DEVELOPER SURVIVAL GUIDE

EXPRESS SERVERLESS PLATFORM FOR GCP vs. GOOGLE CLOUD FUNCTIONS + CLOUD ENDPOINTS

LunchBadger

Express Serverless Platform for GCP vs Google Cloud Platform (Google Cloud Functions + Endpoints)

ABSTRACT

We've created this comparison page to make it easy to understand the major differences (and similarities) between Google Cloud's platforms for the developing serverless functions as microservices and running Express Serverless Platform's inside Google Cloud for an even better experience. In this overview we'll be reviewing Express Serverless Platform for Google Cloud Platform (GCP) and Google Cloud Functions across multiple dimensions and "at-a-glance."'

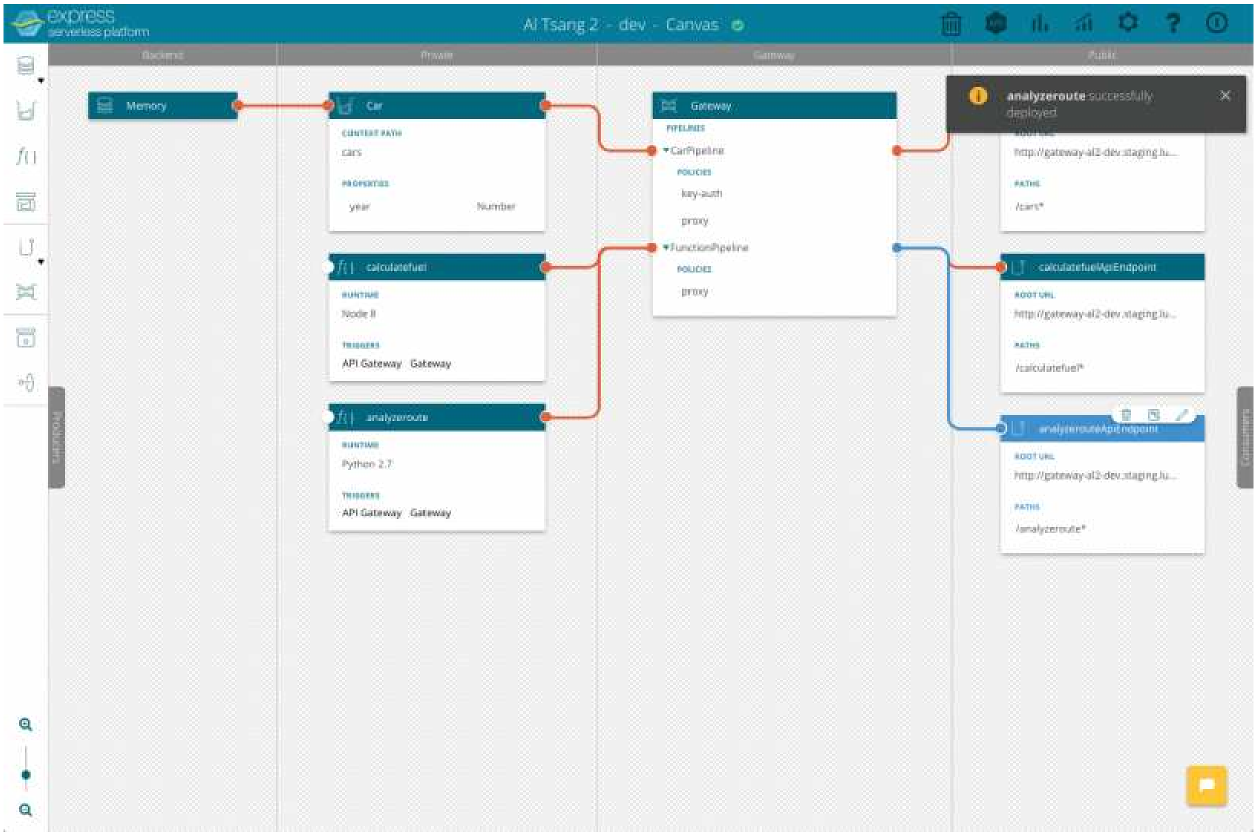
What is Express Serverless Platform?

Express Serverless Platform provides a uniform and seamless development experience for microservices and APIs as simple functions.

Express Serverless Platform can be deployed to any public or private cloud.

Enterprises can have multiple deployments of Express Serverless Platform at the same time to achieve a multi cloud and or hybrid cloud strategy.

