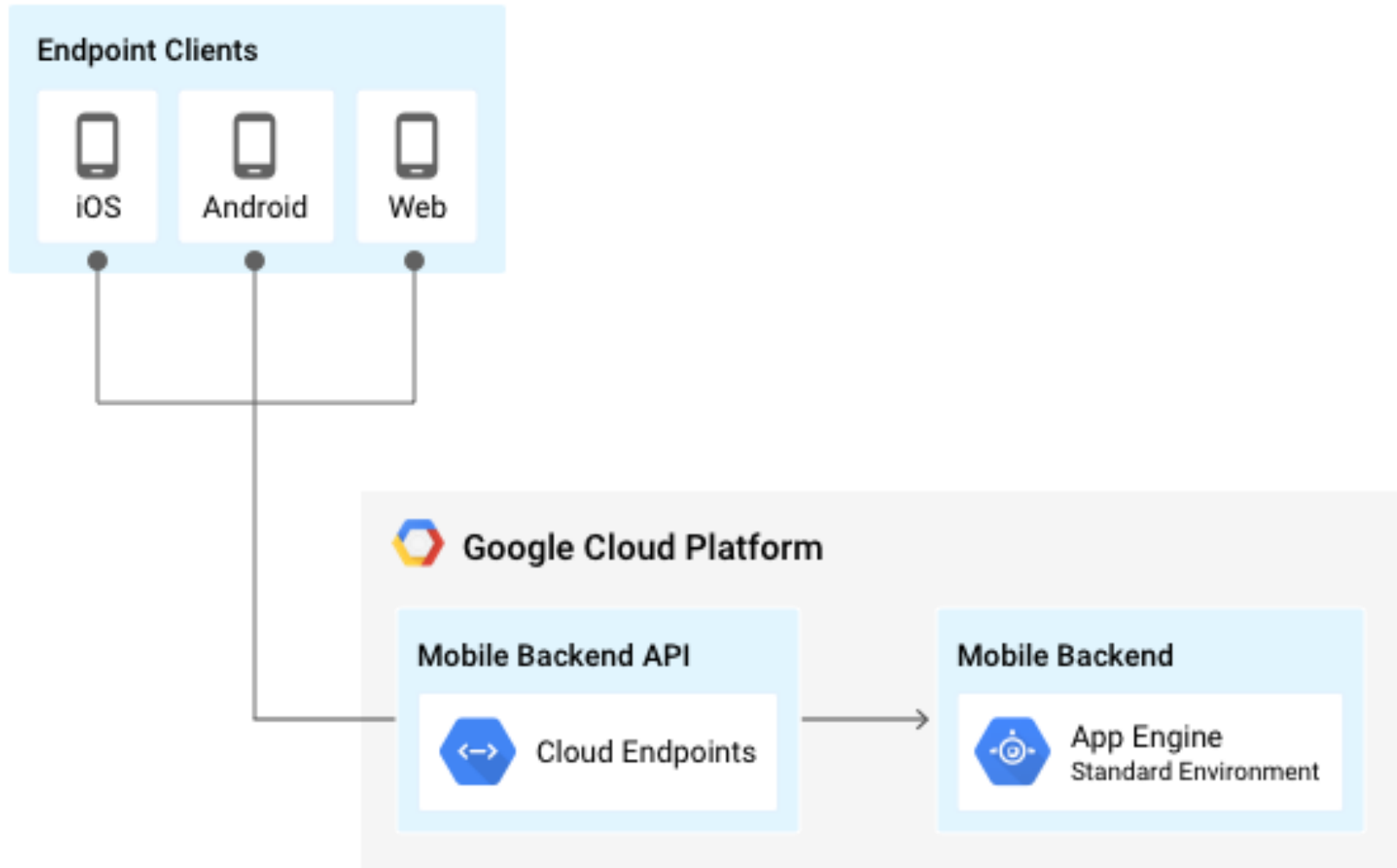
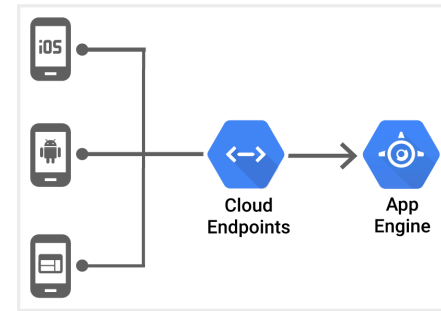


Firestore and App Engine flexible environment



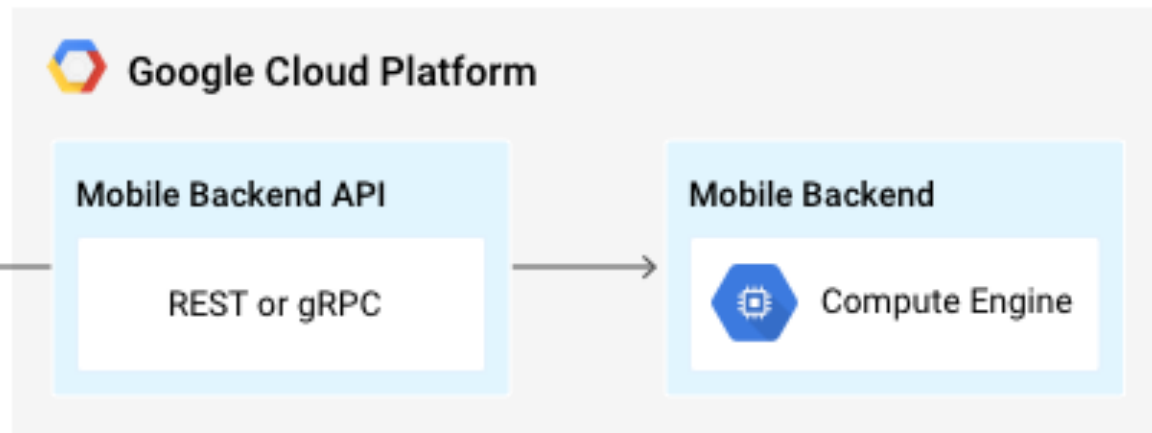
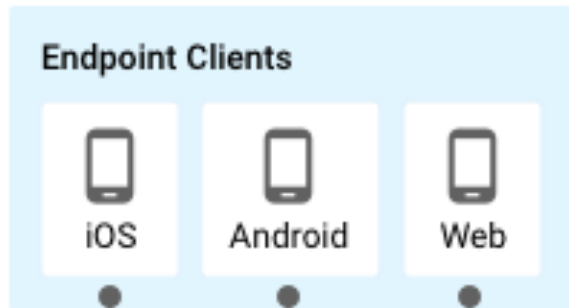
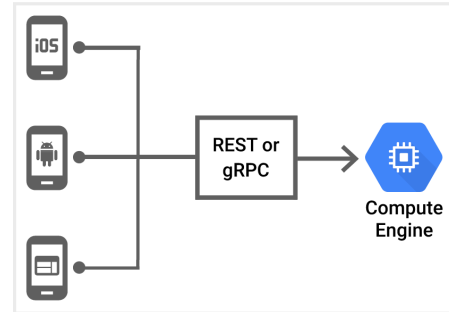


App Engine and Cloud Endpoints





Compute Engine and REST or gRPC





Feature	Firebase	Firebase & App Engine standard environment	Firebase & App Engine flexible environment	App Engine standard environment & Endpoints	Compute Engine & REST/gRPC
Automatic capacity scaling	✓	✓	✓	✓	If you configure an autoscaler.
Automatic real-time data synchronization	✓	✓	✓		
Automatic server maintenance	✓	✓	✓	✓	
Backend logic		✓	✓	✓	✓
Call native binaries, write to the file system, or make other system calls.			✓		✓
Data storage	✓	✓	✓	If you add other Cloud Platform services	If you add other Cloud Platform services
File storage	✓	✓	✓	✓	✓
				with Cloud Storage	with Cloud Storage





Feature	Firebase	Firebase & App Engine standard environment	Firebase & App Engine flexible environment	App Engine standard environment & Endpoints	Compute Engine & REST/gRPC
Easy user authentication	✓	✓	✓	OAuth 2.0	
Language support for backend service logic	N/A	Java, Python, Go, PHP	Any	Java, Python, Go (Cloud Endpoints for Go .)	Any
Messages and notifications, such as push notifications	✓	✓	✓	✓ with Cloud Messaging	✓ with Cloud Messaging
Platform support	iOS, Android, Web	iOS, Android, Web	iOS, Android, Web	iOS, Android, Web	iOS, Android, Web
Requires code to run within a sandbox .	N/A	✓		✓	
Requires SSL		✓		✓	

Google Cloud Platform

<https://cloud.google.com/>

[←](#) [→](#) [↻](#) [🔒 https://cloud.google.com](#) [☆](#)

 **Google Cloud Platform**

[Console](#) 

[Why Google](#) [Products](#) [Solutions](#) [Launcher](#) [Pricing](#) [Customers](#) [Documentation](#) [Support](#) [Partners](#) [Try It Free](#) [Contact Sales](#)

Everything You Need To Build And Scale



Compute

From virtual machines with proven price/performance advantages to a fully managed app development platform.

[Compute Engine](#)

[App Engine](#)

[Container Engine](#)



Storage and Databases

Scalable, resilient, high performance object storage and databases for your applications.

[Cloud Storage](#)

[Cloud SQL](#)

[Cloud Bigtable](#)



Networking

State-of-the-art software-defined networking products on Google's private fiber network.

[Cloud Virtual Network](#)

[Cloud Load Balancing](#)

[Cloud CDN](#)



Big Data

Fully managed data warehousing, batch and stream processing, data exploration, Hadoop/Spark, and reliable messaging.

[BigQuery](#)

[Cloud Dataflow](#)

[Cloud Dataproc](#)



Machine Learning

Fast, scalable, easy to use ML services. Use our pre-trained models or train custom models on your data.

[Cloud Machine Learning Platform](#)

[Jobs API](#)

[Natural Language API](#)



Management Tools

Monitoring, logging, and diagnostics and more, all in an easy to use web management console or mobile app.

[Stackdriver Overview](#)

[Monitoring](#)

[Logging](#)





Google Cloud Platform

**Hosting +
Compute**

Storage

Big Data

Services



Google Cloud Platform

Compute



App Engine



Compute Engine



Container Engine

Storage



Cloud Datastore



Cloud SQL



Cloud Storage

Big Data



Big Query

Services



Cloud Endpoints



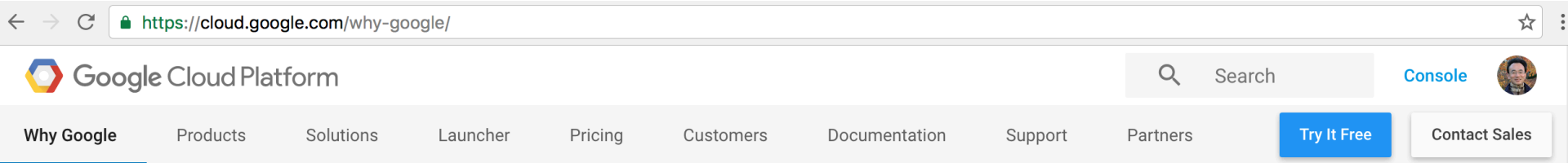
Translate API



Prediction API



Why Google Cloud Platform



What Next Means

A global fiber network, connecting you to the world. Analytics that crunch petabytes in minutes. No-ops services that just scale. Here's a taste of what's next, available now.



No-Ops Apps

Google App Engine lets you build, run, and scale applications without breaking a sweat.

[Learn more](#)



Custom Machine Types

Google Compute Engine, tailored to your needs and budget - by you.

[Learn more](#)



Containers Made Easy

Google Container Engine - the fastest way to implement Kubernetes.

[Learn more](#)



Process Terabytes in Minutes

Google BigQuery: Load, SELECT,



Batch, Stream, or Both

Google Cloud Dataflow delivers high-scale data processing through

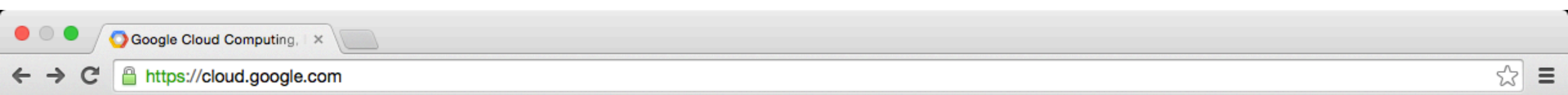


Managed Deep Learning

Google Cloud Machine Learning lets your app interpret images, text,



Why Google Cloud Platform



Why Google



Run on Google's infrastructure

Build on the same [infrastructure](#) that allows Google to return billions of search results in milliseconds, serve 6 billion hours of YouTube video per month and provide storage for 425 million Gmail users.

[Find out more](#)



Mix and match services

Virtual machines. Managed platform. Blob storage. Block storage. NoSQL datastore. MySQL database. Big Data analytics. Google Cloud Platform has all the services your application architecture needs.

[Find out more](#)



Performance you can count on

Every millisecond of latency matters. Google's compute infrastructure gives you consistent CPU, memory and disk performance. Our network and edge cache serve responses rapidly to your users across the world.

[Find out more](#)



Focus on your product

Rapidly develop, deploy and iterate your applications without worrying about system administration. Google manages your application, database and storage servers so you don't have to.

[Find out more](#)



Scale to millions of users

Applications hosted on Cloud Platform can automatically scale up to handle the most demanding Internet-scale workloads and scale down when traffic subsides. You pay only for what you use.

[Find out more](#)



Get the support you need

With our worldwide community of users, partner ecosystem and premium support packages, Google provides a full range of resources to help you get started and grow.

[Find out more](#)

Customers of Google Cloud Platform



Snapchat

"App Engine enabled us to focus on developing the application. We wouldn't have gotten here without the ease of development that App Engine gave us."

Bobby Murphy CTO and co-Founder

[Read Snapchat's story](#)



Webfilings

"Google App Engine has the breadth and the depth to grow with you. Every 6 months, it gets better. The Google App Engine team knows what you need to make a competitive application."

Brett Harper Director of Product Development

[Read Webfilings' story](#)



Khan Academy

"If we didn't have Google App Engine, we'd be spending a lot more time figuring out server setup and working on routers. Our ability to focus on the actual product is the benefit of Google App Engine."

Ben Kamens Lead Developer

[Read Khan Academy's story](#)



Rovio

"Google App Engine allows us to launch games very quickly with teams of one or two developers per game. Because Google manages all the servers, there is little required of us in terms of maintenance."

Stefan Hauk lead server developer for web games

[Read Rovio's story](#)



MAG Interactive

"Our rapid growth to 5M Ruzzle players in less than six months required a highly scalable server solution. Google App Engine transformed this huge challenge into a picnic."

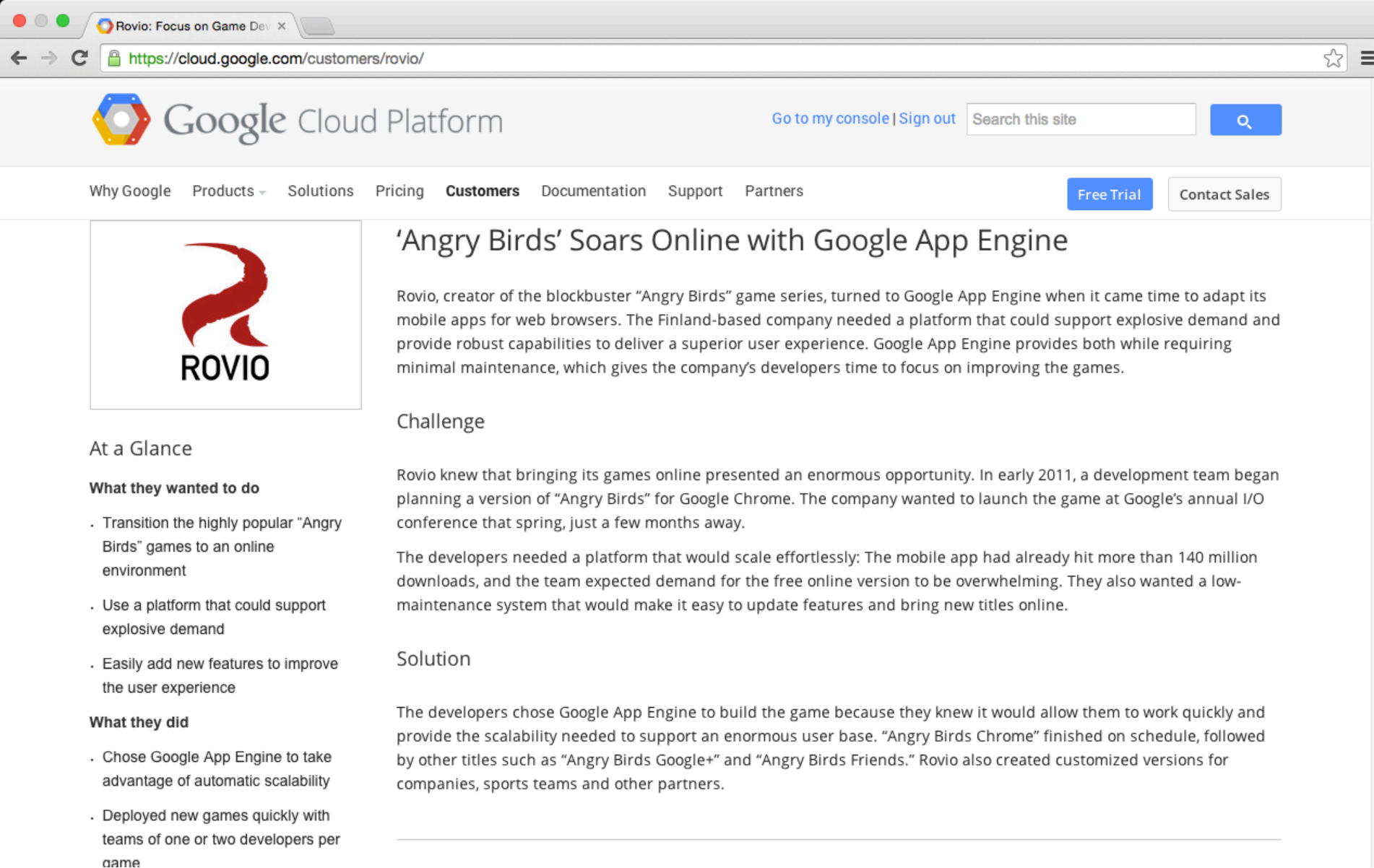


Interactions Marketing


"We are always looking for ways to maximize return and minimize investment. BigQuery is the perfect combination. It's an on-demand, scalable resource."

