

Still in public beta.

Powered by Cloudflare

Vercel's Edge Functions are built on top of Cloudflare Workers. The runtime is designed for any similar provider, but we've chosen Cloudflare for now because they have an amazing product.

\ [Source](#)

[Only 20 regions currently supported](#). Functions can now be [deployed to a specific region](#).

Cannot use Node APIs in either edge functions or middleware.

Regular serverless function runtimes include Node, Go, Ruby and Python but edge functions run on a V8 runtime.

No local KV but there is a cache available for edge functions. However:

...directly connecting to your database over TCP does not work with [Edge Functions](#). With Edge Functions, you need to use the Web `fetch` API to access your data store over HTTP.

## Edge Functions

Edge functions are regular API functions that run on the `experimental-edge` runtime.

```
export const config = {  
  runtime: 'experimental-edge',  
};
```

The maximum size for an Edge Function is 1 MB, [including all the code that is bundled in the function](#).

Key difference between edge functions and middleware:

Edge Functions run *after* the cache, and unlike Edge Middleware, *can* cache responses

Also "Edge Functions *can* return responses"

New support for [OpenGraph \(OG\) image generation](#). These take advantage of the edge cache.

## Edge Middleware

Defined in a single `middleware.js` (or `.ts`) file in the root of the project. By default middleware will run on every single route unless you define a matcher config.

Includes the `@vercel/edge` package with `rewrite` to redirect the request to a different URL, `geolocation` and `ipaddress`.

The maximum duration for Edge Middleware execution is 5 seconds.

The maximum size for Edge Middleware is 1 MiB after compression, including your JavaScript code, imported libraries, and all files bundled in the function.