Each deployed instance of Express Serverless Platform allows you to take advantage of a cloud's native infrastructure without coupling your code to its proprietary interfaces.



analyzeroute successfully deployed

Express Serverless Platform's visual interface is called the Canvas. The Canvas gets everyone on the same page on what microservices are there and where and how they are running.

Actions done inside the Canvas are orchestrated and automated within Kubernetes as pods running container based microservices in real time for development and modeling purposes. Other actions in the Canvas can also automate and deploy native cloud infrastructure such as serverless offerings (e.g. Google Cloud Functions) running alongside the cloud's Kubernetes offering.

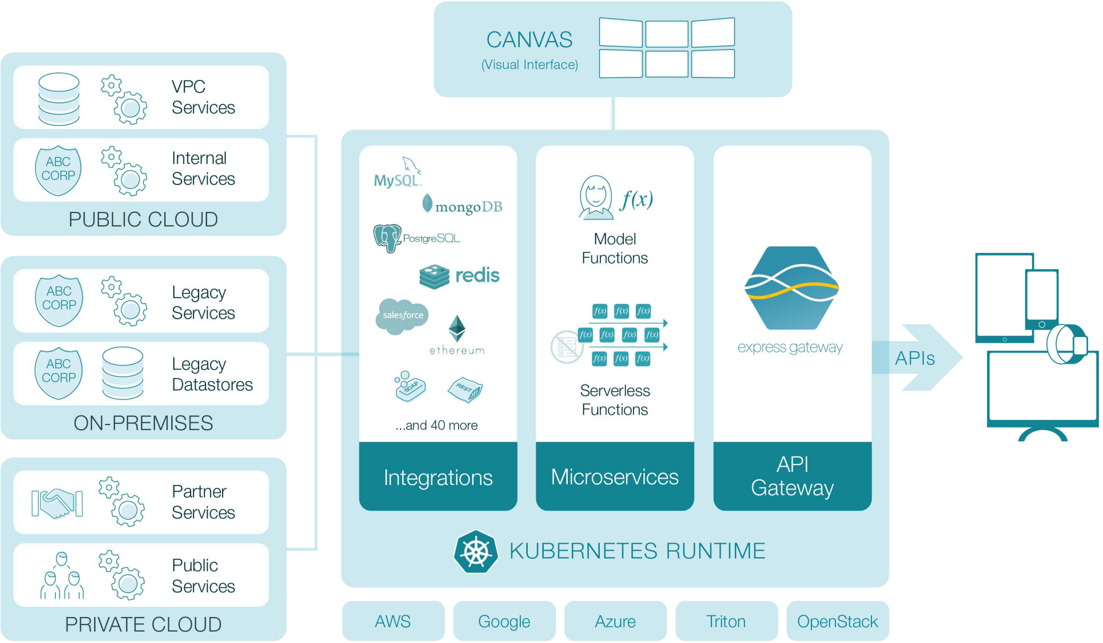
Here is an example;
Express Serverless Platform for GCP is installed in Google Kubernetes Engine as a Helm application.

Express Serverless Platform builds and automates microservices inside of GKE running as containers and also serverless functions into GCP's native serverless offering - Cloud Functions.

Express Serverless Platform automates and manages both container and serverless microservices seamlessly giving you unified view of what your microservices application looks like and how they're orchestrated across different pieces of infrastructure.

Features & Architecture

Express Serverless Platform Features & Architecture



Express Serverless Platform can be installed on bare metal, VMs, or any managed Kubernetes offering with all the necessary infrastructure components included: an API Gateway, Kubernetes Based Runtime, Serverless Engine and visual Canvas that serves as a "single pane of glass" to visualize your microservices and API development.

It is fully modularized and integration ready. Any deployment can take full advantage of a particular public cloud’s native proprietary offerings such as its own serverless infrastructure and managed Kubernetes offering.

Express Serverless Platform also utilizes the best of breed open source technologies in its reference implementation that is ready to be used out of the box as is:

- Express Gateway, an open source API Gateway written in Node.js and built on top of Express.js
- Loopback.js, and enterprise Node.js framework for building microservices as model based functions and vast microservice integration library of connectors
- Kubeless, a Kubernetes native serverless engine to run polyglot functions
- Serverless (the framework), a framework that provides abstraction of functions to multiple serverless implementations - AWS Lambda, Azure Functions, Google Cloud Functions, Openwhisk, Kubeless and others
- Kubernetes, the leading container orchestrator to run microservices running in containers, supported by all public and private clouds

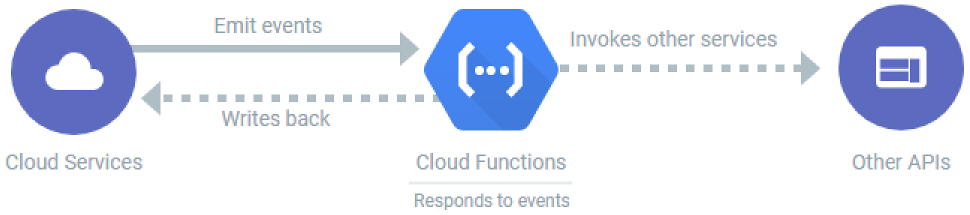
Express Serverless Platform Features & Architecture Cont.