Source: <https://cloud.google.com/customers/>

Angry Birds



The screenshot shows a web browser window with the URL <https://cloud.google.com/customers/rovio/>. The page header includes the Google Cloud Platform logo, navigation links like 'Go to my console | Sign out', and a search bar. The main navigation bar lists 'Why Google', 'Products', 'Solutions', 'Pricing', 'Customers' (highlighted), 'Documentation', 'Support', and 'Partners'. There are also buttons for 'Free Trial' and 'Contact Sales'.



ROVIO

'Angry Birds' Soars Online with Google App Engine

Rovio, creator of the blockbuster "Angry Birds" game series, turned to Google App Engine when it came time to adapt its mobile apps for web browsers. The Finland-based company needed a platform that could support explosive demand and provide robust capabilities to deliver a superior user experience. Google App Engine provides both while requiring minimal maintenance, which gives the company's developers time to focus on improving the games.

Challenge

Rovio knew that bringing its games online presented an enormous opportunity. In early 2011, a development team began planning a version of "Angry Birds" for Google Chrome. The company wanted to launch the game at Google's annual I/O conference that spring, just a few months away.

The developers needed a platform that would scale effortlessly: The mobile app had already hit more than 140 million downloads, and the team expected demand for the free online version to be overwhelming. They also wanted a low-maintenance system that would make it easy to update features and bring new titles online.

Solution

The developers chose Google App Engine to build the game because they knew it would allow them to work quickly and provide the scalability needed to support an enormous user base. "Angry Birds Chrome" finished on schedule, followed by other titles such as "Angry Birds Google+" and "Angry Birds Friends." Rovio also created customized versions for companies, sports teams and other partners.

At a Glance

What they wanted to do

- Transition the highly popular "Angry Birds" games to an online environment
- Use a platform that could support explosive demand
- Easily add new features to improve the user experience

What they did

- Chose Google App Engine to take advantage of automatic scalability
- Deployed new games quickly with teams of one or two developers per game

Source: <https://cloud.google.com/customers/>



Compute



Compute Engine

Run large-scale workloads on virtual machines hosted on Google's infrastructure



App Engine

A platform for building scalable web apps and mobile backends



Container Engine

Run Docker containers on Google's infrastructure, powered by Kubernetes



Container Registry

Fast, private Docker image storage on Google Cloud Platform



Cloud Functions ALPHA

A serverless platform for building event-based microservices



Storage and Databases



Cloud Storage

Powerful, simple and cost effective object storage service with global edge-caching



Cloud SQL

Store and manage data using a fully-managed, relational MySQL database



Cloud Bigtable

Cloud Bigtable is a fast, fully managed, massively scalable NoSQL database service



Cloud Datastore

A managed, NoSQL, schemaless database for storing non-relational data



Persistent Disk

Reliable, high-performance block storage for virtual machine instances



Big Data



BigQuery

A fast, economical and fully managed data warehouse for large-scale data analytics



Cloud Dataflow

Cloud Dataflow is a real-time data processing service for batch and stream data processing



Cloud Dataproc

Cloud Dataproc is a managed Spark and Hadoop service that is fast, easy to use, and low cost



Cloud Datalab BETA

An easy to use interactive tool for large-scale data exploration, analysis and visualization



Cloud Pub/Sub

Connect your services with reliable, many-to-many, asynchronous messaging hosted on Google's infrastructure



Genomics

Power your science with Google Genomics



Machine Learning



Cloud Machine Learning Services BETA

Fast, large scale and easy to use Machine Learning services



Jobs API ALPHA

Powerful job search and discovery built for the talent industry



Natural Language API

Derive insights from unstructured text using Google machine learning



Speech API BETA

Speech to text conversion powered by machine learning



Translation API

Dynamically translate between thousands of available language pairs



Vision API

Derive insight from images with our powerful Cloud Vision API



Management Tools



Cloud Endpoints BETA

Create RESTful services from your code and make them accessible to iOS, Android, and Javascript clients using App Engine



Cloud Console

Your integrated Google Cloud Platform management console



Cloud Shell

Manage your infrastructure and applications from the command-line in any browser



Cloud Mobile App

Manage Google Cloud Platform services from your Android or iOS device



Billing API

The Google Cloud Billing API provides methods that you can use to programmatically manage billing for your projects in the Google Cloud Platform



Cloud APIs

Programmatic interfaces for all Google Cloud Platform services



Developer Tools



Cloud SDK

Command-line interface for Google Cloud Platform products and services



Deployment Manager

Create and manage cloud resources with simple templates



Cloud Source Repositories **BETA**

Fully-featured private Git repositories hosted on Google Cloud Platform



Cloud Tools for Android Studio

Build backend services on Google Cloud Platform for your Android apps



Cloud Tools for IntelliJ

Debug production cloud applications right inside of IntelliJ



Cloud Tools for PowerShell **BETA**

Full cloud control from Windows PowerShell



Cloud Tools for Visual Studio **BETA**

Deploy Visual Studio applications to Google Cloud Platform



Google Plugin for Eclipse

Simplifies development of GWT and App Engine projects in the Eclipse IDE



Cloud Test Lab

On-demand app testing with the convenience and scalability of a cloud service

Compute



Compute Engine

Compute Engine is Google's Infrastructure-as-a-Service (IaaS). Run large-scale workloads on virtual machines hosted on Google's infrastructure. Choose a VM that fits your needs and gain the performance and consistency of Google's worldwide fiber network. With per-minute billing, you pay only for what you use.



App Engine

App Engine is Google's Platform-as-a-Service (PaaS). Develop your application easily using built-in services that make you more productive. Deploy to a fully-managed platform and let Google carry the pager. Just download the SDK and start building immediately for free with no credit card required.



Container Engine

Container Engine makes it easy to run Docker containers on Google Cloud Platform. Powered by Kubernetes, Container Engine takes care of provisioning and maintaining the underlying virtual machine cluster, scaling your application, and operational logistics like logging, monitoring, and health management.

Storage



Cloud Datastore

Cloud Datastore provides a managed, NoSQL, schemaless database for storing non-relational data. Cloud Datastore automatically scales as you need it and supports transactions as well as robust, SQL-like queries.



Cloud SQL

Store and manage data using a fully-managed, relational MySQL database. Google handles replication, patch management and database management to ensure availability and performance.



Cloud Storage

Use a durable and highly available object storage service. With global edge-caching, your users have fast access to your app's data from any location. Google manages versioning, guarantees a strong SLA and provides a simple API that allows you to manage your data programmatically.



BigQuery

- Analyze Big Data in the cloud with BigQuery.
- Run fast, SQL-like queries against multi-terabyte datasets in seconds.
- Scalable and easy to use, BigQuery gives you real-time insights about your data

Services



Cloud Pub/Sub

Connect your services with reliable, many-to-many, asynchronous messaging hosted on Google's infrastructure. Cloud Pub/Sub automatically scales as you need it and provides a foundation for building your own robust, global services.



Cloud DNS

Reliable, resilient, low-latency DNS serving from Google's worldwide network of Anycast DNS servers. Create DNS records with an easy to use command line interface, or program against a full featured RESTful API to customize the service to your specific needs.



Cloud Endpoints

Create RESTful services from your code and make them accessible to iOS, Android and Javascript clients. Automatically generate client libraries to make wiring up the frontend easy. Built-in critical infrastructure includes denial-of-service protection, OAuth 2.0 support and client key management.



Translate API

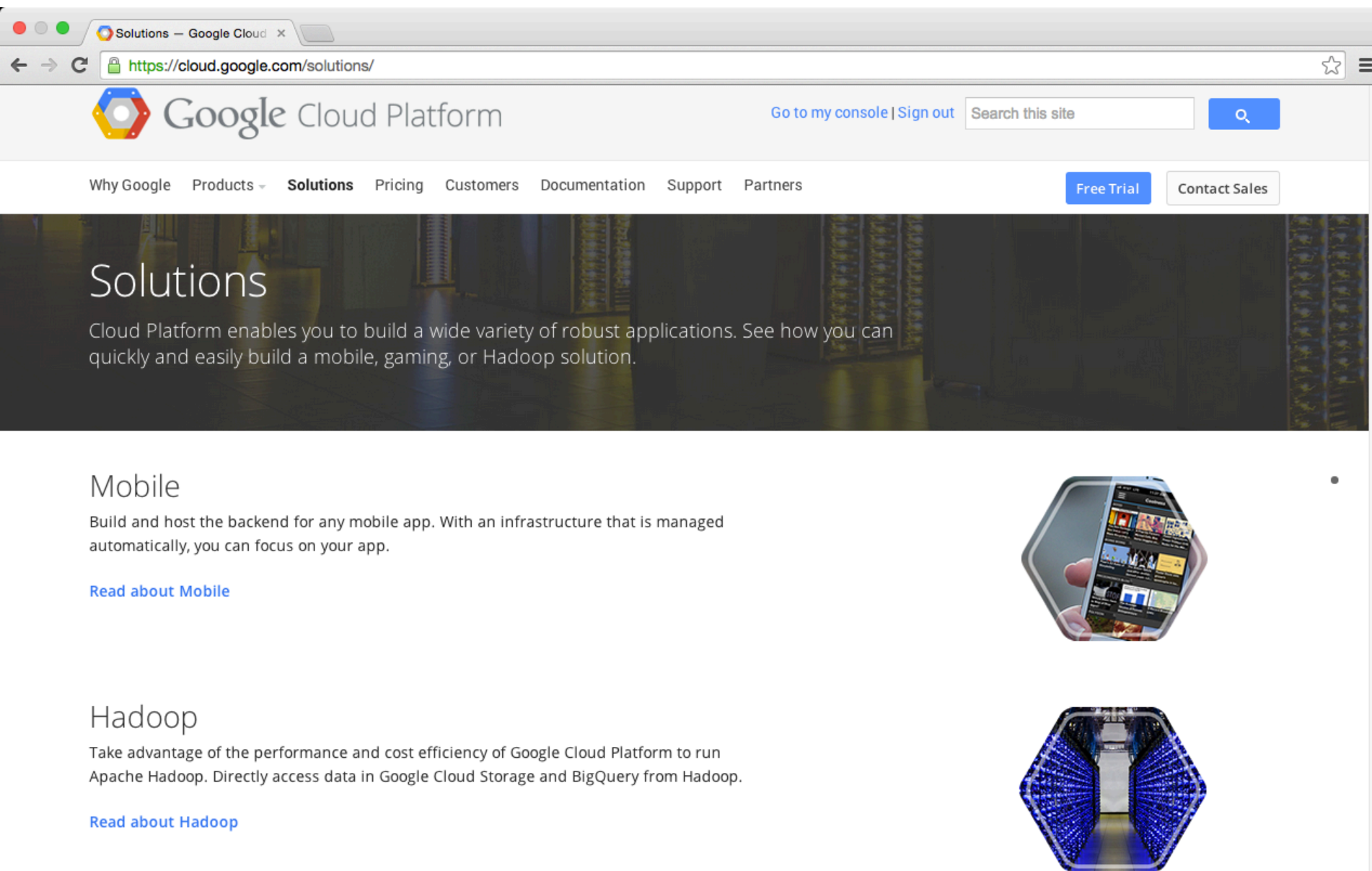
Create multilingual apps and translate text into other languages programmatically. Thousands of language pairs are available.



Prediction API

- Use Google's machine learning algorithms to analyze data and predict future outcomes using a familiar RESTful interface.

Google Cloud Platform Solutions



The image is a screenshot of a web browser displaying the Google Cloud Platform Solutions page. The browser's address bar shows the URL <https://cloud.google.com/solutions/>. The page header features the Google Cloud Platform logo, navigation links for 'Go to my console', 'Sign out', and a search bar. A secondary navigation bar includes links for 'Why Google', 'Products', 'Solutions' (which is highlighted), 'Pricing', 'Customers', 'Documentation', 'Support', and 'Partners'. There are also buttons for 'Free Trial' and 'Contact Sales'. The main content area has a dark background with server racks and the heading 'Solutions'. Below this, there are two sections: 'Mobile' and 'Hadoop'. The 'Mobile' section describes building and hosting mobile app backends and includes a link to 'Read about Mobile'. The 'Hadoop' section describes leveraging Google Cloud Platform for running Apache Hadoop and includes a link to 'Read about Hadoop'. Each section is accompanied by a hexagonal image: a hand holding a smartphone for Mobile, and a server room for Hadoop.

Solutions

Cloud Platform enables you to build a wide variety of robust applications. See how you can quickly and easily build a mobile, gaming, or Hadoop solution.

Mobile

Build and host the backend for any mobile app. With an infrastructure that is managed automatically, you can focus on your app.


[Read about Mobile](#)

Hadoop


Take advantage of the performance and cost efficiency of Google Cloud Platform to run Apache Hadoop. Directly access data in Google Cloud Storage and BigQuery from Hadoop.

[Read about Hadoop](#)

Try Google Cloud Platform for Free

 Google Cloud Platform

Search this site

My console 

Why Google Products Solutions Launcher Pricing Customers Documentation Support Partners


Free Trial Contact Sales

Try Google Cloud Platform for Free

Get \$300 in credit to use for 60 days.
This trial is absolutely free and you will not be billed unless you decide to upgrade to a paid account.


- ✓ Platform as a service
- ✓ Virtual machines
- ✓ Big data solutions
- ✓ SQL and NoSQL databases
- ✓ Object storage
- ✓ Application services

Why Sign Up?




\$300 Credit for Free

Sign up and get \$300 to spend on Google Cloud Platform over the next 60 days and discover the power of our products.



Access to Google Cloud Platform Products

Test drive virtual machines, autoscaling web applications, databases, object storage, network and DNS services, big data ETL and real-time queries, translation, machine learning, and more.

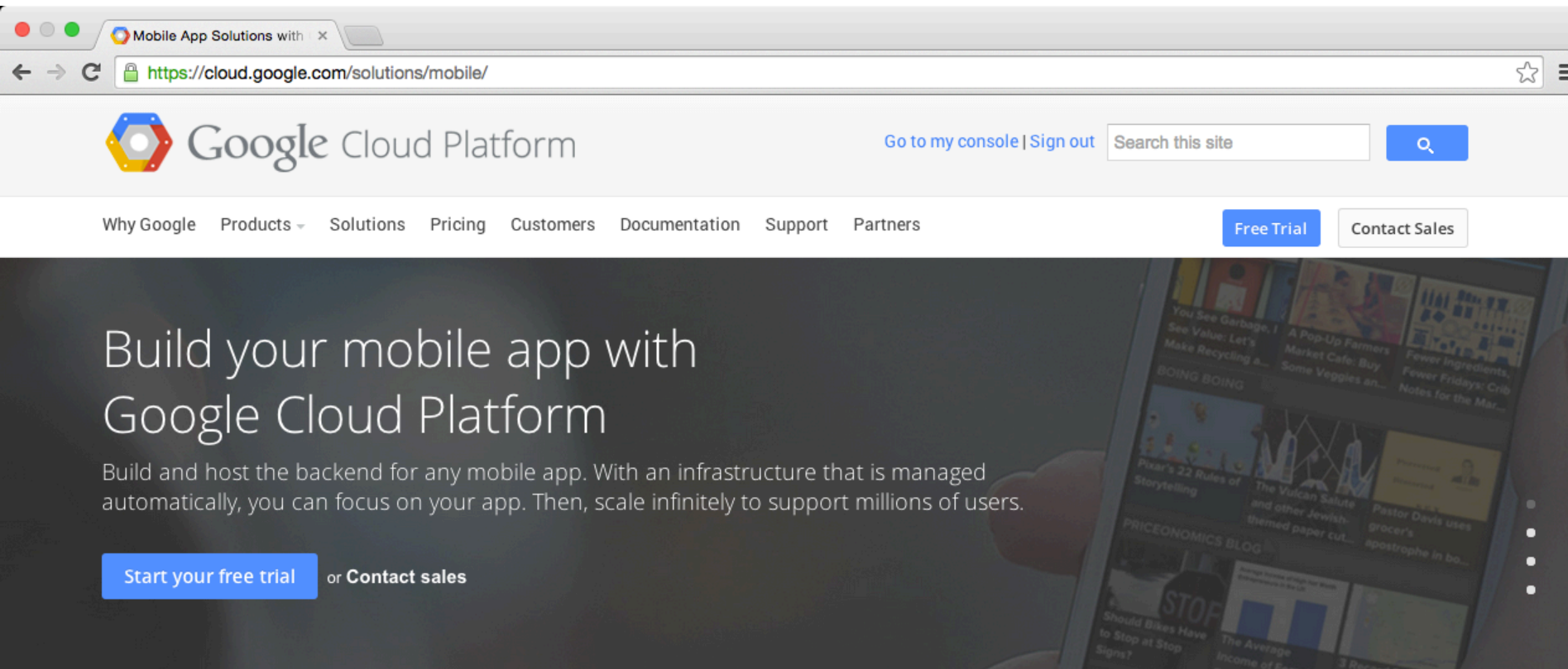


You Won't be Billed

Why Sign Up

- FAQ
- Resources

Mobile App with Google Cloud Platform

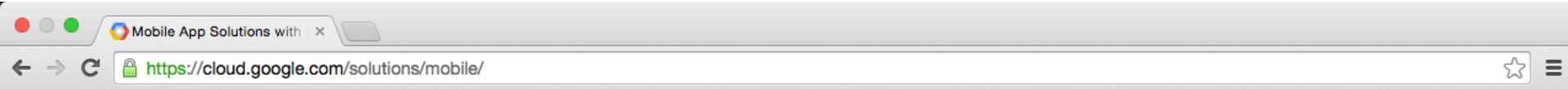


Deploy an app in 300 seconds

See how easy it is to develop a mobile backend using custom code with Google Cloud Endpoints. Automatically generated strongly-typed client libraries for iOS, Android, and Javascript make calling backend APIs simple.

