A preemptible TPU node is a Cloud TPU node that runs at a much lower price than normal nodes. However, Cloud TPU might terminate (preempt) these nodes, at any time, if it requires access to the resources for another purpose.

You can create a preemptible TPU node using the Cloud Console, the `gcloud` command-line tool, or the `ctpu` utility.

Preemptible quota is only available for v2-8 and v3-8 TPU types. It is not available for TPU types with greater than 8 c
ample, v3-32, v3-128, etc.).

The preemptible status of the TPU is independent of any preemptible status of your VM instance. See the discussion of preemptible VMs and TPUs (#tpu-vm-preemptible) below.

Pricing for preemptible TPUs is significantly lower than for normal TPUs. For details, see the pricing page (/tpu/docs/pricing). You are not charged for TPUs if they are preempted in the first minute after you create them.

Quota for preemptible TPUs is generally higher, and is separate from the quota for normal TPUs. See the quota page (/tpu/docs/pricing).

As described in the quickstart guide (/tpu/docs/quickstart), you need a Compute Engine virtual machine (VM) in order to connect to a TPU. Note that the preemptible status of the TPU is independent of the preemptible status of the VM. You can define your TPU as preemptible and the VM as not preemptible, or the other way round. You can also define them both as preemptible.

The most likely combination is a **preemptible TPU** and a **non-preemptible VM**. Note the following points:

- The charges for the VM are likely to be low in relation to the charges for the TPU. The VM charges depend on the machine type you use. See the pricing page (/tpu/docs/pricing) for a simple example of the relative costs.
- Cloud TPU does not coordinate the preempting of the VM and the TPU. If you define them both as preemptible, the VM and the TPU can be preempted at different times.
- If Compute Engine preempts your VM, you are still charged for the TPU (unless the TPU is itself preempted). Note that the TPU is idle while the VM is preempted.
- Preemptible instances, both Compute Engine VM and Cloud TPU instances, are always preempted after they run for 24 hours. Certain actions (https://cloud.google.com/compute/docs/instances/preemptible#preemption_selection) reset this 24-hour counter.

You can use the `ctpu` command or the `gcloud` command to define a preemptible VM:

See the Compute Engine documentation on creating preemptible VM instances (/compute/docs/instances/create-start-preemptible-instance).

You can use the `ctpu` command or the `gcloud` command to check whether the Cloud TPU service has preempted your TPU:

To check whether the VM instance has been preempted, use the `gcloud compute operations list` command to get a list of recent system operations. Add a `name` filter to only display the instances you currently have running or add the `operationType` filter to only display resources that have been preempted. For example, use the following command to display only the instances with the specified instance name:

The following example displays only the resources that have been preempted:

For more details, see the Compute Engine guide (/compute/docs/instances/create-start-preemptible-instance#detecting_if_an_instance_was_preempted).

Make sure your application is resilient to restarts of the VM and TPU, by saving model checkpoints regularly and by configuring your application to restore the most recent checkpoint on restart.

The TensorFlow TPUEstimator API (https://www.tensorflow.org/api_docs/python/tf/contrib/tpu/TPUEstimator) takes care of saving and restoring model checkpoints for you. If you use TPUEstimator, you don't need to worry about saving or restoring checkpoints of your TPUs or VMs. Read more about using the TPUEstimator with Cloud TPU (/tpu/docs/using-estimator-api).

Best practice is to use the TPUEstimator with Cloud TPU as described above. However, if you want to investigate how to write the checkpoint saving and restoration functionality into your model yourself, see the following resources in the TensorFlow `tf.train` module:

- `tf.train.checkpoint_exists` (https://www.tensorflow.org/api_docs/python/tf/train/checkpoint_exists)
- `tf.train.latest_checkpoint` (https://www.tensorflow.org/api_docs/python/tf/train/latest_checkpoint)

- For a guide to creating your TPU resources, see the quickstart guide (/tpu/docs/quickstart).