Features Include

- Composition of Serverless Functions
- Composition of Model based Functions
- Supports Node.js, Python, Ruby, Go, PHP, .Net Core, and Java
- Completely extensible through open source modules
- Suite of Enterprise connectors to tap into legacy systems for Model based Functions
- Connects multi-cloud environments (private or public cloud)
- Supports any microservice use cases, patterns and designs
- Auto deployment to a Kubernetes Runtime
- Works with any Kubernetes cluster (EKS, GKE, etc)
- Auto exposes functions as APIs through the gateway
- Plugs directly into existing DevOps tooling and pipelines
- Can take advantage of public cloud’s proprietary infrastructure services
- Reference implementation comes complete with all infrastructure components like API gateway, serverless engine, container orchestrator, and GUI

Further reading: LunchBadger Documentation

Google Cloud Functions Features & Architecture



Google Cloud Functions are Google Cloud's equivalent to AWS Lambda. Like Lambda, Google Cloud Functions let you write custom functions and expose them via an HTTP interface through either Google Cloud Function's minimal HTTP configuration or Google Cloud Endpoints for more sophisticated use cases.

Features Include:

- Deploys, runs and scales your code with no server management
- Spin up Cloud Functions on-demand and back down in response to events in the environment
- Connects Google Cloud Platform, Firebase, Google Assistant
- Call your code directly from any web, mobile, or backend application via HTTP
- Support for Cloud Pub/Sub, Cloud storage, HTTP, Stackdriver Logging,Firebase event sources
- Log and monitor performance
- Debugging and Error Reporting
- Create, Manage and Deploy Cloud Functions with standard Cloud SDK, web interface and both REST and gRPC APIs in a standard Node.js runtime

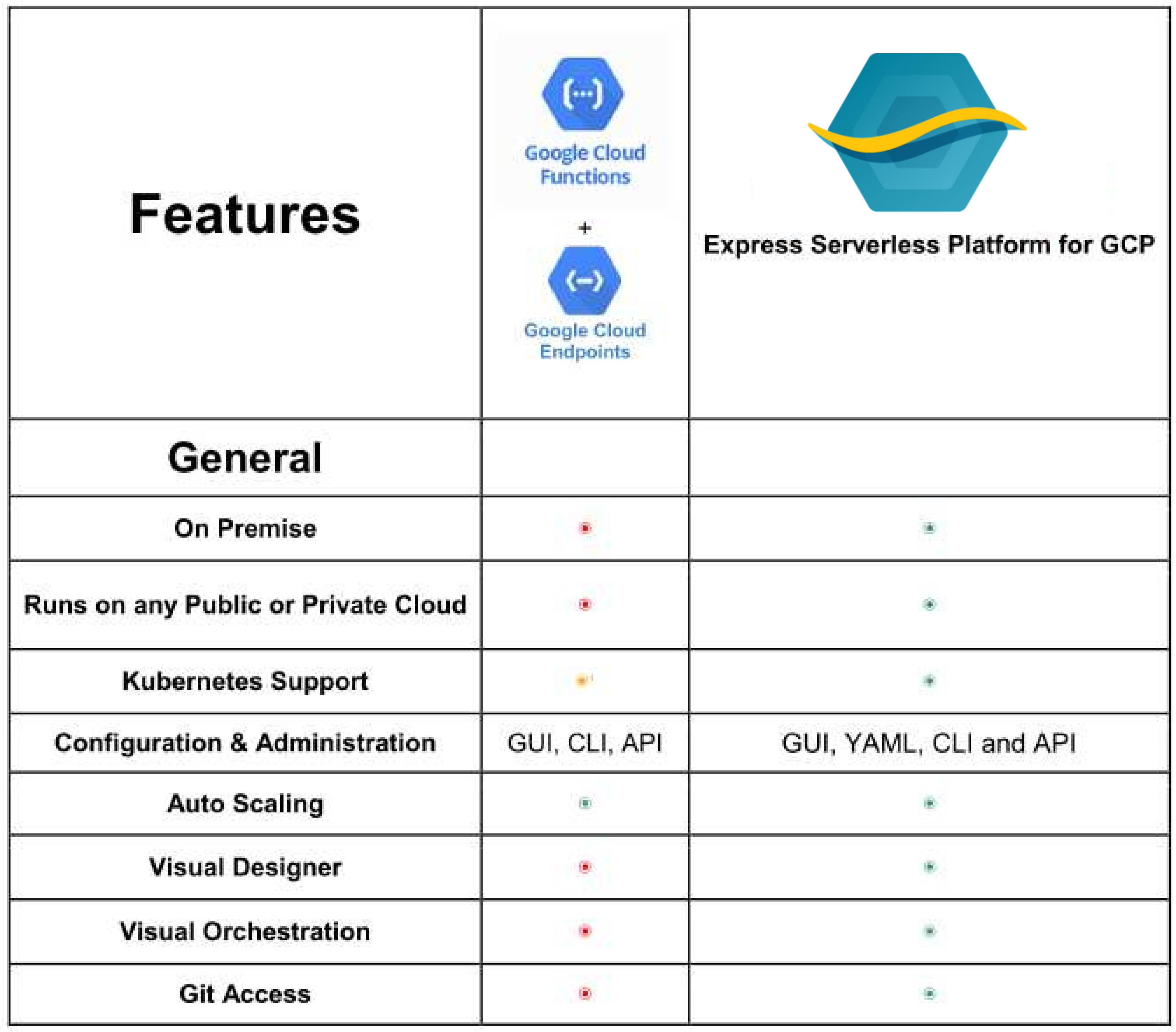
Further reading: Google Cloud Function Features