Benefits of building mobile apps on Google Cloud Platform:

Mobile App with Google Cloud Platform



Deploy an app in 300 seconds

See how easy it is to develop a mobile backend using custom code with Google Cloud Endpoints. Automatically generated strongly-typed client libraries for iOS, Android, and Javascript make calling backend APIs simple.

Benefits of building mobile apps on Google Cloud Platform:



Automatic scaling and load balancing

All you have to do is write your application code and we'll handle the rest. No matter how many users you have, App Engine will scale to your needs.



Push notifications

Use Google Cloud Messaging to send data to your Android users and the Sockets API to send push notifications to your iOS users.



Geo-proximity search

Leverage App Engine's Search API for geo-proximity search.



Data processing

Application logs can provide insight into app usage and backend behavior. BigQuery enables you to gather real time ad-hoc analysis of your log files, analyzing terabytes of data in seconds.



Static content serving

App Engine serves static files from dedicated servers and caches that are separate from the application servers.



Data storage

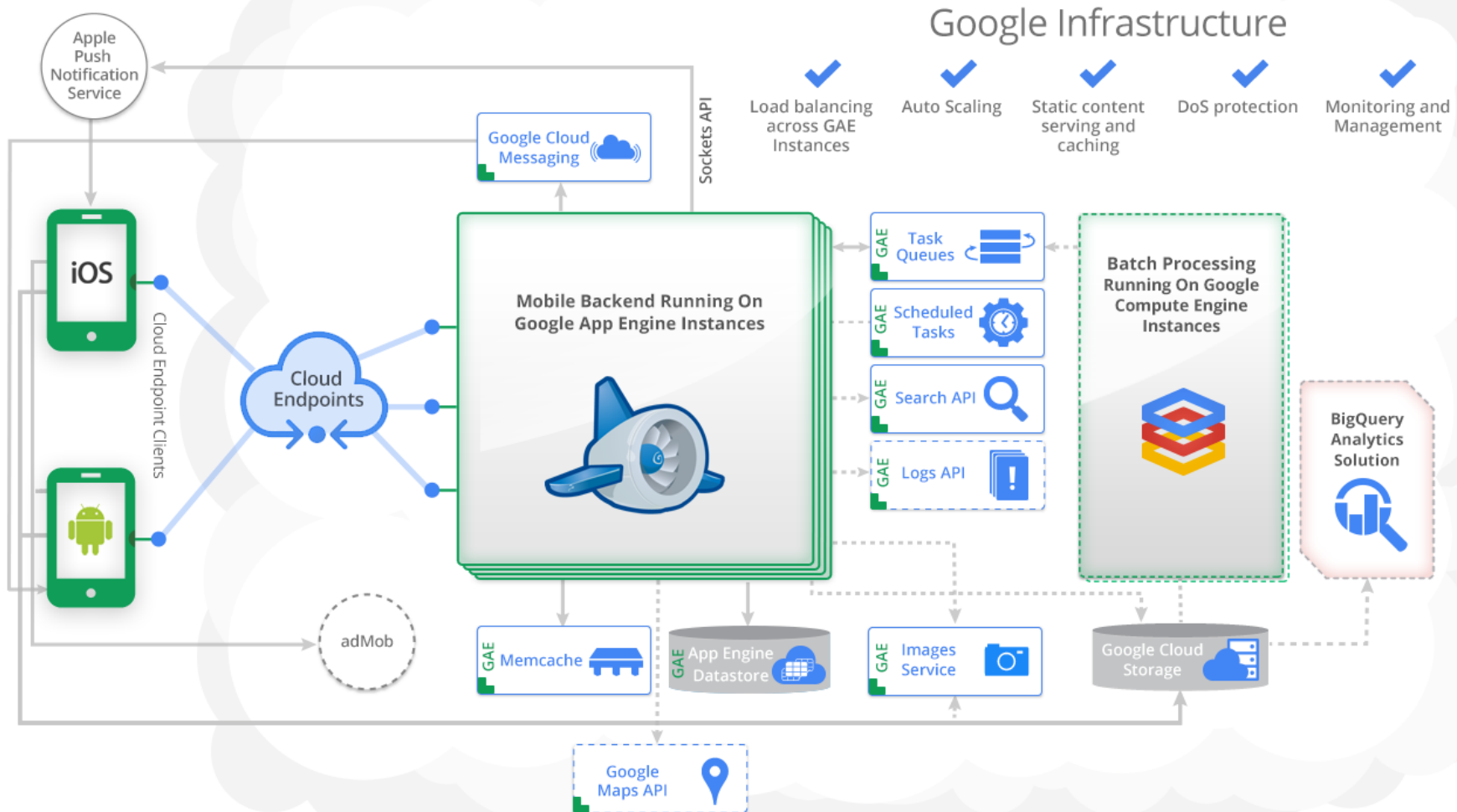
Google Cloud Storage lets you store objects and files up to terabytes in size using a simple REST based interface.

Mobile Solutions on the Google Cloud Platform

 Your Application Code running on Google App Engine (GAE), Google Compute Engine (GCE), and Client Devices

 Google Cloud Platform Services  Optional components

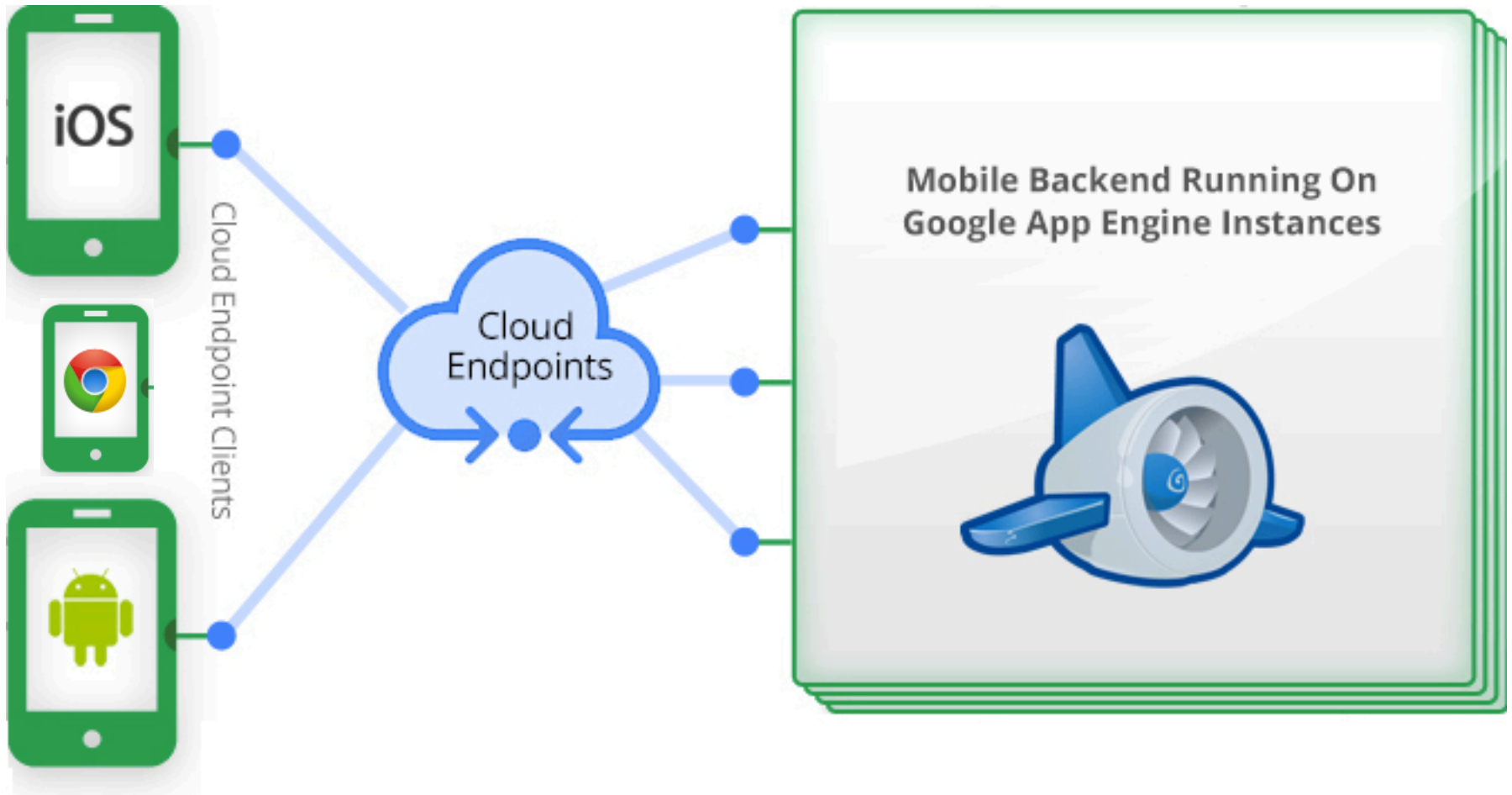
 Capabilities Included



Mobile Solution on Google Cloud Platform

- Support for Android and iOS devices through native applications
- Storage, retrieval, and processing data outside of mobile devices
- Orchestrating push notification to Android and IOS devices
- Geo-location awareness and geo-proximity search
- User authentication
- High scalability

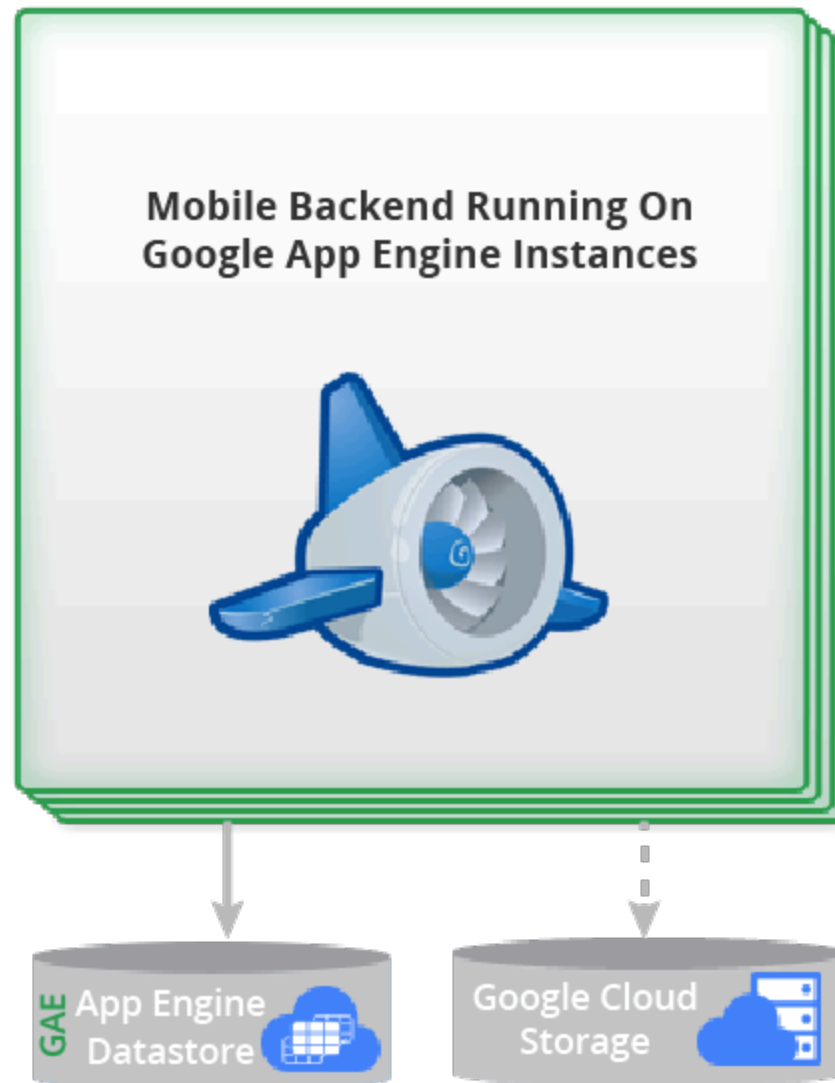
Mobile App Solutions Architecture



Mobile App Solutions Architecture

- Android and/or iOS mobile clients.
- Google Cloud Endpoints used for communications between the clients and the backend over REST API with optional OAuth2 authentication.
- Your mobile backend application code running on Google App Engine and responsible for serving requests from the clients.

Storing data

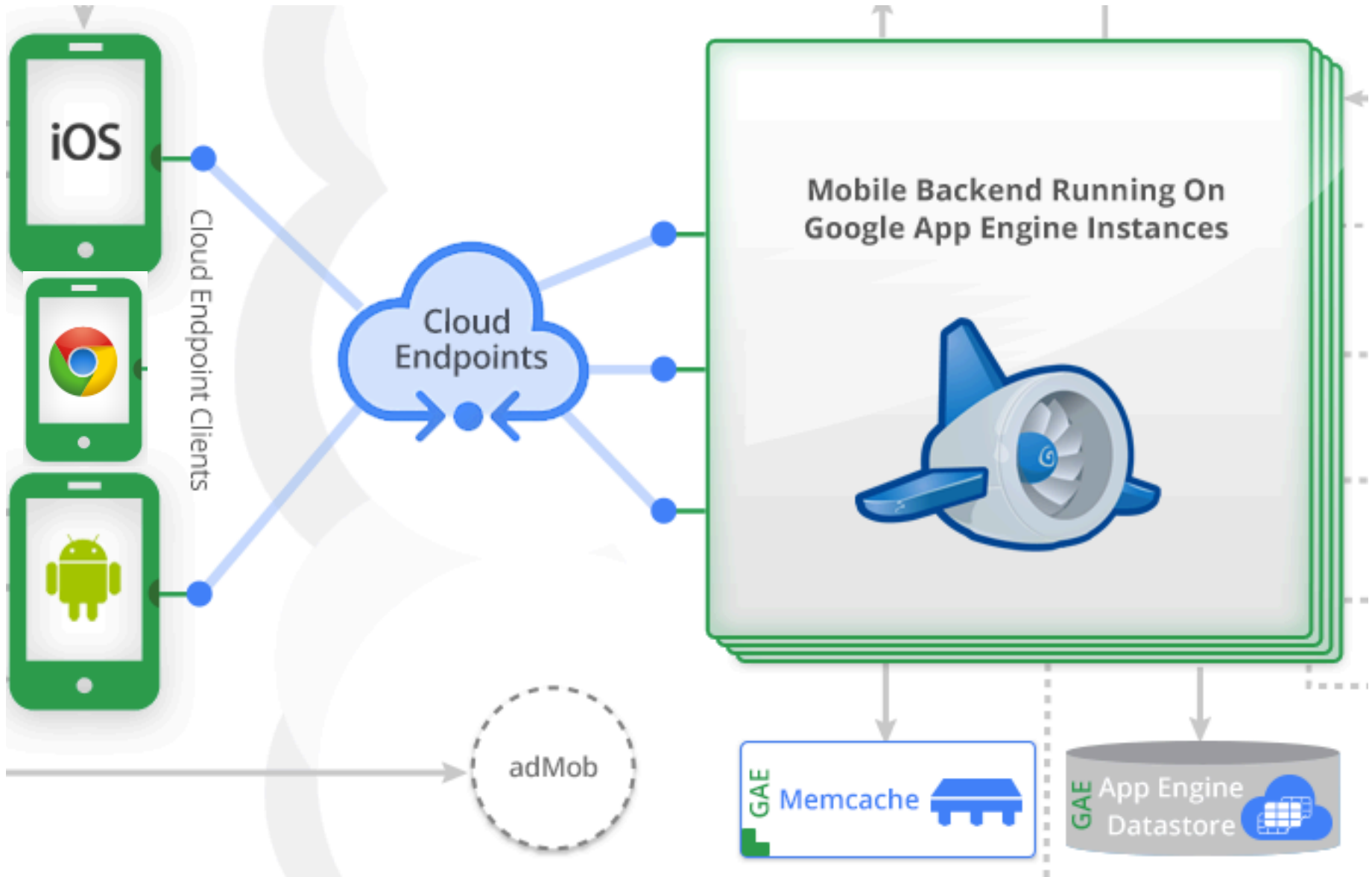


Google App Engine

Datastore Quotas and limits

Limit	Amount
Maximum entity size	1 megabyte
Maximum transaction size	10 megabytes
Maximum number of index entries for an entity	20000
Maximum number of bytes in composite indexes for an entity	2 megabytes

