

Anarion Technologies – Apache Beam

Note: Search product on Azure marketplace and click on "Get it now"

 Microsoft | Marketplace

Products ▾ 🔍 Search Marketplace

☰ All Products ▾ Categories ▾ Industries ▾ Partners ▾

Products > Apache Beam on Ubuntu26



Apache Beam on Ubuntu26

by Anarion Technologies

Virtual Machines

Starting at \$0.03 [Get it now](#) ☆ Save to my list

[Overview](#) [Plans + Pricing](#) [Ratings + reviews](#) [Details + support](#)

Production Ready VM on Ubuntu26.04 LTS + Free Support

Apache Beam on Ubuntu26 is a pre-configured virtual machine image designed for developers, data engineers, and organizations that require a fast, reliable, and scalable environment for building and executing data processing pipelines on Microsoft Azure. This image provides Apache Beam pre-installed on Ubuntu 26 along with Python and all required dependencies, enabling users to start developing and running pipelines immediately without the complexity of manual software installation and environment configuration.

Apache Beam is an open-source unified programming model that simplifies the development of both batch and streaming data processing applications. It provides a portable API that allows developers to write data pipelines once and execute them across multiple distributed processing engines. Apache Beam supports a wide range of runners, including Apache Spark, Apache Flink, Google Cloud Dataflow, and Direct Runner for local development and testing.

Click on Continue

Confirm your details to continue



Apache Beam on Ubuntu26

By Anarion Technologies

Plan

Apache Beam on Ubuntu26



Pricing

Starting at \$0.03/hour

Description

Apache Beam on Ubuntu26

You're signed in as **Market Place Products** (mp@anariontech.com).

[Edit your details](#)

 By clicking **Get it now**, I give Microsoft permission to use or share my account information so that the provider can contact me regarding this product. I agree to the provider's [terms of use](#) and [privacy policy](#) and understand that the rights to use this product do not come from Microsoft, unless Microsoft is the provider. Use of Marketplace is governed by separate [terms](#) and [privacy](#).

Get it now

Click on Create

Home

Apache Beam on Ubuntu26

Anarion Technologies



Apache Beam on Ubuntu26 [Add to Favorites](#)

Anarion Technologies | Virtual Machine

Subscription

Azure subscription 1

Plan

Apache Beam on Ubuntu26

Create

Start with a pre-set configuration

Want to deploy programmatically? [Get started](#)

Overview

Plans + Pricing

Usage Information + Support

Ratings + Reviews

Apache Beam on Ubuntu26 is a pre-configured virtual machine image designed for developers, data engineers, and organizations that require a fast, reliable, and scalable environment for building and executing data processing pipelines on Microsoft Azure. This image provides Apache Beam pre-installed on Ubuntu 26 along with Python and all required dependencies, enabling users to start developing and running pipelines immediately without the complexity of manual software installation and environment configuration.

Apache Beam is an open-source unified programming model that simplifies the development of both batch and streaming data processing applications. It provides a portable API that allows developers to write data pipelines once and execute them across multiple distributed processing engines. Apache Beam supports a wide range of runners, including Apache Spark, Apache Flink, Google Cloud Dataflow, and Direct Runner for local development and testing.

The platform is optimized for large-scale data processing workloads and offers a flexible architecture for handling structured and unstructured datasets. Developers can leverage Beam's rich set of transforms and SDKs to build ETL pipelines, stream analytics applications, event-driven architectures, and machine learning data preparation workflows. The included Python SDK and supporting libraries provide a ready-to-use development environment for creating and deploying modern data engineering solutions.

Creating a virtual machine, enter or select appropriate values for zone, machine type, resource group and so on as per your choice.

» Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure subscription 1

Resource group * ⓘ Demo [Create new](#)

Instance details

Virtual machine name * ⓘ Demo ✓

Region * ⓘ (US) East US

[Review + create](#) < Previous Next : Disks >

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

Disk options

OS disk type * ⓘ Premium SSD (locally-redundant storage) ▼

Encryption type * (Default) Encryption at-rest with a platform-managed key ▼

Enable Ultra Disk compatibility ⓘ ☐
Ultra disk is available only for Availability Zones in eastus.