Features Comparison

The following table has Google Cloud Functions and Google Cloud Endpoints (two separate products used together) compared with the Express Serverless Platform for GCP.

Note - Google also offers Apigee as an enterprise level offering that has significantly more gateway features, but this is again another product all together.

🟢= Supported
🟠= Supported with caveats
🔴= Not supported



Features	Google Cloud Functions + Google Cloud Endpoints	Express Serverless Platform for GCP
General		
On Premise	🔴	🟢
Runs on any Public or Private Cloud	🔴	🟢
Kubernetes Support	🟠[1]	🟢
Configuration & Administration	GUI, CLI, API	GUI, YAML, CLI and API
Auto Scaling	🟢	🟢
Visual Designer	🔴	🟢
Visual Orchestration	🔴	🟢
Git Access	🔴	🟢

Serverless	Google Cloud Functions + Google Cloud Endpoints	Express Serverless Platform on GCP
HTTP Functions	🟢	🟢
Background Functions	🟢	🔴
Model Functions	🔴	🟢
Node.js Functions	🟢	🟢
Python Functions	🟢	🟢
Go Functions	🔴	🟢
.NET Core	🔴	🟢
Ruby Functions	🔴	🟢
PHP Functions	🔴	🟢
Java Functions	🔴	🟢
Auto REST Scaffolding	🔴	🟠[2]
Events and Triggers	🟢	🟠[3]
Pre-Built Connectors	GCP only	8/20+[4]

API Management	Google Cloud Functions + Google Cloud Endpoints	Express Serverless Platform on GCP
HTTPS	🟢	🟢
CORS	🟢	🟢
Basic Auth	through Auth0	🟢
OAuth2	through Auth0	🟢
Key Authentication	through GCP	🟢
JWT	🟢	🟢
Finegrain Access Control	🔴	🟢
Rate Limiting	🟢	🟢
Quotas	🟢	🔴
Request Transformation	🔴	🟢
Response Transformation	🔴	🟢
Pipeline Driven Conditional Actions	🔴	🟢
Pipeline Driven Expressions	🔴	🟢
Consumer Management	🔴	🟢
API Portal	🟡	🔴[5]
Plugins Framework	🔴	🟢
Open Source Ecosystem	🔴	🟢
Configuration Database	n/a	In-Memory Redis

1. Cloud Functions are deployed to Google's infrastructure behind the scenes. Google Cloud Endpoints can be deployed to any Kubernetes cluster. Express Serverless Platform deploys functions and gateways in Kubernetes as pods. Express Serverless Platform can also deploy serverless functions to GCP's Google Cloud Functions as well.

2. Auto REST scaffolding for Model Functions only

3. HTTP triggers only at this time (general pub/sub in roadmap)

4. Utilizes LoopBack.js OSS framework, 8 connectors displayed, 40+ connector developed by community that can be plugged in

5. API REST Testing Interface can be integrated into an existing portal.

Getting Started with the Express Serverless Platform

Getting started with Express Serverless Platform is dead simple.