Optimizing data access with Memcache





Google App Engine Platform as a Service (PaaS)

build and run applications on
Google's infrastructure



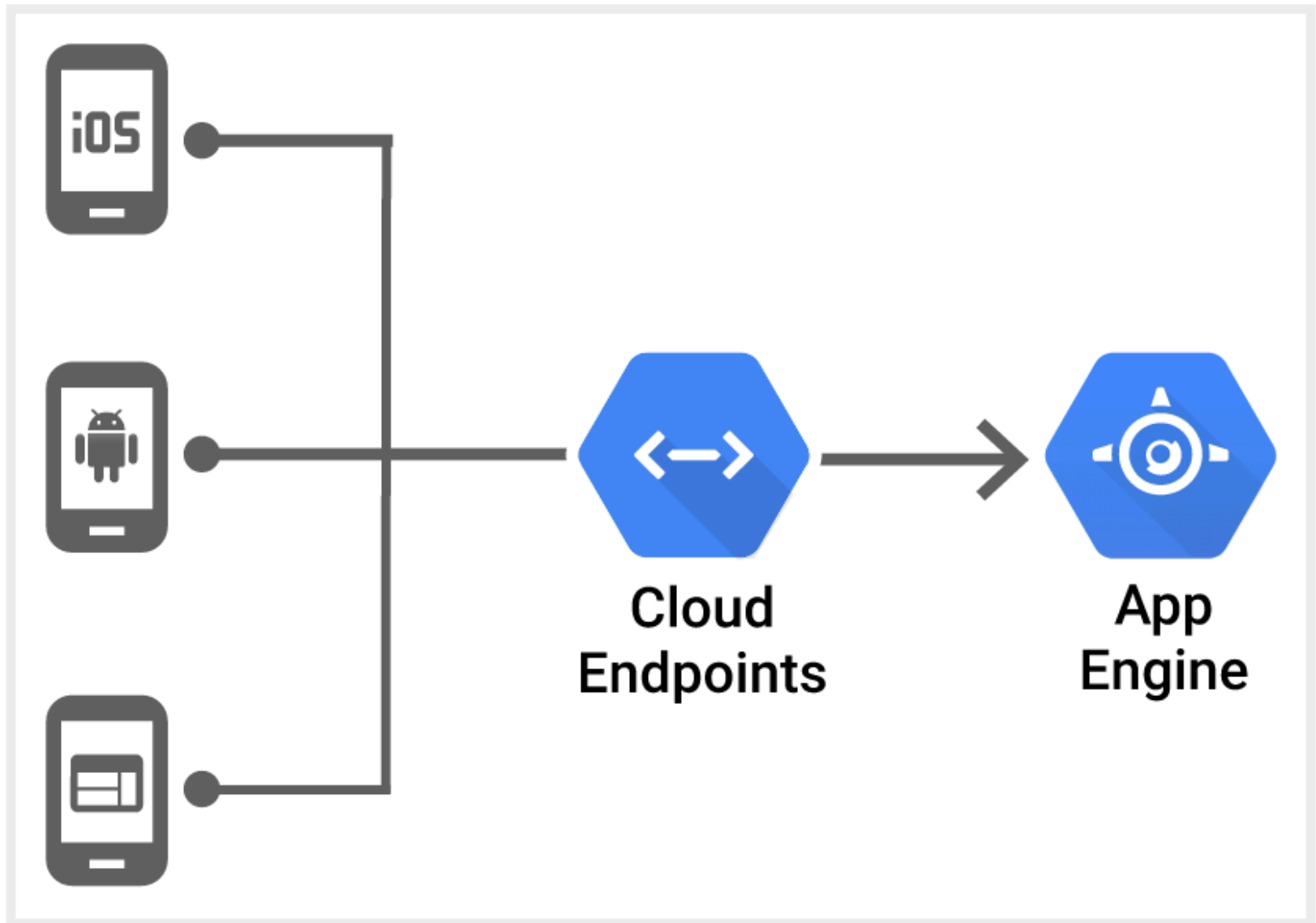
Google App Engine Platform as a Service (PaaS)



Google Compute Engine Infrastructure as a Service (IaaS)

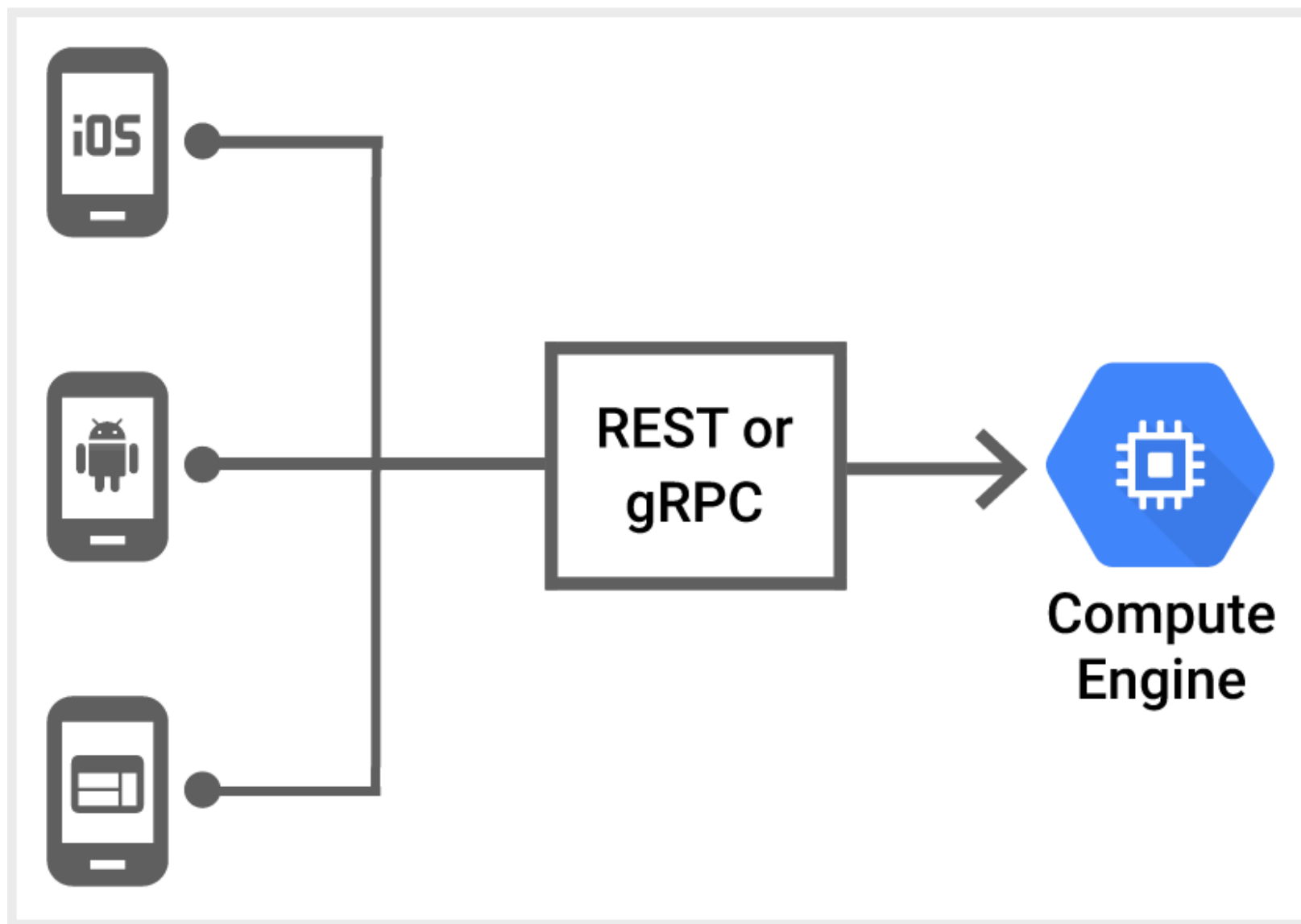


Mobile App Backend Services





Mobile App Backend Services



Google App Engine

- 1 GB of data storage and traffic for free
 - can be increased by enabling paid applications

Google App Engine

Google App Engine supports apps written in a variety of programming languages: Python, Java, PHP, Go



Python



Java



PHP



Go



Google App Engine

The screenshot shows the Google App Engine homepage in a web browser. The browser's address bar displays the URL <https://cloud.google.com/appengine/>. The page header features the Google Cloud Platform logo, navigation links for 'Go to my console' and 'Sign out', a search bar, and buttons for 'Free Trial' and 'Contact Sales'. The main content area has a background image of server racks and includes the App Engine logo, a descriptive paragraph about PaaS, and a 'Start your free trial' button. A secondary navigation bar at the bottom of the main section lists 'Features', 'Case Studies', 'Pricing Calculator', 'Pricing', and 'Documentation'.

Features



Popular languages and frameworks

Write applications in some of the most popular programming languages: Python, Java, PHP and Go. Use existing frameworks such as Django, Flask, Spring and webapp2. Develop locally with



Focus on your code

Let Google worry about database administration, server configuration, sharding and load balancing. With Traffic Splitting, you can A/B test different live versions of your app. Multitenancy support lets you compartmentalize your application data.



Multiple storage options

Choose the storage option you need: a traditional MySQL database using Cloud SQL, a schemaless NoSQL datastore, or object storage using Cloud Storage.

<https://cloud.google.com/appengine/>



Google App Engine

Features



Popular languages and frameworks

Write applications in some of the most popular programming languages: Python, Java, PHP and Go. Use existing frameworks such as Django, Flask, Spring and webapp2. Develop locally with language-specific SDKs. Pair your applications with Compute Engine to integrate other familiar technologies such as Node.js, C++, Scala, Hadoop, MongoDB, Redis and more.



Powerful built-in services

App Engine makes you more productive by eliminating the need to write boilerplate code. Managed services, such as Task Queues, Memcache and the Users API, let you build any application.



Focus on your code

Let Google worry about database administration, server configuration, sharding and load balancing. With Traffic Splitting, you can A/B test different live versions of your app. Multitenancy support lets you compartmentalize your application data.



Familiar development tools

Use the tools you know, including Eclipse, IntelliJ, Maven, Git, Jenkins, PyCharm and more. The App Engine SDK allows you to test applications locally in a simulated environment and then deploy your app with simple command-line tools or the desktop launcher.



Multiple storage options

Choose the storage option you need: a traditional MySQL database using Cloud SQL, a schemaless NoSQL datastore, or object storage using Cloud Storage.



Deploy at Google scale

Some of the world's most popular web services are built on our platform. You can scale up to 7 billion requests per day and automatically scale down when traffic subsides.



Google Cloud Datastore

The screenshot shows the Google Cloud Datastore website. At the top, there's a navigation bar with the Google Cloud Platform logo, a search bar, and links to 'Go to my console' and 'Sign out'. Below this is a secondary navigation bar with links for 'Why Google', 'Products', 'Solutions', 'Pricing', 'Customers', 'Documentation', 'Support', and 'Partners'. There are also 'Free Trial' and 'Contact Sales' buttons. The main hero section features the 'Cloud Datastore' title, a brief description of the service, and a 'Start your free trial' button. At the bottom of the hero section, there are links for 'Features', 'Pricing Calculator', 'Pricing', and 'Documentation'.

Cloud Datastore - NoSQL

<https://cloud.google.com/datastore/>

Go to my console | Sign out

Search this site

Why Google Products Solutions Pricing Customers Documentation Support Partners

Free Trial Contact Sales

Cloud Datastore

Use a managed, NoSQL, schemaless database for storing non-relational data. Cloud Datastore automatically scales as you need it and supports transactions as well as robust, SQL-like queries.

Start your free trial

Features Pricing Calculator Pricing Documentation

Features



Schemaless access, with SQL-like querying

No need to worry about data models and migration. Cloud Datastore is a schemaless storage service that allows you to be agile by removing the need to think about the underlying structure of the



Managed database

Cloud Datastore is fully managed. Google automatically handles sharding and replication in order to provide you with a highly available and consistent database.



Autoscale with your users

Cloud Datastore automatically scales depending on your needs. This allows you to focus on building your application and not on worrying about provisioning and load anticipation.

<https://cloud.google.com/datastore/>



Google Cloud Datastore



Schemaless access, with SQL-like querying

No need to worry about data models and migration. Cloud Datastore is a schemaless storage service that allows you to be agile by removing the need to think about the underlying structure of the data. Cloud Datastore provides a [robust query engine](#) that allows you to search for data across multiple properties and sort as needed.



Managed database

Cloud Datastore is fully managed. Google automatically handles sharding and replication in order to provide you with a highly available and consistent database.



Autoscale with your users

Cloud Datastore automatically scales depending on your needs. This allows you to focus on building your application and not on worrying about provisioning and load anticipation.



ACID transactions

Cloud Datastore provides ACID transactions using optimistic concurrency control. Your application can execute multiple datastore operations in a single transaction in which either all succeed or all fail, ensuring the integrity of your data.



Built-in redundancy

With a single API call, data is automatically replicated across multiple data centers. High availability and durability are built in from the very core.



Local development tools

With the [Cloud Datastore Development Kit](#), you can develop, test and iterate your applications locally without doing full deployments.



Access your data from anywhere

Build solutions that span App Engine and Compute Engine, and rely on Cloud Datastore as the integration point. With the [RESTful interface](#) that is exposed by Cloud Datastore, data can easily be accessed by any deployment target.

<https://cloud.google.com/datastore/>



Google Cloud Endpoints

The screenshot shows the Google Cloud Endpoints website. At the top, there's a navigation bar with the Google Cloud Platform logo, a search bar, and links like 'Go to my console | Sign out'. Below this is a secondary navigation bar with links for 'Why Google', 'Products', 'Solutions', 'Pricing', 'Customers', 'Documentation', 'Support', and 'Partners'. The main hero section features the Cloud Endpoints logo and a description: 'Create RESTful services and make them accessible to iOS, Android and Javascript clients. Automatically generate client libraries to make wiring up the frontend easy. Built-in features include denial-of-service protection, OAuth 2.0 support and client key management.' A 'Start your free trial' button is prominently displayed. At the bottom of the hero section, there are links for 'Features', 'Pricing Calculator', 'Pricing', and 'Documentation'.

Features



One tool, multiple clients

Build client libraries for Android, iOS and web-based clients from one source. Cloud Endpoints wraps your code to build an [API server](#) in just a few steps. Cloud Endpoint API libraries are available in Java, Python, Go and PHP. [Learn more](#)



Extending App Engine infrastructure

All of the tools and libraries made available in App Engine are now available to your mobile devices. Access Datastore, Cloud Storage and Task Queues using your App Engine backend with no extra



Low maintenance client-server

Because Cloud Endpoints is backed by App Engine, you have no servers to maintain, no load balancing to worry about and the same quick and painless scaling. Like App Engine and our other Cloud services, you only pay for what you use.

Source: <https://cloud.google.com/endpoints/>



Google Cloud Endpoints

Features



One tool, multiple clients

Build client libraries for Android, iOS and web-based clients from one source. Cloud Endpoints wraps your code to build an [API server](#) in just a few steps. Cloud Endpoint API libraries are available in Java, Python, Go and PHP. Learn more about Cloud Endpoints in our [documentation](#) or just dive in and try our sample [Tic Tac Toe](#) web application.



Extending App Engine infrastructure

All of the tools and libraries made available in App Engine are now available to your mobile devices. Access Datastore, Cloud Storage and Task Queues using your App Engine backend with no extra work. Integrate OAuth 2.0 authentication quickly by following our [examples](#).



Low maintenance client-server

Because Cloud Endpoints is backed by App Engine, you have no servers to maintain, no load balancing to worry about and the same quick and painless scaling. Like App Engine and our other Cloud services, you only pay for what you use.

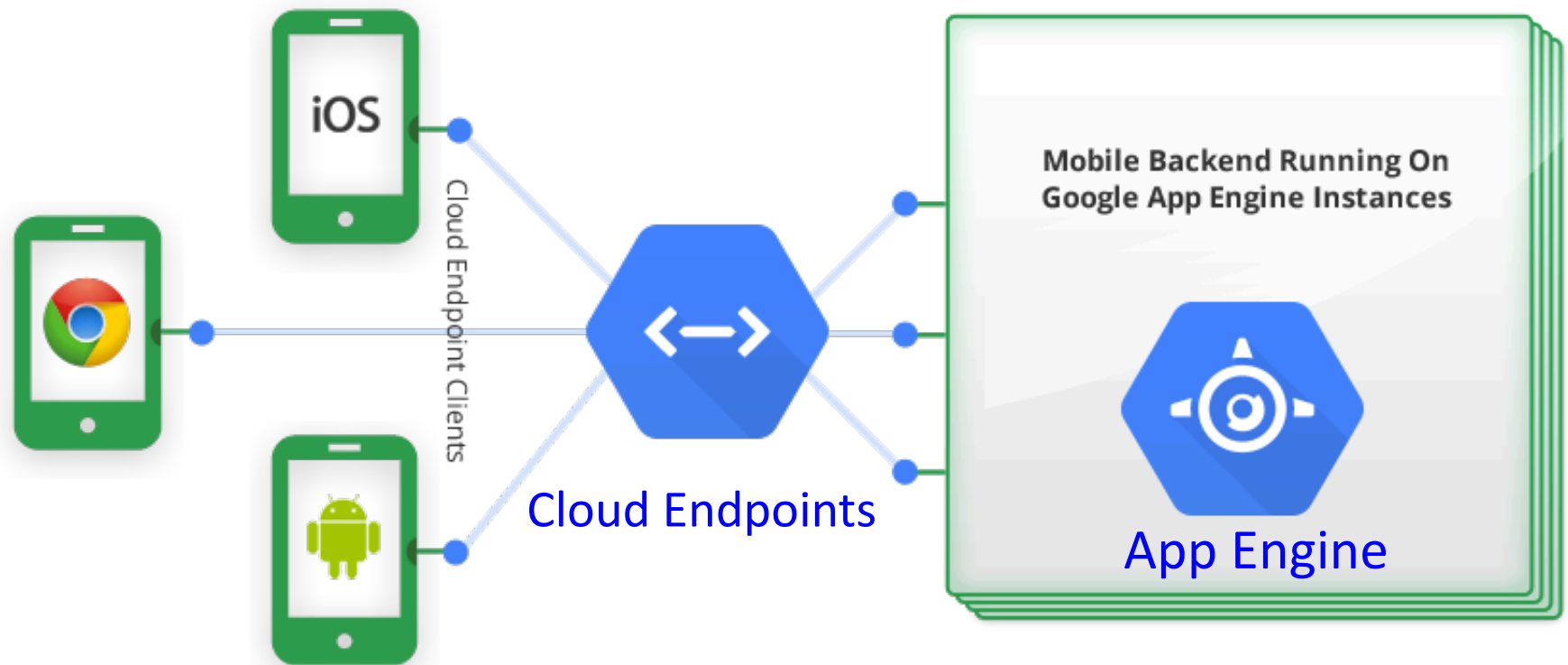


Flexible client-side integration

[Annotate](#) your server-side API and then build your client libraries automatically. Client libraries are built for [Android](#) and [iOS](#). Get standard web clients up and running with a minimal [JavaScript client](#) library. All of your clients use similar APIs and the same backend, which keeps development time down.



