



Installing the 1.5 Release

- [Migrating to CML 1.5, on page 1](#)
- [Scalability, on page 2](#)
- [CML Server Component Versions, on page 3](#)

Migrating to CML 1.5



Important

Do **NOT** upgrade to CML 1.5 at this time if your installation uses the `virtl_local_ip` setting, normally used when running your CML server behind a firewall that does NAT. See the explanation for VIRLDEV-6250 below.

Performing a New Installation

If you do not already have CML installed, please use the updated CML 1.5 Installation Guide. Select the relevant instructions for your selected deployment option.

Migrating a CML Cluster Installation

A CML cluster is an installation that consists of a controller and one or more compute nodes. If your current CML installation is a cluster, you must perform a complete reinstallation of your cluster controller *and* your compute nodes. The installation procedures and configuration process for a CML cluster has changed significantly in CML 1.5 to fix limitations in the CML 1.3 cluster configuration process. The new cluster installation process is more robust while also providing more flexibility in configuring certain cluster settings.

Upgrade CML Client (VM Maestro)

You should update VM Maestro to version 1.5.0 build 510 or later. Older releases should still work since there were no changes in the file format or APIs from CML 1.2 or 1.3. However, running the latest version is recommended.

To download the new VM Maestro client:

1. Open a web browser and navigate to the CML host or virtual machine's IP address.
2. Login to the User Workspace Management (UWM).
3. Select **CML Server** from the menu that appears on the left.

4. Select the **Download** sub-menu.
5. Select **VM Maestro Clients** from the list of options.
6. From the list of files presented, download the VM Maestro client appropriate to your local platform (setup EXE for Windows, DMG for OS X, or zip file for Linux).
7. Install VM Maestro.

Once you have installed VM Maestro, you may want to update the node types shown in the Palette to match any changes on the CML server:

1. Launch VM Maestro
2. Select **File > Preferred > Node Subtypes**.
3. Click the **Fetch From Server** button.
4. Click OK.

Scalability

CML System Scaling

- Do NOT oversubscribe hardware resources at multiple levels.
 - It is possible to oversubscribe CPU and memory resources at both the CML System Configuration level and at the VMware ESXi level.
 - By default, CML applies an oversubscription factor of 2.0 for memory resources and 3.0 for CPU resources.
 - The recommended configuration is to use dedicated resources for the CML VM at the ESXi layer and control the hardware oversubscription via the **UWM > CML Server > System Configuration** pages.
 - System performance should be closely monitored and the following caveats should be taken into account when running large topologies to this scale.
- The ability to run larger simulations, approaching the node limit or the total CPU and memory capacity of the system, is truly a factor of available resources (memory, CPU, i/o speed, networking configuration, etc.). In particular, node types that are *heavier* than IOSv might or might not work, depending on available memory and CPU resources.
- Additional features (routing protocols, MPLS, etc.) might impact the ability to reach the node limit by using more shared resources of the simulation environment.
- At this time, when launching large simulations approaching the node limit or the system memory and CPU capacity, users must stagger the launch manually (see below for instructions on performing a staggered launch). Most of the Cisco node types place a higher load on the CPU just as the node boots up and loads its configuration. Cisco node types do not always react well to CPU starvation. The system generally functions properly with modest CPU oversubscription, but running simulations close to the total hardware capacity, especially with CPU oversubscription, and starting all of the nodes at once can lead to CPU starvation. A staggered launch will help to avoid this problem.

Staggered Launch of a Topology Simulation

When launching a large topology simulation (i.e., a topology that approaches the limits of the CML installation's hardware), it is recommended to avoid booting up every node at once when the simulation first starts. Instead, stagger the launch so that only a subset of the nodes is booting up at once. In the current release, topologies are *not* automatically staggered during launch. To perform a staggered launch of a topology simulation, the topology must indicate which nodes to start when the simulation is first launched. The back end will start the simulation, but it will only boot those nodes. The remaining nodes will remain in off / ABSENT state until they are manually started.

In VM Maestro, set the **Exclude from Launch** setting on nodes that you do *not* want to boot when the simulation first starts. As a starting point, pick a number of nodes equal to the number of physical cores, N, on your system. Try setting the “exclude from launch” setting on all nodes except for that “initial set” of nodes. Note that VM Maestro supports bulk editing: select multiple nodes at once in the topology editor, click the Properties view, and then edit a value to apply or remove the setting to all selected nodes. Once the **Exclude from Launch** setting has been applied to all but N nodes of the topology, the topology is ready for a staggered launch.

Start the simulation. Wait for just the initial set of N nodes to boot up and settle down. The nodes should at least go to ACTIVE - REACHABLE state, and it's probably best to leave them for a few minutes even after that to make sure that the configuration is loaded and the initial protocol processing is complete. In the running simulation view, select another batch of N nodes, right-click and select **Start Node** . Wait until that batch finishes booting up. Then start another batch of N nodes. Repeat until all of the nodes are booted up, running and ACTIVE - REACHABLE.

CML Server Component Versions

This release contains the following component versions:

Component Name	Version
OpenStack	Mitaka
CML_CORE	0.10.35.35
AutoNetkit	0.24.0/0.23.10
Topology Visualization Engine	0.17.28
Live Network Collection Engine	0.12.6
VM Maestro	1.5.0-510

Cisco Platform VMs

Platform	Version	Notes
IOSv	15.6(3)M	
IOSv L2	15.2 (03.2017)	
IOS XRv	6.1.3	

Platform	Version	Notes
IOS XRv 9000	6.2.2	New. This image is supported but NOT BUNDLED . It must be downloaded and installed separately.
CSR 1000v	16.6.1 IOS XE-based image	New
NX-OSv	7.3.0.1	Unsupported image based on the Nexus 7000 codebase.
NX-OSv 9000	7.0.3.I7.1	New. This image is supported but NOT BUNDLED . It must be downloaded and installed separately.
ASAv	9.8.2	New
CoreOS	899.13.0	
Ubuntu Cloud-init image	16.04.3	New

Linux Container Images

Container	Version
Ubuntu LXC	16.04
iPerf LXC	2.0.2
Routem LXC	2.1(8)
Ostinato-drone LXC	0.8