
eNlight Auto-Scaling – What is it?

Auto-scaling is an approach that automatically scales up or scales down the number of computing resources that are being allocated to your active application at any given point of time, based on its needs. eNlight Cloud empowers you with the ability to meet the predictable and sometimes unpredictable bursts in computing demands with absolutely no manual intervention. Auto scaling responds automatically and ensures that applications continue running smoothly, without any interruptions, no matter how erratic the resource usage patterns maybe. eNlight Cloud scales up and down your resources, i.e. the CPU and RAM as a percentage of resource consumption patterns. The scaling up is a steady but quick and automatic percentage of the pattern as required by your application, and so is the scaling down. Due to thick provisions of disk on storage “Disk Scaling” is not supported in eNlight however eNlight comes with an extraordinary option called “Elastic Storage” which enabled client to add multiple disks on the fly and also attach / detach them between multiple VMs without losing any data.

1. Pay-as-you-consume

As opposed to traditional IT billing techniques and other Cloud providers, eNlight is true to its commitment of pay as per Consumption. You will be charged only for the resources you use, thus completely eliminating hidden costs or surcharges. eNlight’s “pay-as-you-consume” model involves no surcharges or concealed costs. You pay only for the resources you use, as eNlight can intelligently compute resource usage. Also, the resource scaling is automatic, whether increase or decrease, and does not require any intervention, neither manual nor otherwise. eNlight automatically scales resources without having to wait for allocation or rebooting a user’s machine, and does not calculate cost on the resources added to the VM, but on the basis of the resources actually consumed, which is as accurate as it can possibly get.

2. Performance Models with eNlight Cloud

Cache memory is an extension of Random Access Memory (RAM) that a computer accesses more quickly than a regular Disk. The computer, when processing data, first looks in the cache memory that stores information from a previous reading of data and moves on to the much larger memory only if it is not able to retrieve it from the Cache. At times, the cache memory can grow very large unnecessarily reserving most of the RAM to store cache for performance; this is a normal scenario with Linux OS, to make the execution of a command that has been previously executed, easier. The amount of cache memory on a RAM can have huge implications on the performance of a machine. To make sure that you truly enjoy the freedom that eNlight brings to you, we offer two performance models depending on your preference:

Economy Model

With the economy based model, eNlight Cloud Storage devices are programmed to set a maximum limit to store cache on RAM, to stunt the growth of cache. Thus there is limited cache on RAM and hence the RAM on the server does not grow larger. The economy model thus focuses on limiting the cache to certain size and read data from the storage. As the name suggests the cost of a VM is kept low due to limited resource, compromising performance by accessing data from the disk and not from the RAM. This model suits those customers who has small size of database and doesn’t require high performance for their dynamic applications plus to keep the VM invoices low.

Performance Model

As a part of the Performance based Model, eNlight ensure a considerably large cache size for the storage devices. This is done to increase performance and to avoid repetition of fetching the same data from the drive several times, so that the future requests can be processed immediately. This model warrants more investment, since the RAM size needs to be heftier, but it leads to efficient and reduced response time and comes highly recommended. This model best suits those customers who have heavy traffic on their application and the size of the dynamic data is also very large.