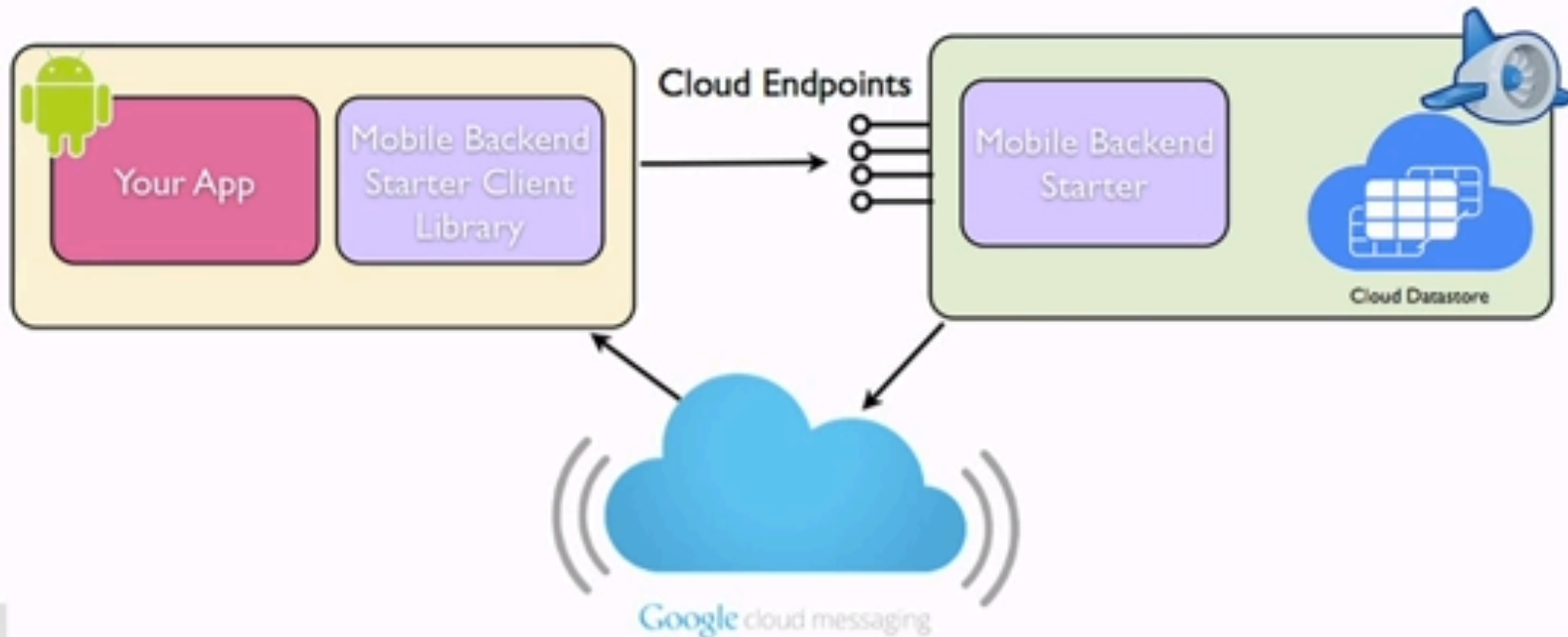
Mobile Apps Backend on Google App Engine



Google Cloud Endpoints Architecture

Mobile App, Google App Engine, Cloud Datastore

Mobile Backend Starter



Mobile, Web and Cloud

Google Cloud by Ivan Hawkes



Mobile, Web, and Cloud

The Triple Crown of Modern Applications

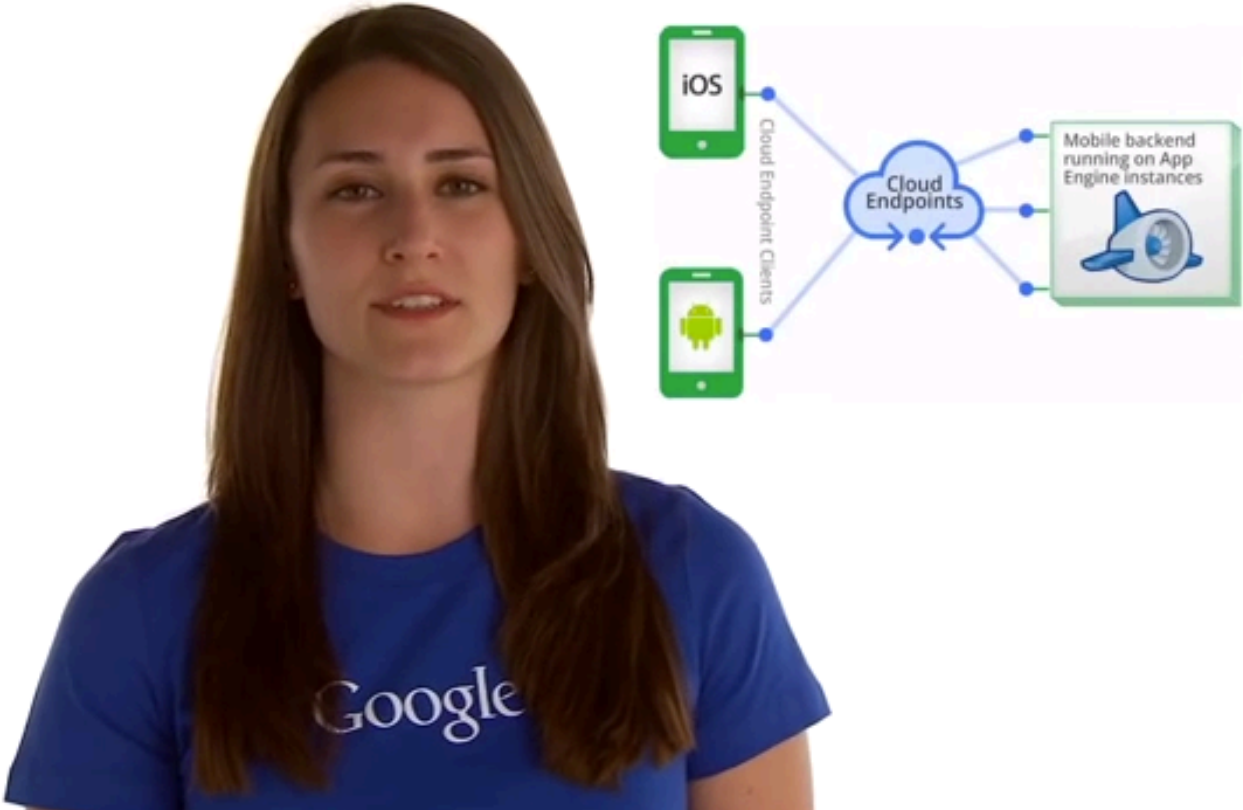
Ido Green - Developer Advocate, Google

Danny Hermes - Developer Programs Engineer, Google

Google I/O 2013 - Mobile, Web and Cloud - The Triple Crown of M...


http://www.youtube.com/watch?v=6_oO9Gwf_do


Build your mobile app with Google Cloud Platform



0:22 / 5:11

Build your mobile app with Google Cloud Platform

 GoogleDevelopers · 2,566 videos

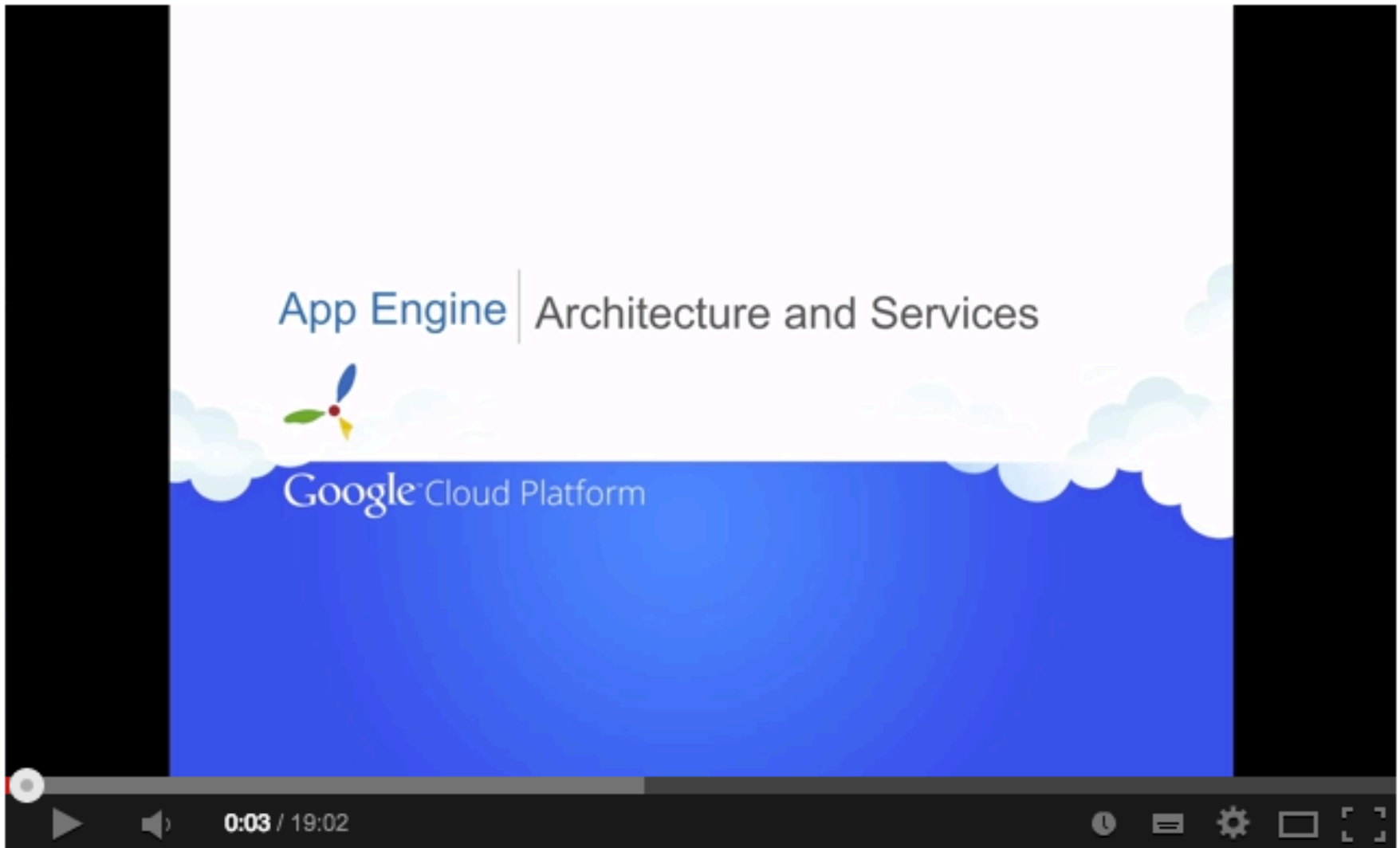
 432,318

10,744

👍 163 👎 15

<http://www.youtube.com/watch?v=ZZNb1NOPTp8>

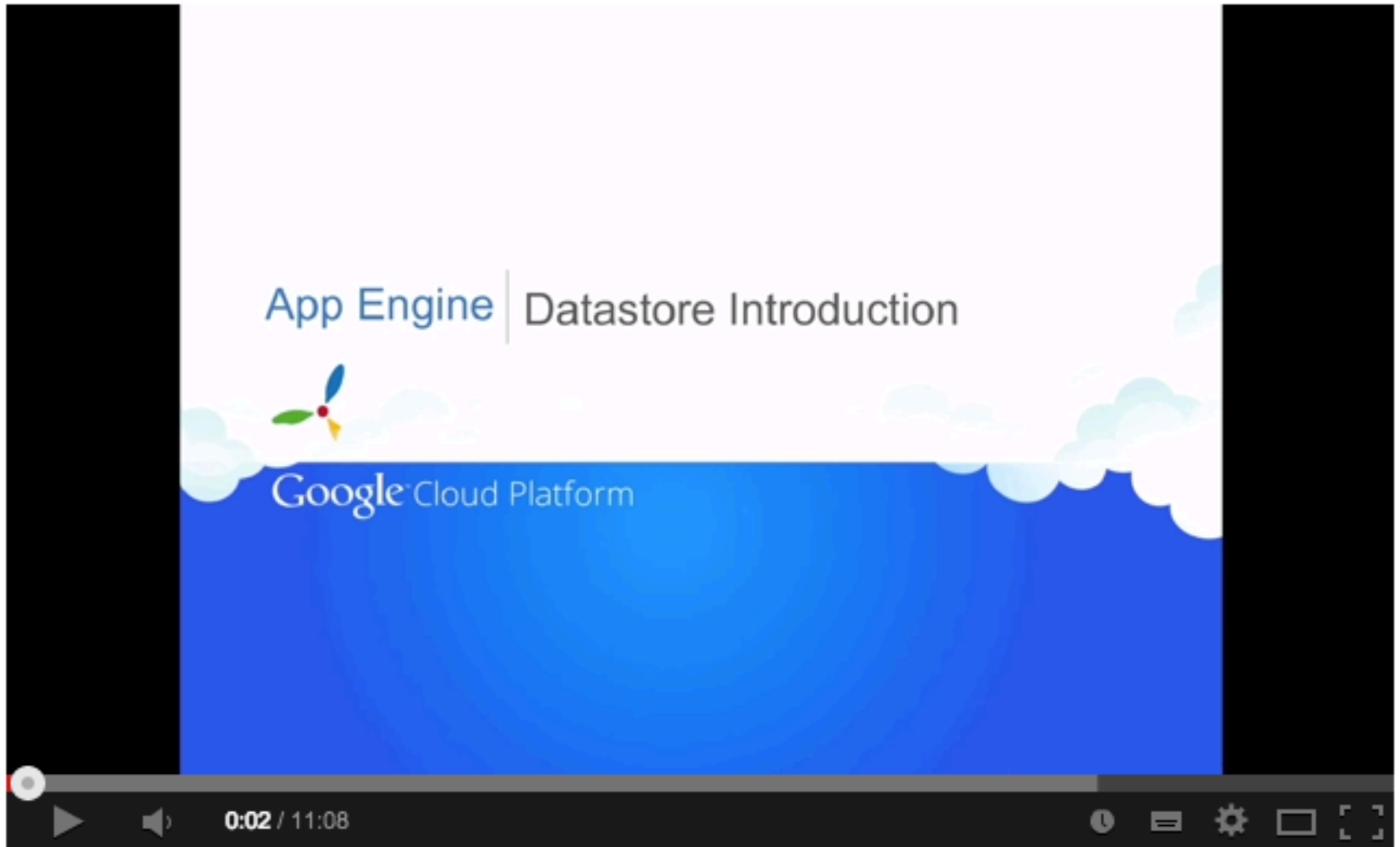
App Engine Architecture and Services



App Engine Architecture and Services

<http://www.youtube.com/watch?v=QJp6hmASstQ>

Datastore Introduction

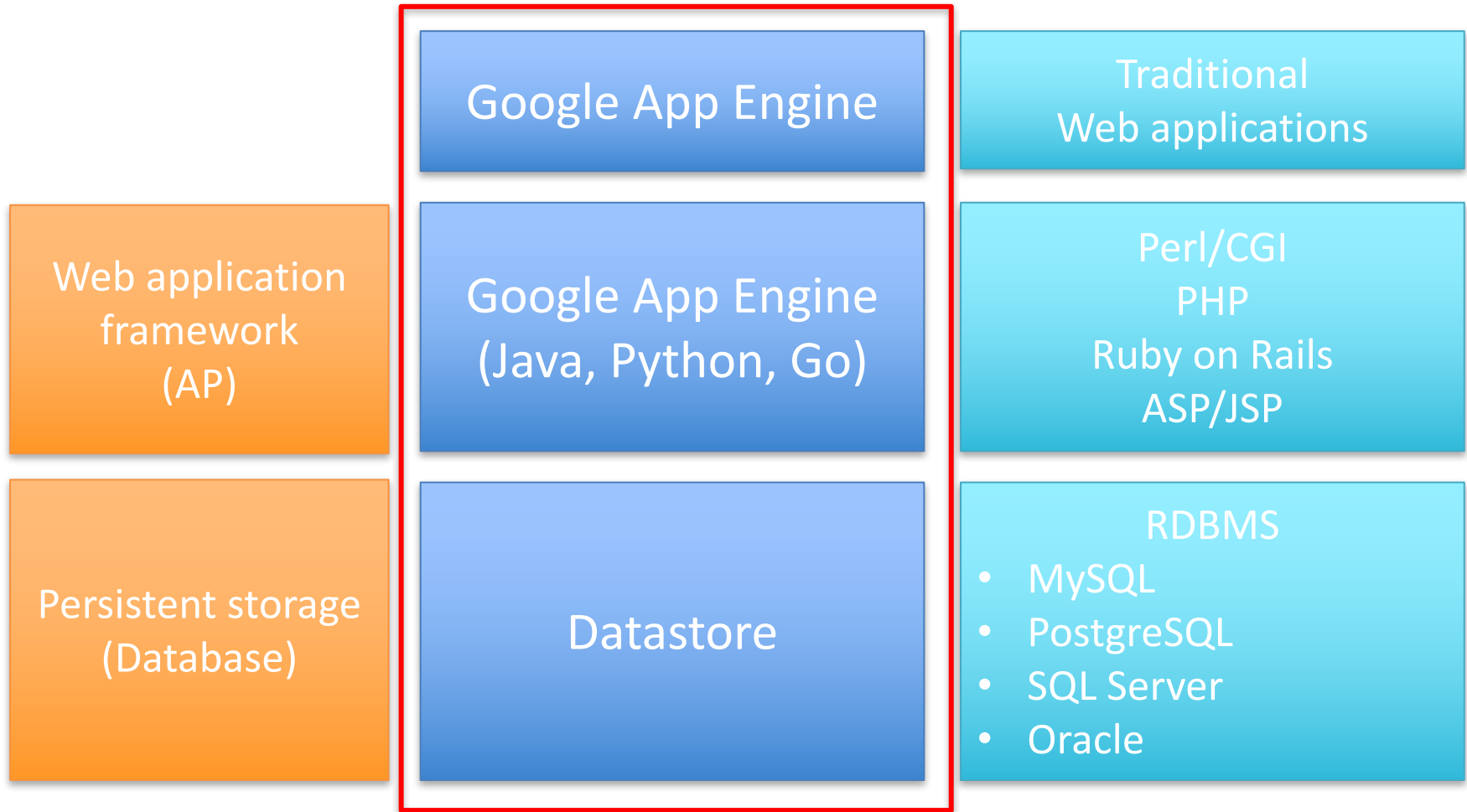


Datastore Introduction

Source: Datastore Introduction,
<http://www.youtube.com/watch?v=fQazhzcC-rg>

Google Cloud Datastore

Datastore is a database (persistent storage) for **App Engine**





Google Cloud Platform


<https://cloud.google.com/docs/>

Google Cloud Platform - Docs


← → ↺

https://cloud.google.com/docs/

☆ ☰

Google Cloud Platform

[Go to my console](#) | [Sign out](#)



[Why Google](#) [Products](#) [Solutions](#) [Pricing](#) [Customers](#) [Documentation](#) [Support](#) [Partners](#)

Free Trial

Contact Sales

Products > Documentation

Documentation  878

[Report documentation issue](#)

▼ Compute

App Engine

Compute Engine

Container Engine

▼ Storage

Cloud SQL

Cloud Storage

Cloud Datastore

▼ Networking

Cloud DNS

Carrier Interconnect

Direct Peering

HTTP Load Balancing Beta

Network Load Balancing

▼ Big Data

BigQuery

Hadoop on Google

Google Cloud Platform

With Google Cloud Platform, developers can build, test and deploy applications on Google's highly-scalable and reliable infrastructure for your web, mobile and backend solutions.