LunchBadger offers a free 14-day trial of Express Serverless Platform with no credit card required.

The trial runs in LunchBadger's cloud so you don't need to have a public cloud account nor worry about installing anything on premises - simply sign up

The trial provides a complete walkthrough building a microservices application and its API through a series of guided and interactive steps - all in less than five minutes.

Once built, you'll trace your API workflow from public API Endpoints to your in-memory data source and be able to experiment with a live and running microservices application to see the value of the time savings and functionality it provides.

QuickStart

The following is an overview of the getting started experience provided by the self guided walkthrough.

- Deploy and use a Memory Connector to connect to an in-memory database
- Create and deploy a "Car" that will be a Model based microservice
- Connect the Car Model to the Memory Connector to read and write Car data
- Deploy and configure an API Gateway – an instance of Express Gateway
- Connect the Car Model to the API Gateway
- Expose the Car Model microservice as a Car API Endpoint that we can call through an API Request using cURL
- Deploy a Function called MyFunction that will be a "serverless" Function based microservice
- Connect the MyFunction Function to the API Gateway
- Expose the MyFunction microservice as a MyFunction API Endpoint that we can call through an API Request using cURL

Running Express Serverless Platform

Express Serverless Platform can run in any Kubernetes cluster on any cloud - private or public. It's a true multi-cloud solution.

Express Serverless Platform for GCP is Express Serverless Platform installed and ran on GKE within Google Cloud Platform.

Out of the box, Express Serverless Platform runs serverless functions in the Kubeless serverless engine in your Kubernetes cluster via the Serverless Framework.
By auto deploying serverless functions via the Serverless Framework, you can avoid vendor lock in and use any cloud provider or on premise solution.

Express Serverless Platform using the Serverless Framework will also allow you to run serverless functions in a public cloud provider's proprietary offering if you sign up for an enterprise level plan.

In this comparison, the cloud provider's offering for serverless would be Google Cloud Functions.

Getting Started with Google Cloud Functions

The easiest way to deploy Google Cloud Functions is using the Google Cloud Functions landing page and click “Try it free”.

You can check out Google’s cloud functions emulator that lets you run cloud functions locally and deploy them from the command line.

- Go to Cloud Functions Overview page in the GCP Console.
- Make sure that the project for which you enabled Cloud Functions is selected.
- Click Create function.
- Name your function.
- In the Trigger field, select HTTP Trigger.
- In the Source code field, select Inline editor.
- Use the default function provided in the editor.
- The function logs a message that you provide in a later step.
- Use the Runtime dropdown to select a runtime.
- Deploy the Function

Running Google Cloud Functions

Getting started with Google Cloud Functions means you must run your code on Google Cloud and manage your API using Google Cloud Endpoints. Cloud Functions execute in runtime environments based on open source languages and frameworks so you can build and test your functions locally using standard Node.js 6, Node.js 8 and Python runtimes and development tools.

Google Cloud Functions run in a fully-managed environment. Google handles infrastructure, operating systems, and runtime environments which all come with their own constraints. Additionally, each Cloud Function runs in its own isolated secure execution context, scales automatically, and has a lifecycle independent from other functions.

You can deploy Cloud Functions from your local machine or from your GitHub or Bitbucket source repository (via Cloud Source Repositories). Alternatively, if you use the GCP Console UI, you can also create and deploy functions from the Cloud Functions page in the GCP Console.

Further Reading: Deploying Google Cloud Functions

Pricing

Express Serverless Platform

Express Serverless Platform is offered as a free 14-day trial.

License plans come packaged with a number of users, microservice functions, and API gateways at a low cost that predictably scales as you realize the value of what the platform brings.

There are no hidden or extra compute fees for operating in Google Cloud Platform. Operating costs are transparently passed onto you by running in your account.

Google Cloud Functions

Google Cloud offers a free trial with $300 in credits for 1 year.
Function invocations are charged at a flat rate including functions invoked from an HTTP request or resulting from API calls. Invocations are charged at a per-unit rate of $0.0000004 per invocation.
The first 2 million invocations per month are free.
Additional charges for Compute Time, Networking and you can view pricing examples on the Pricing page.

Further Reading: Google Cloud Functions Pricing

Quick Reference Links

Express Serverless Platform

- Documentation
- Installation and Getting Started
- Free Trial

Google Cloud Functions

- Documentation
- Installation and Getting Started
- Commercial Support

For a more in-depth comparison, you can check out the just getting started experience comparison between Google Cloud Functions and Express Serverless Platform.