

# OpenStack: How do I upgrade Bright OpenStack?

*How do I upgrade Bright OpenStack?*

## Upgrading OpenStack and Bright Openstack Major Versions

Let's get right to it. Upgrading any OpenStack distribution, from one major version, to another, is not easy.

In fact, it's pretty challenging.

There are many reasons why that's the case. But the vast majority of them are, at the core, caused by the same single thing. OpenStack is infinitely customizable. The sky's the limit. That's one of the reasons OpenStack is so very popular. But it's also the reason that upgrading it can be problematic. The more customized your deployment is, the more challenging the upgrade is.

This article is in two sections.

[The first section](#) is about the current state of OpenStack, and the complexities involved in upgrading. This applies to pretty much all distributions of OpenStack out there. It explains in more detail why upgrading *any* OpenStack distribution is not easy, and also provides a rationale and context for the second section.

[The second section](#), in contrast, is specific to Bright OpenStack. It covers the recommended approach to upgrading Bright OpenStack.

Note: This article covers "upgrades", not "updates". By "upgrade" we mean moving from one major version of software (OpenStack, in this case) to another (e.g. from OpenStack Mitaka to Newton). That's in contrast to the term "update", which describes applying *minor* updates, and is rarely problematic in any way (e.g. update from OpenStack Keystone 8.0.0 to 8.0.2).

### Current state of OpenStack upgrades

Many OpenStack vendors out there claim that their OpenStack solutions allow for painless in-place upgrades of OpenStack. The reality is often different.

Behind a facade of alluring marketing messages, the reality is that such painless upgrades can normally only be carried out if you have stuck to the -- often very restrictive -- reference architecture of that vendor. It's unlikely you have. But even if you have, such upgrades would often require things like:

- purchasing consulting services from those vendors (or their partners),
- days, if not weeks, of preparation (depending on your scale)
- creating a lab mimicking your production cloud, where the upgrade procedure can be tested in isolation before rolling it out

# OpenStack: How do I upgrade Bright OpenStack?

Note, the list above is optimistic. That is, it describes the case where you have stuck to the vendor's reference architecture.

Most real-world OpenStack deployments, however, do not stick to the reference architecture of the vendor. Instead, they deviate from it, at least to some degree.

The larger the deviation, the more challenging it gets to upgrade OpenStack.

What are some of the common deviations from a reference architecture, then? Let's list just a few, along with some of the things which can go wrong with them during an upgrade.

- Different network layout (the most common deviation from a reference architecture)
  - New version was never tested with your specific NIC configuration, and may not work out of the box
- Use of non-default storage drivers for Cinder, Glance, or Nova. For example, the driver
  - may not (yet) be available in the newer version of OpenStack.
  - may have a different behavior in the newer OpenStack.
  - may even have its own upgrade procedure.
  - may have to be configured in a new way
- Customization for OpenStack services
  - The config files structure may have changed (it changes often). So, your customization to the new OpenStack config files may now be wrong.
  - The key=value pairs you modified may no longer be there in the newer version.
- You might be using an obscure feature of OpenStack (without even knowing it)
  - The obscure feature you're using might have not been given proper testing upstream, and simply doesn't work yet in newest OpenStack
  - The feature you are using was removed (e.g. did you know that OpenStack Ocata has stopped supporting popular (not even obscure) PKI Keystone tokens? Chances are you did not know that)

Note, this list is far from exhaustive.

# OpenStack: How do I upgrade Bright OpenStack?

Bright OpenStack comes with a certain version of Bright Cluster Manager, and with a certain version of Bright OpenStack packages.

For example:

- Bright OpenStack 7.3 comes with Bright Cluster Manager 7.3, and OpenStack Mitaka.
- Bright OpenStack 7.2 came with OpenStack Liberty (one version before Mitaka).

The naive (but not recommended) upgrade approach from Bright OpenStack 7.2 to 7.3 would therefore involve first upgrading Bright Cluster Manager, and then upgrading OpenStack in-place from Liberty to Mitaka. Similar steps would be required for an upgrade from 7.3 to 8.0, and so on. As mentioned in the earlier section, upgrading any OpenStack distro in-place is not easy. Such a naive approach to an in-place upgrade would run into some tough issues in practice. There are alternatives, however.

First of all, ask yourself if you really need to upgrade. If everything is still working fine, chances are you don't really need those new features from the latest shiny version of OpenStack. In which case, an upgrade might simply not be worth the trouble.

If, however, you do feel you must upgrade, Bright recommends one of the following approaches.