Focus on writing code, not on infrastructure, and use the same infrastructure that Google uses for your application, computing and big data needs.



Get started quickly

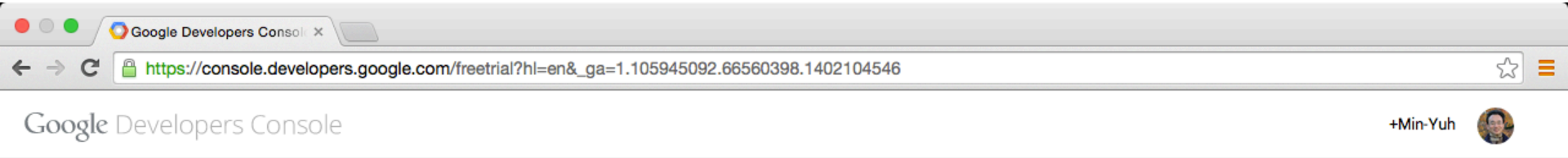
If you're ready to get started, try these solutions for getting software up and running on our platform!

 LAMP development stack

 Ruby development stack



Try Google Cloud Platform for free



Try Google Cloud Platform for free

Build on top of the infrastructure that powers Google.

Sign up for free and get \$300 to spend over 60 days on all Google Cloud Platform services.

Certain terms and conditions apply. [Learn more](#)

Country

Taiwan

Account type

☒ Business

☐ Individual

Name and address ?

Business name

Name

Street address

士林區



Get \$300 to kick start your app.

Sign up for free and get \$300 to spend on Google Cloud Platform over the next 60 days.



Why do you need my billing information?

We use your billing information to verify that you're a real person. Don't worry, you will not be billed for the free trial.



Do I have to pay when my free trial ends?

No. You're under no obligation to buy anything when the free trial ends. If you want to continue to use Google Cloud Platform, just upgrade before your trial runs out.



Questions?

Check out the [FAQ](#) or [leave us a message](#).



Try Google Cloud Platform for free

Google Developers Console

士林區

台北市

Postal code

Primary contact

Name

Phone number

imyday@gmail.com

What you pay with

Credit or debit card

Card number

VISA

AMEX

JCB

MM / YY

CVC

?

Cardholder name

☒ Credit or debit card address is same as above

Billing communication language

English (United States)

☐ I have read and agree to the Google Cloud Platform Free Trial [Terms of Service](#).

Accept and start free trial

Clear

Check out the [FAQ](#) or [leave us a message](#).

+Min-Yuh



Google Cloud Platform

<https://cloud.google.com/docs/>

Balancing **Beta**

Network Load

Balancing

▼ **Big Data**

BigQuery

Hadoop on Google Cloud Platform

▼ **Services**

Cloud Endpoints

Cloud Pub/Sub

Cloud Monitoring

Prediction API

Translate API

▼ **Management**

Deployment Manager

▼ **Tools**

Overview

Cloud SDK

▶ Android Studio

Eclipse Plugin

Cloud Repositories

Source Code Tools

Release Pipelines

Cloud Debugger

Cloud Trace


Cloud Playground

▶ Cloud Logging


▼ **Click-to-Deploy**

Get started quickly


If you're ready to get started, try these solutions for getting software up and running on our platform!

 **LAMP development stack**


LAMP (an acronym for Linux, Apache, MySQL, and PHP) is the archetypal open-source web development stack for many developers, and it runs great on Compute Engine!

 **Ruby development stack**

Ruby on Rails is one of the most popular frameworks for developing web applications, powering sites like Github, Basecamp and Shopify. Rapidly create new features, easily maintain code, and take advantage of the many open source contributions to Ruby on Rails, running on Compute Engine.

 **Quickstart for WordPress**

Set up a project, download a zip, change your config file, and deploy -and you'll have a working WordPress project running on Google Cloud Platform, with App Engine as your hosting environment.


 **App Engine "Hello World" starter**


Start editing a working "Hello World" app right now, in the browser. This gives you a good starting point and a feel for what it's like editing a working App Engine application.

Documentation

Use the following section or the left-hand navigation to access the various sets of documentation that cover Google Cloud Platform. Choose from [computing and hosting](#), [storage](#), [big data](#), [management](#), [services](#), and [developer tools](#).

Computing and hosting

 **App Engine**

 **Compute Engine**



Google Cloud Platform

<https://cloud.google.com/docs/>

Google Cloud Platform - Docs

← → ↻ <https://cloud.google.com/docs/> ☆ ☰

Cloud Trace

Cloud Playground

▶ Cloud Logging

▼ Click-to-Deploy

Cassandra

GitLab

LAMP Stack

MEAN Stack

MongoDB

RabbitMQ

Redis

Ruby on Rails

▶ Architecture Diagrams

▼ Solutions

Mobile

Hadoop

Gaming

MongoDB

RabbitMQ

Redis

Cassandra

▼ Sample Code and Videos


Videos

Sample Applications

Support Center

Terms of Service


Computing and hosting



App Engine

App Engine is Google's Platform-as-a-Service (PaaS). Develop your application easily using built-in services that make you more productive. Deploy to a fully-managed platform and let Google carry the pager. Just download the SDK and start building immediately for free with no credit card required.

[Overview](#)
[Get Started](#)
Tutorials: [Java](#), [PHP](#), [Python](#), [Go](#)
[Documentation](#)




Compute Engine

Compute Engine is Google's Infrastructure-as-a-Service (IaaS). Run large-scale workloads on virtual machines hosted on Google's infrastructure. Choose a VM that fits your needs and gain the performance and consistency of Google's worldwide fiber network. With per-minute billing, you pay only for what you use.

[Overview](#)
[Get Started](#)
[Tutorial](#)
[Documentation](#)


Storage



Cloud SQL

Store and manage data using a relational MySQL database. Google handles replication, patch management and database management to ensure availability and performance, and you can even have your instance automatically co-locate with your deployed applications.


[Overview](#)
[Tutorial](#)



Cloud Storage

Use a durable and highly available object storage service. With global edge-caching, your users have fast access to your app's data from any location. Google manages versioning, guarantees a strong SLA and provides a simple API that allows you to manage your data programmatically.

[Overview](#)
[Tutorial](#)



Cloud Datastore

Cloud Datastore provides a managed, NoSQL, schemaless database for storing non-relational data. Cloud Datastore automatically scales as you need it and supports transactions as well as robust, SQL-like queries.

[Overview](#)
[Tutorial](#)
[Documentation](#)

65



Google App Engine

App Engine "Hello World" starter

<https://console.developers.google.com/start/appengine>

Deploy your first app in five minutes

- Start editing a working "Hello World" app right now, in the browser.
- This gives you a good starting point and a feel for what it's like editing a working App Engine application.



Try Google App Engine Now

1. NAME YOUR PROJECT
2. SELECT YOUR LANGUAGE
3. EXPLORE THE STARTER CODE
4. INSTALL GOOGLE CLOUD SDK
5. RUN YOUR APP LOCALLY
6. CREATE YOUR PROJECT AND DEPLOY



Google App Engine

Google Developers Console

← → ↺ https://console.developers.google.com/start/appengine?_ga=1.26824890.66560398.1402104546 ☆ ☰

Google Developers Console

[Sign up for a free trial.](#)
[Go to my console](#)

+Min-Yuh

1. NAME YOUR PROJECT

2. SELECT YOUR LANGUAGE

3. EXPLORE THE STARTER CODE

4. INSTALL GOOGLE CLOUD SDK

5. RUN YOUR APP LOCALLY

6. CREATE YOUR PROJECT AND DEPLOY

Try Google App Engine Now

Creating an App Engine app is easy, and it's free to start. Upload your app and share it with users right away, at no charge and with no commitment required.

1 NAME YOUR PROJECT

You use your project to manage all of the Google Cloud Platform resources for your app, including deployment, access control, billing, and services. You can change your project name later.

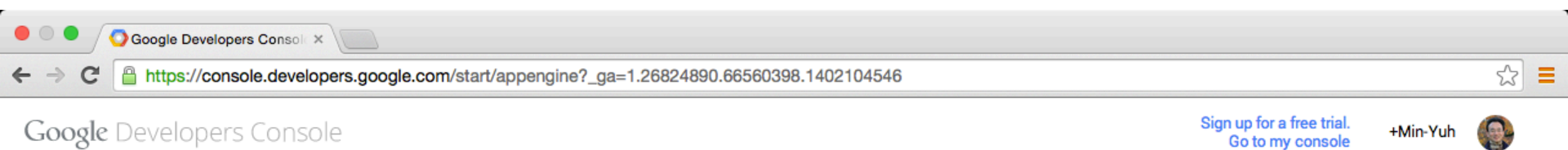
2 SELECT YOUR LANGUAGE

Python

Java



Google App Engine



2 SELECT YOUR LANGUAGE



3 EXPLORE THE STARTER CODE

Browse the starter code and see the app running below.

HELLO WORLD - JAVA

```
build.xml      1 package myapp;
pom.xml        2
src/main/java/myapp/D 3 import java.io.IOException;
src/main/webapp/WEB-INF 4 import javax.servlet.http.*;
src/main/webapp/WEB-INF 5
src/main/webapp/WEB-INF 6 public class DemoServlet extends HttpServlet {
src/main/webapp/WEB-INF 7     @Override
src/main/webapp/WEB-INF 8     public void doGet(HttpServletRequest req, HttpServletResponse resp)
src/main/webapp/WEB-INF 9         throws IOException {
src/main/webapp/WEB-INF 10        resp.setContentType("text/plain");
```



Google App Engine

Google Developers Console

Sign up for a free trial.
Go to my console

+Min-Yuh

3 EXPLORE THE STARTER CODE

Browse the starter code and see the app running below.

1. NAME YOUR PROJECT

2. SELECT YOUR LANGUAGE

3. EXPLORE THE STARTER CODE

4. INSTALL GOOGLE CLOUD SDK

5. RUN YOUR APP LOCALLY

6. CREATE YOUR PROJECT AND DEPLOY

HELLO WORLD - JAVA

build.xml

pom.xml

src/main/java/myapp/DemoServlet.java

src/main/webapp/WEB-INF/web.xml

src/main/webapp/WEB-INF/classes

src/main/webapp/index.html

```
1 package myapp;
2
3 import java.io.IOException;
4 import javax.servlet.http.*;
5
6 public class DemoServlet extends HttpServlet {
7     @Override
8     public void doGet(HttpServletRequest req, HttpServletResponse resp)
9         throws IOException {
10         resp.setContentType("text/plain");
11         resp.getWriter().println("{ \"name\": \"World\" }");
12     }
13 }
14
```

▶

Hello, World

70



Google App Engine

HELLO WORLD - JAVA

build.xml
pom.xml
[src/main/java/myapp/D](#)
src/main/webapp/WEB-INF
src/main/webapp/WEB-INF
src/main/webapp/WEB-INF
src/main/webapp/index.t

```
1 package myapp;  
2  
3 import java.io.IOException;  
4 import javax.servlet.http.*;  
5  
6 public class DemoServlet extends HttpServlet {  
7     @Override  
8     public void doGet(HttpServletRequest req, HttpServletResponse resp)  
9         throws IOException {  
10         resp.setContentType("text/plain");  
11         resp.getWriter().println("{ \"name\": \"World\" }");  
12     }  
13 }  
14
```

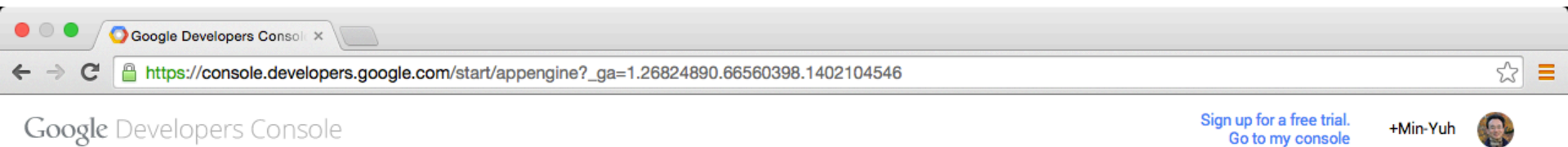


Hello, World

[Download this code](#)



Google App Engine



4 INSTALL GOOGLE CLOUD SDK

1. NAME YOUR PROJECT
2. SELECT YOUR LANGUAGE
3. EXPLORE THE STARTER CODE
4. INSTALL GOOGLE CLOUD SDK
5. RUN YOUR APP LOCALLY
6. CREATE YOUR PROJECT AND DEPLOY

Mac OS X/Linux

1. Download and install Google Cloud SDK by running the following command in your shell or Terminal:

```
curl https://sdk.cloud.google.com/ | bash
```

Follow the prompts to install the **Java** App Engine package.

2. Restart your shell or Terminal to pick up environment changes.
3. Sign in to Google Cloud Platform using this command:

```
gcloud auth login
```

4. Install the App Engine package for Java using this command:

```
gcloud components update gae-java
```

5 RUN YOUR APP LOCALLY




Google App Engine

Google Developers Console

← → ↺ https://console.developers.google.com/start/appengine?_ga=1.26824890.66560398.1402104546 ☆ ☰

Google Developers Console

Sign up for a free trial.
Go to my console

+Min-Yuh 

1. NAME YOUR PROJECT

2. SELECT YOUR LANGUAGE

3. EXPLORE THE STARTER CODE

4. INSTALL GOOGLE CLOUD SDK

5. RUN YOUR APP LOCALLY

6. CREATE YOUR PROJECT AND DEPLOY

5 RUN YOUR APP LOCALLY

Note: App Engine supports Java 7. Make sure you have the [Java 7 JDK](#) installed.

1. Download [appengine-try-java.zip](#) and unpack it. This creates your project directory, including `src/` and `war/` subdirectories.
2. Download and install [Apache Maven](#) version 3.0 or later. The starter app includes an Apache Maven build file.
3. Build and run the sample locally using the following commands:

```
cd appengine-try-java
mvn package
dev_appserver.sh target/appengine-try-java-1.0
```

You can stop the server by pressing Ctrl-C in the command window.

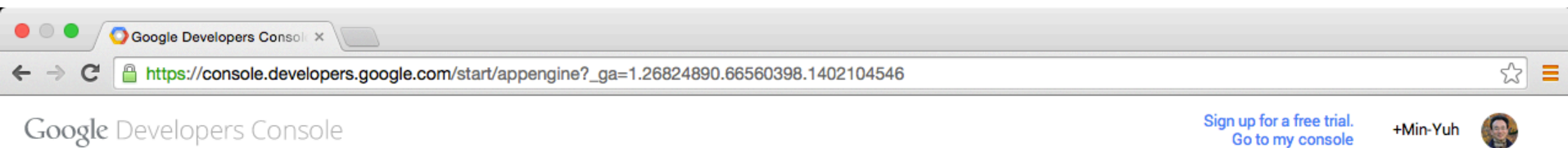
4. Visit the locally running app in your browser: <http://localhost:8080>

6 CREATE YOUR PROJECT AND DEPLOY

Now that you've seen your app running on your local machine, you're ready to create and deploy your project, HelloWorldGoogleAppEngine.



Google App Engine



5 RUN YOUR APP LOCALLY

Note: App Engine supports Java 7. Make sure you have the [Java 7 JDK](#) installed.

1. Download [appengine-try-java.zip](#) and unpack it. This creates your project directory, including **src/** and **war/** subdirectories.
2. Download and install [Apache Maven](#) version 3.0 or later. The starter app includes an Apache Maven build file.
3. Build and run the sample locally using the following commands:

```
cd appengine-try-java
mvn package
dev_appserver.sh target/appengine-try-java-1.0
```

You can stop the server by pressing Ctrl-C in the command window.

4. Visit the locally running app in your browser: <http://localhost:8080>

6 CREATE YOUR PROJECT AND DEPLOY

Now that you've seen your app running on your local machine, you're ready to create and deploy your project, HelloWorldGoogleAppEngine.



