Create endpoints for deployed Azure Machine Learning Studio (classic) web services

02/15/2019 • 2 minutes to read • +4

In this article

> **Note**
>
> This topic describes techniques applicable to a **Classic** Machine Learning web service.

After a web service is deployed, a default endpoint is created for that service. The default endpoint can be called by using its API key. You can add more endpoints with their own keys from the Web Services portal. Each endpoint in the web service is independently addressed, throttled, and managed. Each endpoint is a unique URL with an authorization key that you can distribute to your customers.

Add endpoints to a web service

You can add an endpoint to a web service using the Azure Machine Learning Web Services portal. Once the endpoint is created, you can consume it through synchronous APIs, batch APIs, and excel worksheets.

> **Note**
>
> If you have added additional endpoints to the web service, you cannot delete the default endpoint.

1. In Machine Learning Studio (classic), on the left navigation column, click Web Services.
2. At the bottom of the web service dashboard, click **Manage endpoints**. The Azure Machine Learning Web Services portal opens to the endpoints page for the web

service.

3. Click **New**.
4. Type a name and description for the new endpoint. Endpoint names must be 24 character or less in length, and must be made up of lower-case alphabets or numbers. Select the logging level and whether sample data is enabled. For more information on logging, see Enable logging for Machine Learning web services.

Scale a web service by adding additional endpoints

By default, each published web service is configured to support 20 concurrent requests and can be as high as 200 concurrent requests. Azure Machine Learning Studio (classic) automatically optimizes the setting to provide the best performance for your web service and the portal value is ignored.

If you plan to call the API with a higher load than a Max Concurrent Calls value of 200 will support, you should create multiple endpoints on the same web service. You can then randomly distribute your load across all of them.

The scaling of a web service is a common task. Some reasons to scale are to support more than 200 concurrent requests, increase availability through multiple endpoints, or provide separate endpoints for the web service. You can increase the scale by adding additional endpoints for the same web service through the Azure Machine Learning Web Service portal.

Keep in mind that using a high concurrency count can be detrimental if you're not calling the API with a correspondingly high rate. You might see sporadic timeouts and/or spikes in the latency if you put a relatively low load on an API configured for high load.

The synchronous APIs are typically used in situations where a low latency is desired. Latency here implies the time it takes for the API to complete one request, and doesn't account for any network delays. Let's say you have an API with a 50-ms latency. To fully consume the available capacity with throttle level High and Max Concurrent Calls = 20, you need to call this API 20 * 1000 / 50 = 400 times per second. Extending this further, a Max Concurrent Calls of 200 allows you to call the API 4000 times per second, assuming a 50-ms latency.

Next steps

How to consume an Azure Machine Learning web service.

Is this page helpful?

Yes No