100 XP

Scale Azure Functions

3 minutes

In the Consumption and Premium plans, Azure Functions scales CPU and memory resources by adding more instances of the Functions host. The number of instances is determined on the number of events that trigger a function.

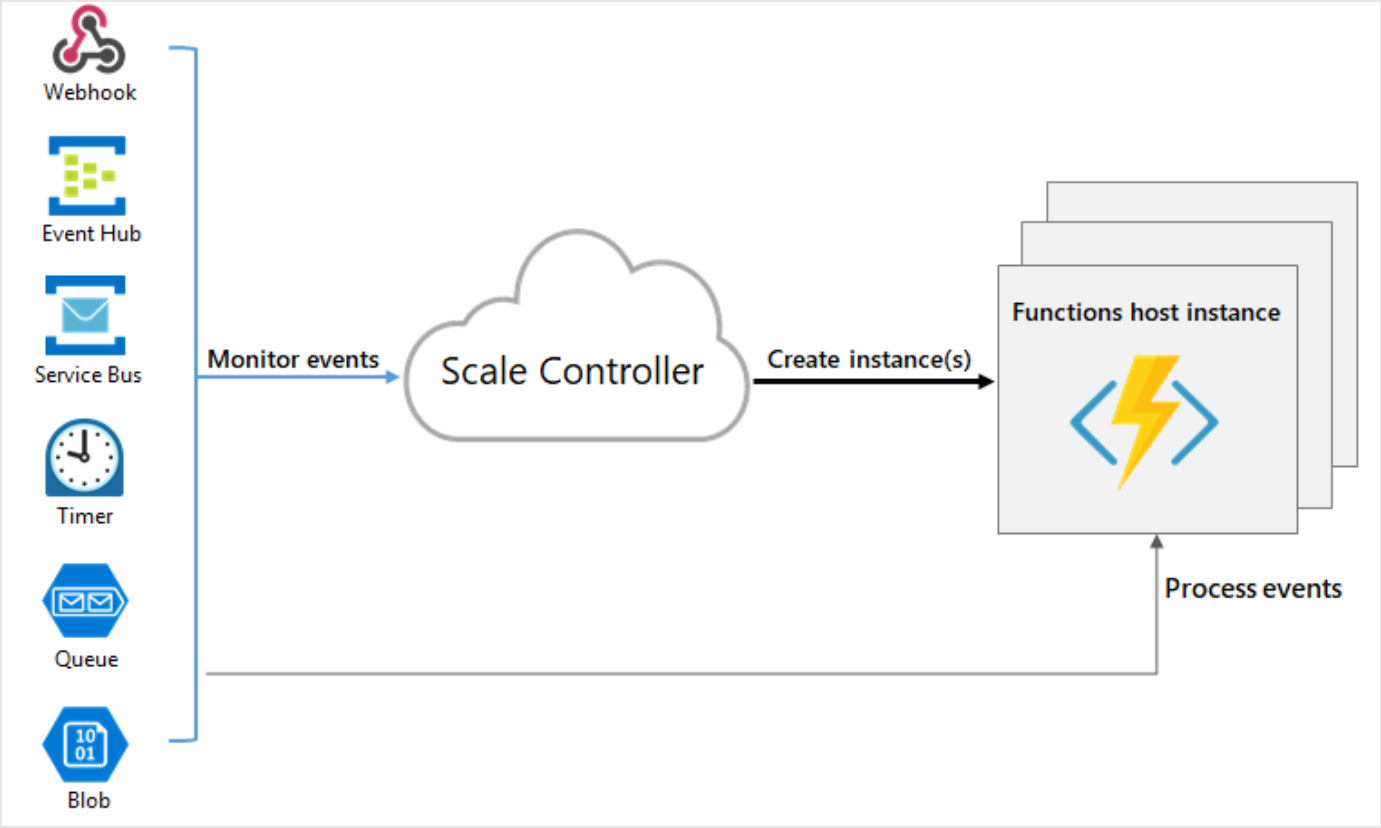
Each instance of the Functions host in the Consumption plan is limited to 1.5 GB of memory and one CPU. An instance of the host is the entire function app, meaning all functions within a function app share resource within an instance and scale at the same time. Function apps that share the same Consumption plan scale independently. In the Premium plan, the plan size determines the available memory and CPU for all apps in that plan on that instance.

Function code files are stored on Azure Files shares on the function's main storage account. When you delete the main storage account of the function app, the function code files are deleted and can't be recovered.

Runtime scaling

Azure Functions uses a component called the *scale controller* to monitor the rate of events and determine whether to scale out or scale in. The scale controller uses heuristics for each trigger type. For example, when you're using an Azure Queue storage trigger, it scales based on the queue length and the age of the oldest queue message.

The unit of scale for Azure Functions is the function app. When the function app is scaled out, more resources are allocated to run multiple instances of the Azure Functions host. Conversely, as compute demand is reduced, the scale controller removes function host instances. The number of instances is eventually "scaled in" to zero when no functions are running within a function app.



> ⓘ Note
>
> After your function app has been idle for a number of minutes, the platform may scale the number of instances on which your app runs in to zero. The next request has the added latency of scaling from zero to one. This latency is referred to as a *cold start*.

Scaling behaviors

Scaling can vary on many factors, and scale differently based on the trigger and language selected. There are a few intricacies of scaling behaviors to be aware of:

- **Maximum instances:** A single function app only scales out to a maximum of 200 instances. A single instance may process more than one message or request at a time though, so there isn't a set limit on number of concurrent executions.
- **New instance rate:** For HTTP triggers, new instances are allocated, at most, once per second. For non-HTTP triggers, new instances are allocated, at most, once every 30 seconds.

Limit scale-out

You may wish to restrict the maximum number of instances an app used to scale out. This is most common for cases where a downstream component like a database has limited throughput. By default, Consumption plan functions scale out to as many as 200 instances, and Premium plan functions scales out to as many as 100 instances. You can specify a lower maximum for a specific app by modifying the `functionAppScaleLimit` value. The `functionAppScaleLimit` can be set to `0` or `null` for unrestricted, or a valid value between `1` and the app maximum.

Next unit: Knowledge check

Continue >