Google App Engine

Google Developers Console

https://console.developers.google.com/start/appengine?_ga=1.26824890.66560398.1402104546

Google Developers Console

Sign up for a free trial.
Go to my console

+Min-Yuh

1. NAME YOUR PROJECT

2. SELECT YOUR LANGUAGE

3. EXPLORE THE STARTER CODE

4. INSTALL GOOGLE CLOUD SDK

5. RUN YOUR APP LOCALLY

6. CREATE YOUR PROJECT AND DEPLOY

```
mvn package
dev_appserver.sh target/appengine-try-java-1.0
```

You can stop the server by pressing Ctrl-C in the command window.

4. Visit the locally running app in your browser: <http://localhost:8080>

6 CREATE YOUR PROJECT AND DEPLOY

Congratulations! Your project is ready. Your unique project ID is **psychic-outcome-783**.

Deploy your app from your local dev environment using this command:

```
appcfg.sh -A psychic-outcome-783 update target/appengine-try-java-1.0
```

After deploying your app, you can visit it with your browser at this URL:

psychic-outcome-783.appspot.com

That's it! You're running on Google App Engine. Go to your project dashboard to see how your app is performing.

View your project dashboard

Activities (Idle)

Create Project: HelloWorldGoogleAppEngine

See all activity



Google App Engine

Build an App Engine Application using Python

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
6. Using Templates
7. Using Static Files
8. Uploading Your Application



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

▼ Further Training

▼ Creating a Guestbook

1. Download the App Engine SDK

2. Explaining the webapp2 Framework

3. Using the Users Service

4. Handling Forms with webapp2

5. Using the Datastore

6. Using Templates

7. Using Static Files

8. Uploading Your Application

Sample Applications

Introduction

Welcome to Google App Engine! Creating an App Engine application is easy, only takes a few minutes, and it's free to start.

Google App Engine applications can be written in the Python 2.7, Java, Go or PHP programming languages. This tutorial covers **Python 2.7**. If you would prefer to use Java, Go or PHP to build your applications, see the [Java](#), [Go](#) or [PHP](#) guides.

In this tutorial, you will learn how to:

- build an App Engine application using Python
- use the [webapp2](#) web application framework
- use the App Engine datastore with the Python modeling API
- integrate an App Engine application with Google Accounts for user authentication
- use [Jinja2](#) templates with your app
- upload your app to App Engine

By the end of the tutorial, you will have implemented a working application, a simple guest book that lets users post messages to a public message board.

Get set up

Before we continue, you will need to download the [App Engine Python SDK](#), which includes a web server application that simulates the

Source: <https://cloud.google.com/appengine/docs/python/gettingstartedpython27/introduction>



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK

2. Explaining the webapp2 Framework

3. Using the Users Service

4. Handling Forms with webapp2

5. Using the Datastore

6. Using Templates

7. Using Static Files

8. Uploading Your Application

Sample Applications

Overviews

How App Engine Works

Developing Python Apps on App Engine

Managing and Configuring Your App

Introduction

Welcome to Google App Engine! Creating an App Engine application is easy, only takes a few minutes, and it's free to start.

Google App Engine applications can be written in the Python 2.7, Java, Go or PHP programming languages. This tutorial covers **Python 2.7**. If you would prefer to use Java, Go or PHP to build your applications, see the [Java](#), [Go](#) or [PHP](#) guides.

In this tutorial, you will learn how to:

- build an App Engine application using Python
- use the [webapp2](#) web application framework
- use the App Engine datastore with the Python modeling API
- integrate an App Engine application with Google Accounts for user authentication
- use [Jinja2](#) templates with your app
- upload your app to App Engine

By the end of the tutorial, you will have implemented a working application, a simple guest book that lets users post messages to a public message board.

Get set up

Before we continue, you will need to download the [App Engine Python SDK](#), which includes a web server application that simulates the App Engine environment, and tools to deploy your application to the App Engine production environment. Follow the directions for your operating system, then come back here so we can get going!

[Explaining the webapp2 Framework! >>](#)



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
6. Using Templates
7. Using Static Files
8. Uploading Your Application

Sample Applications

Overviews

- How App Engine Works
- Developing Python Apps on App Engine
- Managing and Configuring Your App

Explaining the webapp2 Framework

The Web Server Gateway Interface ([WSGI](#)) standard is simple, but it would be cumbersome to write all of the code that uses it by hand. Web application frameworks handle these details for you, so you can focus your development efforts on your application's features. Google App Engine supports any framework written in pure Python that speaks WSGI, including [Django](#), [CherryPy](#), [Pylons](#), [web.py](#), and [web2py](#). You can bundle a framework of your choosing with your application code by copying its code into your application directory.

App Engine includes a simple web application framework, called [webapp2](#). The [webapp2](#) framework is already installed in the App Engine environment and in the [App Engine Python SDK](#), so you do not need to bundle it with your application code to use it. We will use [webapp2](#) for the rest of this tutorial.

Hello, webapp2!

A [webapp2](#) application has two parts:

- one or more [RequestHandler](#) classes that process requests and build responses
- a [WSGIApplication](#) instance that routes incoming requests to handlers based on the URL

Let's take a look at [Hello World!](#):

helloworld.py

[View on GitHub](#)

```
import webapp2

class MainPage(webapp2.RequestHandler):
    def get(self):
        self.response.headers['Content-Type'] = 'text/plain'
        self.response.write('Hello, World!')

app = webapp2.WSGIApplication([
```



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
6. Using Templates
7. Using Static Files
8. Uploading Your Application

Sample Applications

Overviews

- How App Engine Works
- Developing Python Apps on App Engine
- Managing and Configuring Your App

Using the Users Service

Google App Engine provides several useful services based on Google infrastructure, accessible by applications using libraries included with the [App Engine Python SDK](#). One such service is the Users service, which lets your application integrate with Google user accounts. With the Users service, your users can use the Google accounts they already have to sign in to your application.

Let's use the Users service to personalize this application's greeting..

Using Users

We're going to build on the [Hello, World! application](#). You can clone the code for this step from [this branch on GitHub](#), or if you already have the original Hello, World! app, replace its contents with the following:

helloworld.py

[View on GitHub](#)

```
from google.appengine.api import users

import webapp2

class MainPage(webapp2.RequestHandler):

    def get(self):
        # Checks for active Google account session
        user = users.get_current_user()

        if user:
            self.response.headers['Content-Type'] = 'text/html; charset=utf-8'
            self.response.write('Hello, ' + user.nickname())
```

Source: <https://cloud.google.com/appengine/docs/python/gettingstartedpython27/introduction>



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

▼ Further Training

▼ Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
- 4. Handling Forms with webapp2**
5. Using the Datastore
6. Using Templates
7. Using Static Files
8. Uploading Your Application

Sample Applications

Overviews

- ▶ How App Engine Works
- ▶ Developing Python Apps on App Engine
- ▶ Managing and Configuring Your App

Handling Forms with webapp2

If we want users to be able to post their own greetings, we need a way to process information submitted by the user with a web form. The **webapp2** framework makes processing form data easy.

From Hello World to Guestbook

In order to prepare the Hello World app we've created thus far, please make the following changes:

- Rename the top level **helloworld** directory to **guestbook**
- Rename **helloworld.py** to **guestbook.py**
- Replace the **handlers** section of **app.yaml** with:

```
app.yaml
handlers:
- url: /.*
  script: guestbook.app
```

[View on GitHub](#)

Restart the development server using the new **guestbook** directory.

Handling Web Forms With webapp2

Declare that you are using **webapp2** by adding this **libraries** section to your **app.yaml**:

```
app.yaml
```

[View on GitHub](#)



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
- 5. Using the Datastore**
6. Using Templates
7. Using Static Files
8. Uploading Your Application

Sample Applications

Overviews

- ▶ How App Engine Works
- ▶ Developing Python Apps on App Engine
- ▶ Managing and Configuring Your App
- ▶ Storing Data

Using the Datastore

Storing data in a scalable web application can be tricky. A user could be interacting with any of dozens of web servers at a given time, and the user's next request could go to a different web server than the previous request. All web servers need to be interacting with data that is also spread out across dozens of machines, possibly in different locations around the world.

With Google App Engine, you don't have to worry about any of that. App Engine's infrastructure takes care of all of the distribution, replication, and load balancing of data behind a simple API—and you get a powerful query engine and transactions as well.

App Engine's data repository, the *High Replication Datastore (HRD)*, uses the [Paxos algorithm](#) to replicate data across multiple datacenters. Data is written to the Datastore in objects known as *entities*. Each entity has a *key* that uniquely identifies it. An entity can optionally designate another entity as its *parent*; the first entity is a *child* of the parent entity. The entities in the Datastore thus form a hierarchically-structured space similar to the directory structure of a file system. An entity's parent, parent's parent, and so on recursively, are its *ancestors*; its children, children's children, and so on, are its *descendants*. An entity without a parent is a *root entity*.

The Datastore is extremely resilient in the face of catastrophic failure, but its consistency guarantees may differ from what you're familiar with. Entities descended from a common ancestor are said to belong to the same *entity group*; the common ancestor's key is the group's *parent key*, which serves to identify the entire group. Queries over a single entity group, called *ancestor queries*, refer to the parent key instead of a specific entity's key. Entity groups are a unit of both consistency and transactionality: whereas queries over multiple entity groups may return stale, *eventually consistent* results, those limited to a single entity group always return up-to-date, *strongly consistent* results.

The sample application in this guide organizes related entities into entity groups, and uses ancestor queries on those entity groups to return strongly consistent results. In the example code comments, we highlight some ways this approach might affect the design of your application. For more detailed information, see [Structuring Data for Strong Consistency](#).

A Complete Example Using the Datastore

Here is a new version of [guestbook/guestbook.py](#) that creates a page footer that stores greetings in the Datastore. The rest of this page discusses excerpts from this larger example, organized under the topics of storing the greetings and retrieving them.



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
- 6. Using Templates**
7. Using Static Files
8. Uploading Your Application

Sample Applications

Overviews

- How App Engine Works
- Developing Python Apps on App Engine
- Managing and Configuring Your App

Using Templates

HTML embedded in code is messy and difficult to maintain. It's better to use a templating system, where the HTML is kept in a separate file with special syntax to indicate where the data from the application appears. There are many templating systems for Python: [EZT](#), [Cheetah](#), [ClearSilver](#), [Quixote](#), [Django](#), and [Jinja2](#) are just a few. You can use your template engine of choice by bundling it with your application code.

For your convenience, App Engine includes the Django and Jinja2 templating engines.

Using Jinja2 Templates

First modify the `libraries` section at the bottom of `guestbook/app.yaml`:

```
app.yaml
libraries:
- name: webapp2
  version: latest
- name: jinja2
  version: latest
```

[View on GitHub](#)

This configuration makes the newest supported version of Jinja2 available to your application. To avoid possible compatibility issues, serious applications should use an [actual version number](#) rather than `latest`.

Now modify the statements at the top of `guestbook/guestbook.py`:

```
guestbook.py
import os
```

[View on GitHub](#)



Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
6. Using Templates
- 7. Using Static Files**
8. Uploading Your Application

Sample Applications

Overviews

- How App Engine Works
- Developing Python Apps on App Engine
- Managing and Configuring Your App

Using Static Files

Unlike a traditional web hosting environment, Google App Engine does not serve files directly out of your application's source directory unless configured to do so. We named our template file `index.html`, but this does not automatically make the file available at the URL `/index.html`.

But there are many cases where you want to serve static files directly to the web browser. Images, CSS stylesheets, JavaScript code, movies and Flash animations are all typically stored with a web application and served directly to the browser. App Engine can serve specific files directly without you having to code your own handler.

Using Static Files

Edit `guestbook/app.yaml` and replace its contents with the following:

app.yaml

[View on GitHub](#)

```
version: 1
runtime: python27
api_version: 1
threadsafe: true

# Handlers match in order, put above the default handler.
handlers:
- url: /stylesheets
  static_dir: stylesheets

- url: /.*
  script: guestbook.app
```




Python

App Engine Home

Training and Sample Applications

Hello, World! in 5 minutes

Further Training

Creating a Guestbook

1. Download the App Engine SDK
2. Explaining the webapp2 Framework
3. Using the Users Service
4. Handling Forms with webapp2
5. Using the Datastore
6. Using Templates
7. Using Static Files

8. Uploading Your Application

Sample Applications

Overviews

- How App Engine Works
- Developing Python Apps on App Engine
- Managing and Configuring Your App

Uploading Your Application

To upload your application:

1. Sign in to App Engine using your Google account. If you do not have a Google account, you can [create a Google account](#) with an email address and password.
2. If you haven't already done so, create a project for your App Engine app as follows:
 - a. Visit the [Google Cloud Platform Console](#) and click **Create Project**.
 - b. Supply the desired project name in the New Project form. It doesn't have to match your app name, but using the same name as your app might make administration easier.
 - c. Accept the generated project ID or supply your own ID. *This project ID is used as the App Engine application ID.* Note that this ID can only be used once: if you subsequently delete your project, you won't be able to re-use the ID in a new project.

Note: You can specify that your new application should reside in the European Union, rather than the United States. Hosting applications in the European Union is especially useful if your application's users are closer to Europe than to the United States. There is less network latency and the End User Content will be stored at rest in the European Union. You must specify this location when you register the application; you cannot change it later. Click **Show Advanced Options** in the Create Project section to select a location option, either United States or European Union.

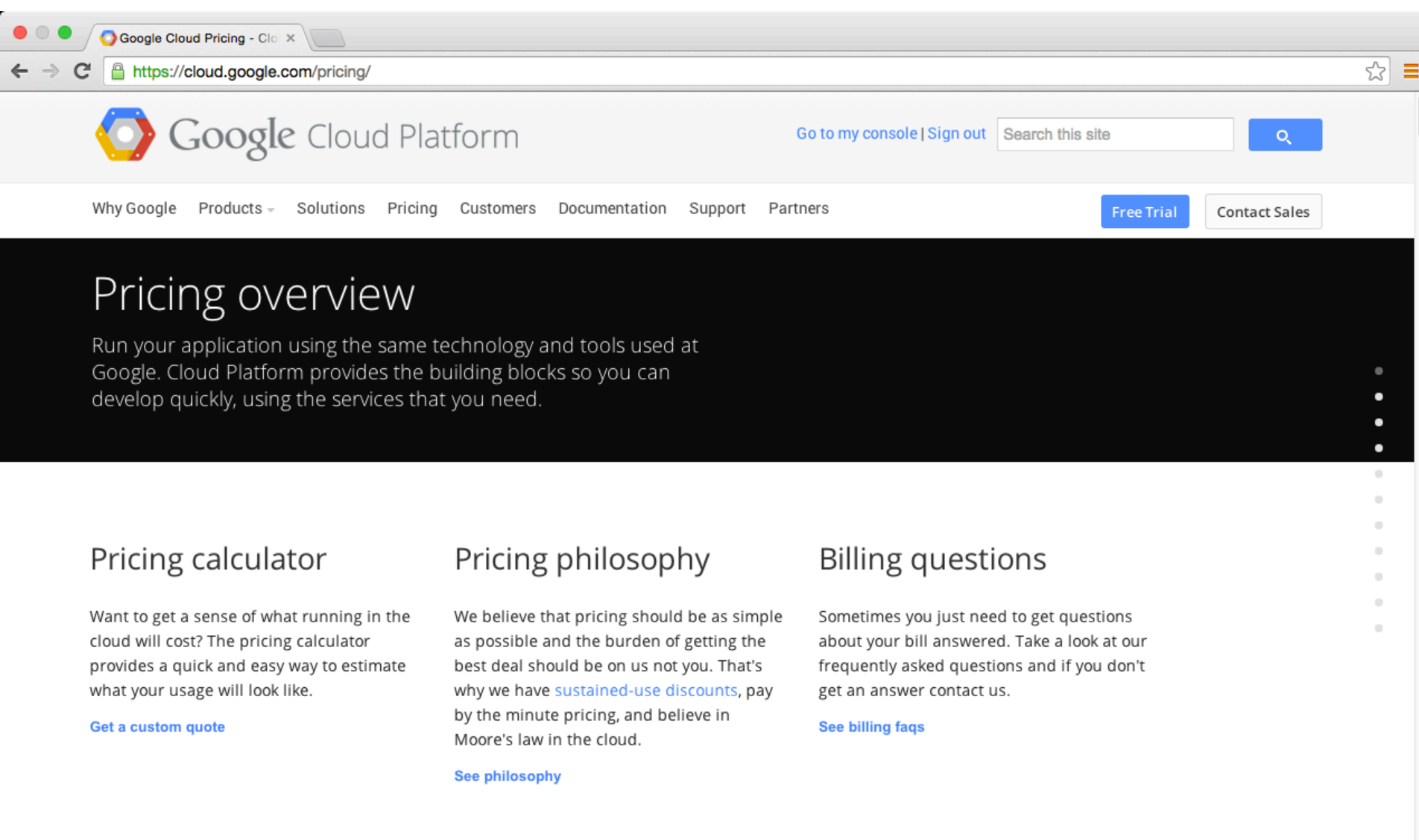
3. Note the application ID (project ID) you created above.
4. Upload your finished application to Google App Engine by invoking the following command. This opens a browser window for you to sign in using your Google account. You'll be providing the project ID as the argument for `-A`.

```
appcfg.py -A <YOUR_PROJECT_ID> update guestbook/
```

5. The [Datastore Indexes](#) may take some time to generate before your application is available. You will receive a `NeedIndexError` when accessing your app if the indexes are still in the process of being generated. This is a transient error for the example, so try a little later if at first you receive this exception.
6. Your app is now deployed and ready for users!



Google Cloud Platform



The screenshot shows a web browser window with the Google Cloud Platform Pricing page. The browser's address bar shows the URL <https://cloud.google.com/pricing/>. The page header includes the Google Cloud Platform logo, navigation links for 'Go to my console' and 'Sign out', a search bar, and a navigation menu with links for 'Why Google', 'Products', 'Solutions', 'Pricing', 'Customers', 'Documentation', 'Support', and 'Partners'. There are also buttons for 'Free Trial' and 'Contact Sales'. The main content area has a dark background with the title 'Pricing overview' and a sub-header 'Run your application using the same technology and tools used at Google. Cloud Platform provides the building blocks so you can develop quickly, using the services that you need.' Below this, there are three columns: 'Pricing calculator' with a link 'Get a custom quote', 'Pricing philosophy' with a link 'See philosophy', and 'Billing questions' with a link 'See billing faqs'.