## [1. Remove and replace](#)

## [2. Mirrored Cloud](#)

The "Remove and replace" approach means you uninstall OpenStack, then upgrade the underlying Bright Cluster Manager in place (or reinstall it from scratch), and then redeploy and reconfigure (the newer) OpenStack.

If needed, you would backup any data you need to migrate from the old OpenStack cloud, and then later re-import it to the new cloud. Those might be things like, for example, some of your "pet" VMs, Glance images, maybe also Keystone user/role/tenant associations.

This is by far most reliable and easiest way to upgrade to a new version of OpenStack. You should go with this approach whenever you can afford a bit of downtime for all of the tenants of your cloud. The downtime is the only downside here, because of course, after removing OpenStack, and up until you redeploy it on the upgraded Bright Cluster Manager, the Cloud will not be accessible to the end users. You should also consider the remove-and-replace method if you don't have any spare hardware for creating the second head node, which is what the "Mirrored cloud" approach requires.

The length of the downtime in the remove-and-replace approach depends on the complexity of your cloud environment. It can vary from a few hours to several days. That said, this time can be further reduced with some additional planning and preparations

# OpenStack: How do I upgrade Bright OpenStack?

The “Mirrored cloud” approach involves deploying a smaller version of your Bright OpenStack in parallel, and next to, to your existing cloud, and doing so with only a few nodes to start with. Then, any resources can be gradually moved from the old cloud to the new cloud. As more and more resources (tenants) are gradually moved to the new environment, the hardware from the old cloud is also moved along with them. Once the new cloud is fully operational and contains all of the tenants, you can make the switch, and completely shut down the old environment.

This approach is recommended in situations where you cannot afford to bring down your entire OpenStack deployment down all at once for the upgrade.

Note, that in practice you don't need to have a lot of extra hardware in order to execute this approach. To begin with it, you only need an additional head node. You can start the process slowly, by taking out only a few hypervisor nodes out of your old cloud, and then proceed with moving additional nodes along with moving your workloads. In other words, the more workloads you move from your old cloud, the more spare hardware you will have to move to the new cloud.

**This approach** also gives you plenty of time to make sure that your new OpenStack cloud works properly with your environment and workflow. When finally the time is just right, it allows you to simply switch over from the old deployment over to the new one. Therefore, this approach is worth considering if your OpenStack cloud is heavily customized.

We recommend that before you opt going for the "mirrored cloud" approach, you first seriously consider going with the "Remove and replace" (or a variation of thereof). In many case that's simply enough, and will make things much easier.

A combination of both approaches is also a feasible alternative. It's good for complex and heavily customized cloud environments which can afford a bit of downtime. In such cases you would deploy a small version of the new Bright OpenStack in your lab, and take it slow, testing your existing workloads/customization in there. During this testing process you might discover that some changes are needed in your customizations and/or configuration of OpenStack. **Make note of those.** Once you've verified that everything is working, and are aware of any changes needed, you can do a quick "remove and replace" procedure of your existing cloud, and get back up and running relatively easily, with minimum downtime.

The thing to keep in mind with both of the above mentioned approaches is that they will result in a fresh environment for your tenants (users). This means all VMs, volumes, networks, etc. will need to be recreated. That said, if those elements were created using a proper cloud-native approach (using a easily reproducible approach, e.g. Heat and cloud-init), this part will be relatively easy. In other words, to make upgrades easier, avoid creating "pet" VMs. Aim at having VMs which are disposable, and easily recreateable.

## Summary

Upgrading any customized OpenStack distribution in place is not trivial. That's in contrast to what all the vendors out there are pitching to you. Yes, upgrading OpenStack in-place is doable, and it gets better and better with every new release of OpenStack. However, at this point it still requires significant amount of pre-planning, solid knowledge of OpenStack, and good execution.

So, at the moment, our recommended approach to tackling this problem is:

- simply avoid it, by not doing the upgrade, unless really necessary

# OpenStack: How do I upgrade Bright OpenStack?

- but, if an upgrade to new version needs to be done, then do not do it in-place on a live production system. Instead:
  - or
    - remove and replace (reinstall) OpenStack
    - set up a new version of OpenStack next to you existing version, and then slowly move your resources and workloads to the new environment.

**Bottom line:** right now, unless you have a small team of OpenStack experts, or you feel particularly adventurous, avoid doing an in-place upgrade of a live production deployment of OpenStack. In the future we plan to make in-place upgrades of OpenStack feasible by rolling out OpenStack services inside containers. This will simplify the overall upgrade procedure.

Unique solution ID: #1337

Author: Piotr Wachowicz

Last update: 2016-12-23 14:34