Data disks

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

i Adding unmanaged data disks is currently not supported at the time of VM creation. You can add them after the VM is

Review + create

< Previous

Next : Networking >

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution. [Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ (new) Demo-vnet ▼
[Create new](#)

Subnet * ⓘ (new) default (10.1.0.0/24) ▼

Public IP ⓘ (new) Demo-ip ▼
[Create new](#)

NIC network security group ⓘ ☐ None
☒ Basic

Review + create

< Previous

Next : Management >

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Configure monitoring and management options for your VM.

Azure Security Center

Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads.

[Learn more](#) 

 Your subscription is protected by Azure Security Center basic plan.

Monitoring

Boot diagnostics  ☐ On
☒ Off

Enable OS guest diagnostics  ☐

Identity

[Review + create](#)

[< Previous](#)

[Next : Advanced >](#)

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more about tags](#) 

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name 	Value 	Resource
	:	12 selected 

[Review + create](#)

[< Previous](#)

[Next : Review + create >](#)

[Home](#) >

Create a virtual machine ...

✓ Validation passed

[Basics](#) [Disks](#) [Networking](#) [Management](#) [Monitoring](#) [Advanced](#) [Tags](#) [Review + create](#)

i Cost given below is an estimate and not the final price. For all your pricing needs, please use the pricing calculator. [↗](#)

Price

1 X Standard B1s
by Microsoft
[Terms of use](#) | [Privacy policy](#)

Subscription credits apply ⓘ

0.8652 INR/hr
[Pricing for other VM sizes](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

[< Previous](#)

[Next >](#)

[Create](#)

After Process of Create Virtual Machine. You have got an Option Go to Resource Group

Click Go to Resource Group

Copy the Public IP Address

SSH into Terminal and Run these Commands:

```
$ sudo su
$ sudo apt update
$ cd ../../
```

Activate venv:

```
$ cd ~/beam
$ source beam-env/bin/activate
```

Verify Apache Beam Version:

```
$ pip --version
```

\$ pip show apache-beam

\$ python3 -c "import apache_beam as beam; print(beam.__version__)"

```
testing@testing:~$ sudo su
root@testing:/home/testing# cd ../../
root@testing:/#
cd ~/beam
root@testing:~/beam# source beam-env/bin/activate
(beam-env) root@testing:~/beam# pip --version
pip 26.1.2 from /root/beam/beam-env/lib/python3.14/site-packages/pip (python 3.14)
(beam-env) root@testing:~/beam# pip show apache-beam
Name: apache-beam
Version: 2.74.0
Summary: Apache Beam SDK for Python
Home-page: https://beam.apache.org
Author: Apache Software Foundation
Author-email: dev@beam.apache.org
License: Apache License, Version 2.0
Location: /root/beam/beam-env/lib/python3.14/site-packages
Requires: beartype, cryptography, envoy-data-plane, fastavro, fasteners, grpcio, httplib2, jsonpickle, numpy, objsize, packaging,
uf, pyarrow, pyarrow-hotfix, pymongo, python-dateutil, pytz, pyyaml, requests, sortedcontainers, typing-extensions, zstandard
Required-by:
```

Run First Apache Beam Program

Create file:

\$ nano hello_beam.py

Paste:

import apache_beam as beam

with beam.Pipeline() as p:

```
(
    p
    | beam.Create(['Hello', 'Apache', 'Beam'])
    | beam.Map(print)
)
```

Save and exit.

Run:

\$ python3 hello_beam.py


```
root@testing: ~/beam
GNU nano 8.7.1 hello_beam.py
import apache_beam as beam

with beam.Pipeline() as p:
    (
        p
        | beam.Create(['Hello', 'Apache', 'Beam'])
        | beam.Map(print)
    )
```

Output:

Hello
Apache
Beam

```
(beam-env) root@testing:~/beam# python3 hello_beam.py
Hello
Apache
Beam
```

Thanks!!!