Google Cloud Pricing - Clo x

<https://cloud.google.com/pricing/>

Google Cloud Platform

[Go to my console](#) | [Sign out](#)

[Why Google](#) [Products](#) [Solutions](#) [Pricing](#) [Customers](#) [Documentation](#) [Support](#) [Partners](#) [Free Trial](#) [Contact Sales](#)

Pricing overview

Run your application using the same technology and tools used at Google. Cloud Platform provides the building blocks so you can develop quickly, using the services that you need.

Pricing calculator

Want to get a sense of what running in the cloud will cost? The pricing calculator provides a quick and easy way to estimate what your usage will look like.

[Get a custom quote](#)

Pricing philosophy

We believe that pricing should be as simple as possible and the burden of getting the best deal should be on us not you. That's why we have [sustained-use discounts](#), pay by the minute pricing, and believe in Moore's law in the cloud.

[See philosophy](#)

Billing questions

Sometimes you just need to get questions about your bill answered. Take a look at our frequently asked questions and if you don't get an answer contact us.

[See billing faqs](#)

Google Cloud Platform

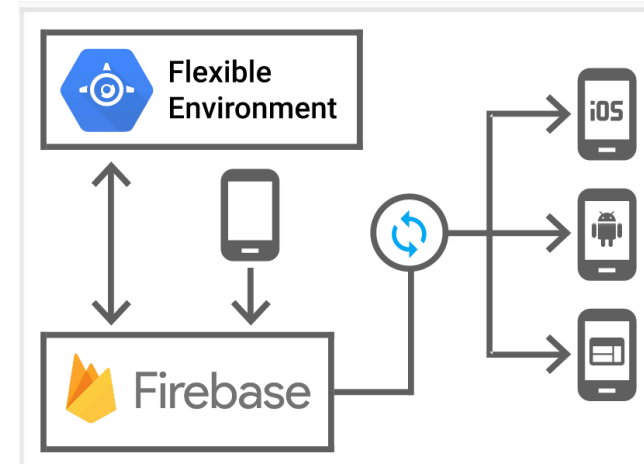
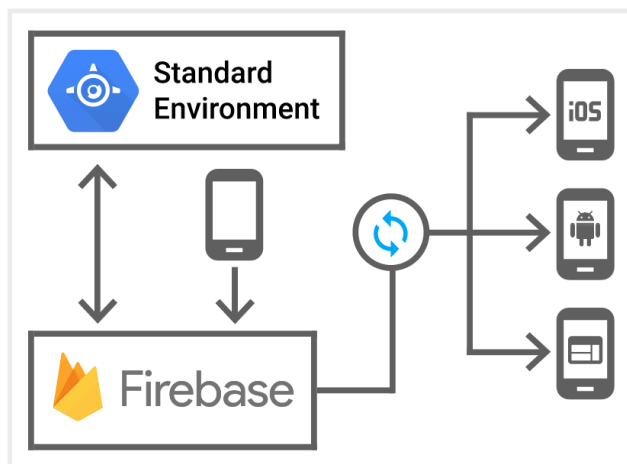
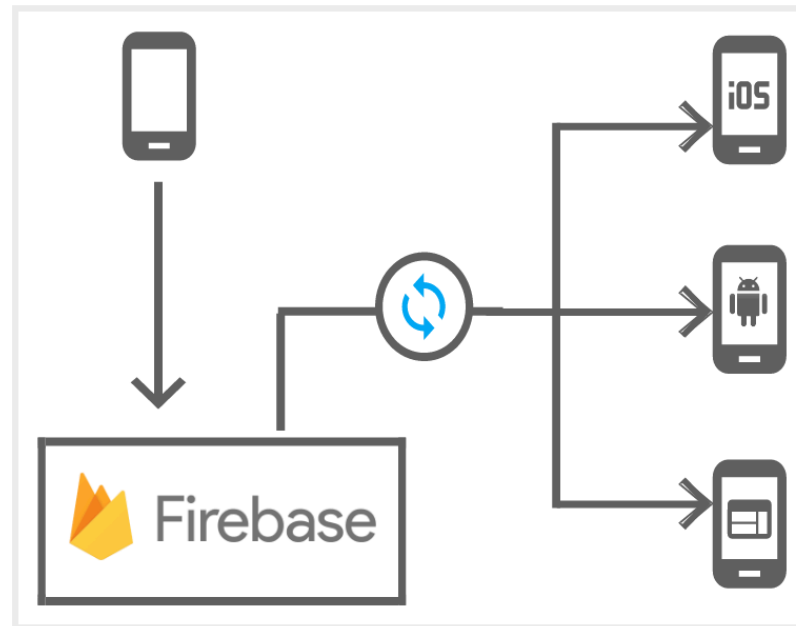
- Google Cloud Platform is a set of modular **cloud-based services** that allow you to create anything from simple websites to complex applications.



Google Cloud Platform



Mobile App Backend Services



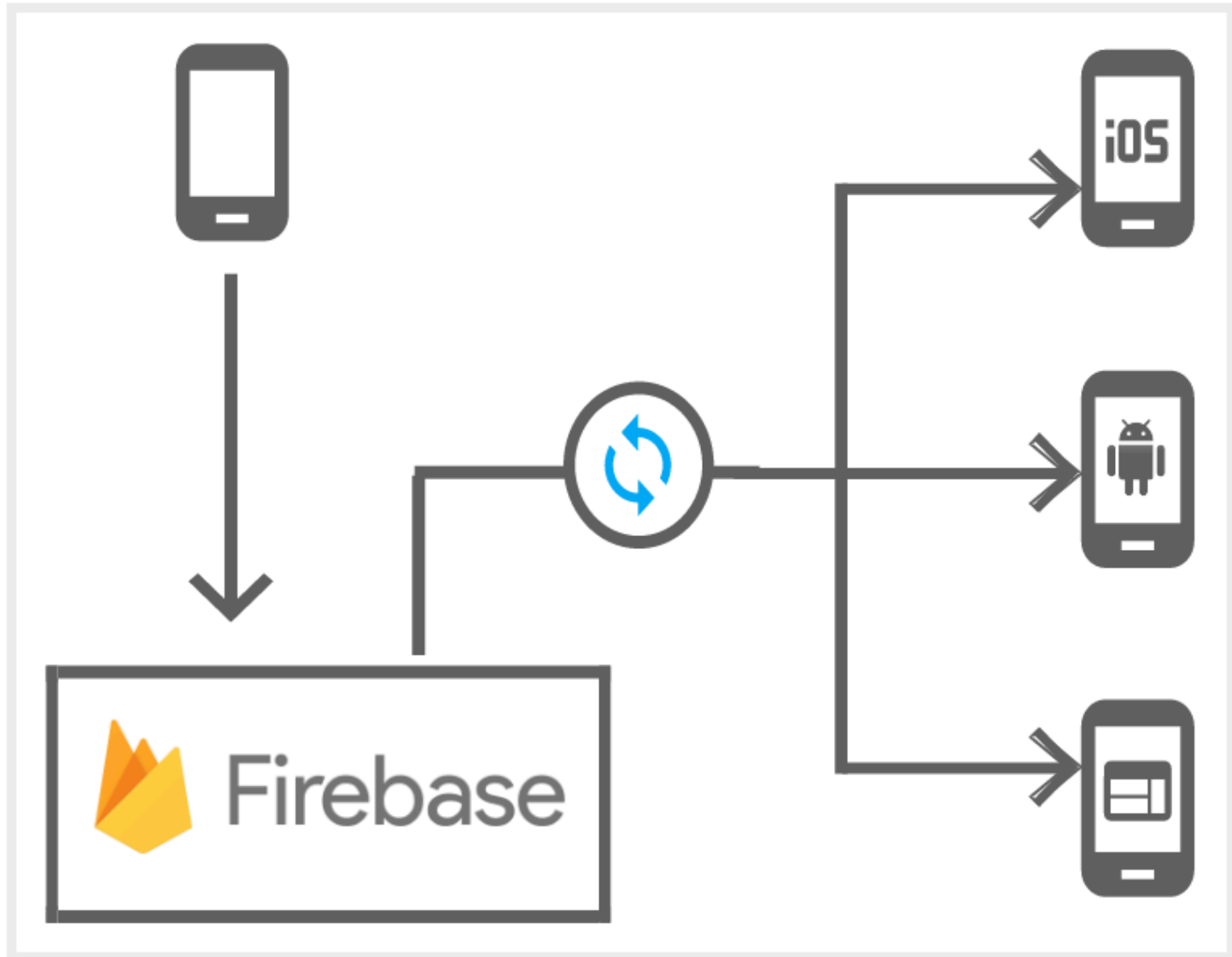


Firebase

Firebase: Backend as a Service (BaaS)

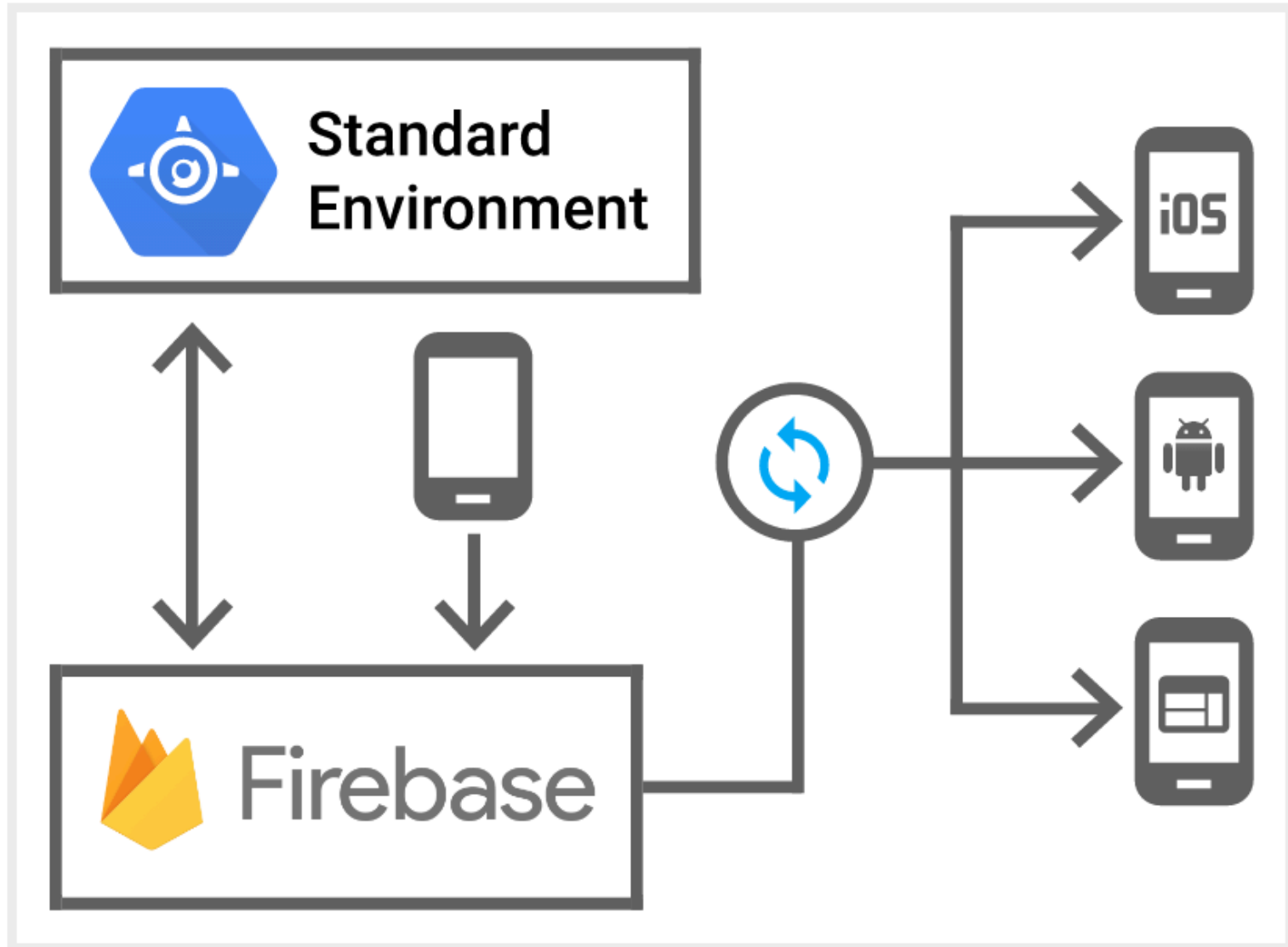


Mobile App Backend Services





Mobile App Backend Services





Firebase

Firebase

← → ↻ <https://firebase.google.com> ☆

Firebase Home Features Pricing Docs Customers Support 🔍 Search [Go to console](#)

App success made simple

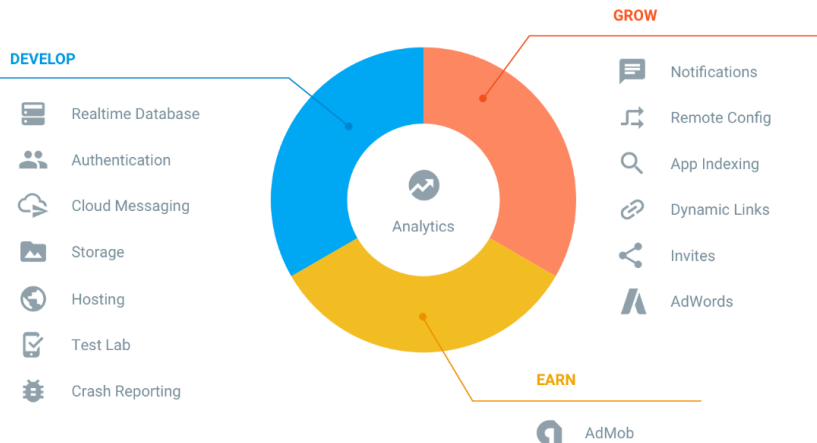
The tools and infrastructure you need to build better apps and grow successful businesses

[GET STARTED FOR FREE](#)

Move fast

Firebase is a mobile platform that helps you quickly **develop** high-quality apps, **grow** your user base, and **earn** more money. Firebase is made up of complementary features that you can mix-and-match to fit your needs.








[ALL FEATURES](#)

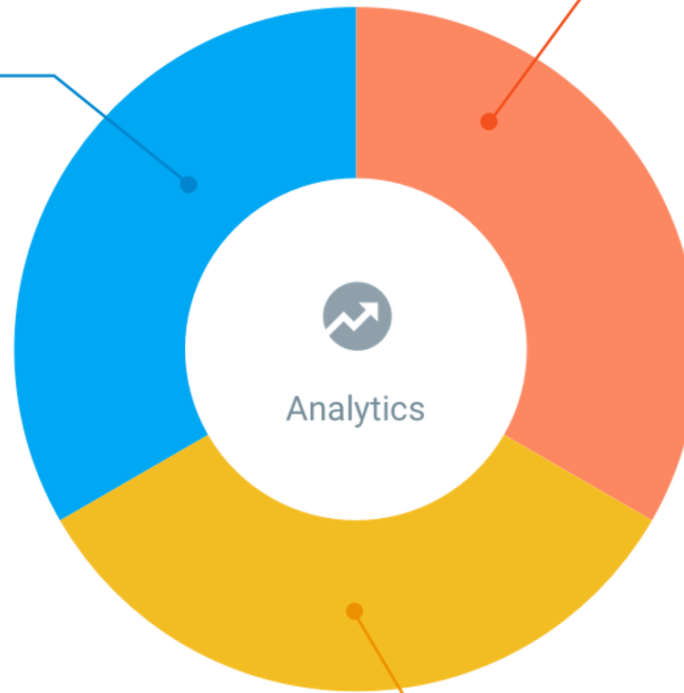


firebase is a **mobile platform**
that helps you quickly
develop high-quality apps,
grow your user base, and
earn more money.







Firebase

DEVELOP

-  Realtime Database
-  Authentication
-  Cloud Messaging
-  Storage
-  Hosting
-  Test Lab
-  Crash Reporting



GROW

-  Notifications
-  Remote Config
-  App Indexing
-  Dynamic Links
-  Invites
-  AdWords

EARN





Firebase

Firebase

← → ↻ <https://firebase.google.com> ☆



Home

Features

Pricing

Docs

Customers

Support



Search

[Go to console](#)



SWIFT

OBJECTIVE-C

JAVA

JAVASCRIPT

MORE ▾

```
FIRAuth.auth()?.addAuthStateChangeListener({ (auth, user) in
  if (user) {
    var email = user.email
    // ...
  }
})
FIRAuth.auth()?.signInWithEmail(email,
                                password: password,
                                completion: { (_, error) in

  if (error) {
    // ...
  }
})
```

Work across platforms

Deliver cross-platform apps with APIs packaged into single SDKs for iOS, Android, JavaScript and C++. Expand to a different platform without modifying your infrastructure.

 [Try iOS tutorial](#)

 [Try Android tutorial](#)

 [Try Web tutorial](#)

Free to start, scale with ease

Most Firebase features are free forever, for any scale. Our four paid features have a generous free tier and two paid plans once you begin to grow.



Summary

- **Google Cloud Platform**

- Google App Engine



- Google Cloud Datastore



- Google Cloud Endpoints



- **Firebase**



Firebase

- **Mobile App with Google Cloud Platform**



Google Cloud Platform

References

- Google Cloud Platform,
<https://cloud.google.com/>
- Google App Engine,
<https://cloud.google.com/appengine/>
- Google Cloud Datastore,
<https://cloud.google.com/datastore/>
- Google Cloud Endpoints,
<https://cloud.google.com/endpoints/>
- Google Firebase
<https://firebase.google.com/>