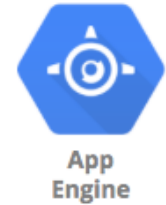


App Engine

App Engine



- **Simplest way to deploy and scale your applications in GCP**
 - Provides **end-to-end application management**
- **Supports:**
 - Go, Java, .NET, Node.js, PHP, Python, Ruby using pre-configured runtimes
 - Use custom run-time and write code in any language
 - Connect to variety of Google Cloud storage products (Cloud SQL etc)
- **No usage charges** - Pay for resources provisioned
- **Features:**
 - Automatic load balancing & Auto scaling
 - Managed platform updates & Application health monitoring
 - **Application versioning**
 - Traffic splitting

Compute Engine vs App Engine

- **Compute Engine**

- **IAAS**
- **MORE Flexibility**
- **MORE Responsibility**
 - Choosing Image
 - Installing Software
 - Choosing Hardware
 - Fine grained Access/Permissions (Certificates/Firewalls)
 - Availability etc

- **App Engine**

- **PaaS**
- **Serverless**
- **LESSER Responsibility**
- **LOWER Flexibility**

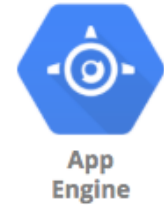


App
Engine



Compute
Engine

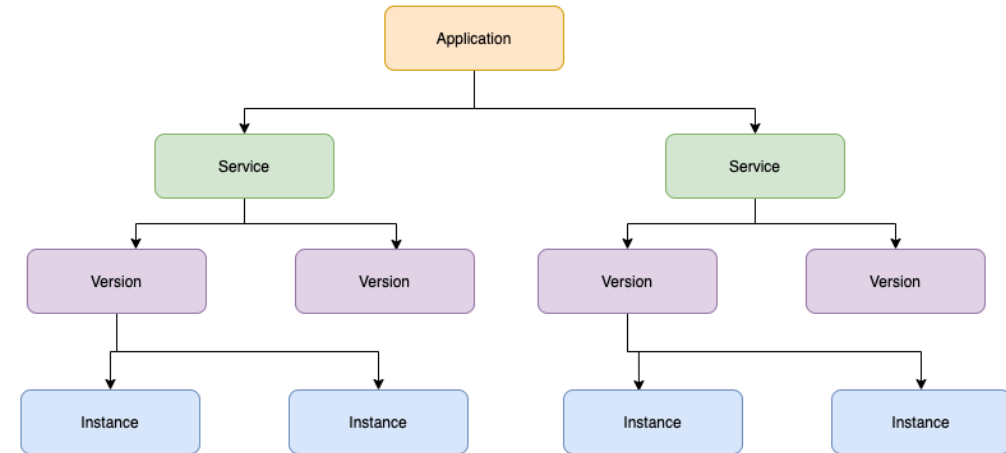
App Engine environments



- **Standard:** Applications run in language specific sandboxes
 - Complete isolation from OS/Disk/Other Apps
 - **V1:** Java, Python, PHP, Go (OLD Versions)
 - ONLY for Python and PHP runtimes:
 - Restricted network Access
 - Only white-listed extensions and libraries are allowed
 - No Restrictions for Java and Go runtimes
 - **V2:** Java, Python, PHP, Node.js, Ruby, Go (NEWER Versions)
 - Full Network Access and No restrictions on Language Extensions
- **Flexible** - Application instances run within Docker containers
 - Makes use of Compute Engine virtual machines
 - Support ANY runtime (with built-in support for Python, Java, Node.js, Go, Ruby, PHP, or .NET)
 - Provides access to background processes and local disks

App Engine - Application Component Hierarchy

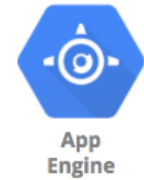
- **Application:** One App per Project
- **Service(s):** Multiple Microservices or App components
 - You can have multiple services in a single application
 - Each Service can have different settings
 - Earlier called Modules
- **Version(s):** Each version associated with code and configuration
 - Each Version can run in one or more instances
 - Multiple versions can co-exist
 - Options to rollback and split traffic



App Engine - Comparison

Feature	Standard	Flexible
Pricing Factors	Instance hours	vCPU, Memory & Persistent Disks
Scaling	Manual, Basic, Automatic	Manual, Automatic
Scaling to zero	Yes	No. Minimum one instance
Instance startup time	Seconds	Minutes
Rapid Scaling	Yes	No
Max. request timeout	1 to 10 minutes	60 minutes
Local disk	Mostly(except for Python, PHP). Can write to /tmp.	Yes. Ephemeral. New Disk on startup.
SSH for debugging	No	Yes

App Engine - Scaling Instances



- **Automatic** - Automatically scale instances based on the load:
 - Recommended for Continuously Running Workloads
 - Auto scale based on:
 - **Target CPU Utilization** - Configure a CPU usage threshold.
 - **Target Throughput Utilization** - Configure a throughput threshold
 - **Max Concurrent Requests** - Configure max concurrent requests an instance can receive
 - Configure **Max Instances** and **Min Instances**
- **Basic** - Instances are created as and when requests are received:
 - Recommended for Adhoc Workloads
 - Instances are shutdown if ZERO requests
 - Tries to keep costs low
 - High latency is possible
 - NOT supported by App Engine Flexible Environment
 - Configure **Max Instances** and **Idle Timeout**
- **Manual** - Configure specific number of instances to run:
 - Adjust number of instances manually over time

AppEngine Demo

- Deploy an application to cloud using App Engine