

Versions and Upgrades

Tim Theisen
Center for High Throughput Computing
HTC 24
2024-07-12

CHTC

HTCCondor
Software Suite

PATH

Overview

- PATH Versions
- Support Life Cycle
- Repositories and Tarballs
- Retention Policies
- CentOS 7?
- Upgrades

HTCondor and OSG Release Manager

I am responsible for overseeing the testing and release of HTCSS and OSG software.

I update the HTCondor Software on the CHTC and Build and Test pools.

I am on the lookout for problems in release candidates.

I view every commit that goes into the HTCSS code.

I write the release announcements and ensure that the documentation and version history is up to date for each release.

~~HTCondor and OSG~~ PATH Versions

Last year:

HTCondor 10.6.0 & 10.0.6, HTCondor-CE 6.0.0, OSG 3.6

It's hard to tell which versions should be used together. Different support life cycles.

With PATH we unified the version numbers and support life cycle

Current: HTCondor 23.8.1 & 23.0.12, HTCondor-CE 23.0.8 (soon 23.8.1 as well), OSG 23 & OSG 23-upcoming

What's in a Number?

We pattern our version numbers after semantic version numbers

<https://semver.org/>

- First number (Major version)
 - Year of initial release (e.g 23)
- Second number (Minor version)
 - 0 - LTS release
 - >0 - Feature release
- Third number (Patch version)
 - 0 - Testing
 - >0 - Could be: testing, final release, patch release

For HTCondor 24, we will release 24.0.x and 24.x.y around the same time (The divergence in x in 23 caused confusion)

Version Compatibility

We strive to maintain backward compatibility (however, older versions may not be able to take advantage of newer features)

Guarantees:

- 23.0.x is compatible with any 23.0.x (LTS, bug fix only)
- 23.x.y is compatible with any 23.0.x (We test for compatibility)
 - Main pool: feature CM, APs, EPs with a few AP and EP LTS nodes
 - Our Build and Test pool: LTS CM, feature AP, mix of EP nodes
- Since 24.0.1 is the culmination of the 23.x work, 24.0.x is compatible with any 23.0.x

In practice, the OSPool still has some 10.9.0 EPs and 9.0.x APs

HTCSS and OSG Support Life Cycle

Currently:

- 23.x is under active development and is fully supported
- 23.0 is getting regular bug fixes and is fully supported
- 10.0 is getting security and critical bug fixes only

When HTCondor 24.0 is released:

- 24.x will be actively developed and fully supported
- 24.0 will get regular bug fixes and will be fully supported
- 23.0 will get security and critical bug fixes only

So, each LTS version gets about two years of support

HTCSS Repositories and Tarballs

Four different places to get our software:

- **daily**: most current HTCondor build (passed unit tests, no integration tests), Use at your own risk, not in production
- **rc**: HTCondor build being tested in CHTC, (integration testing underway), Use at your own risk, not in production
- **update**: HTCondor build being tested on the OSPool, Feel free to test these builds
- **release**: Final HTCondor releases

The repositories definitions contain all four. However, everything except **release** is disabled or commented out.

HTCSS repo migration to OSG repos

HTCondor 23.0 **daily** repo migrates to **osg-development**

HTCondor 23.0 **rc** repo does not migrate

HTCondor 23.0 **update** repo migrates to **osg-testing**

HTCondor 23.0 **release** repo migrates to **osg-release**

HTCondor 23.x **daily** repo migrates to **osg-upcoming-development**

HTCondor 23.x **rc** repo does not migrate

HTCondor 23.x **update** repo migrates to **osg-upcoming-testing**

HTCondor 23.x **release** repo migrates to **osg-upcoming-release**

Retention Policies

As long as practical, we keep **release** repositories and tarballs for the last release of each LTS version online. (Currently, we have versions back to 8.8)

Previous versions in the manual are still online. Links from old release announcements still work. (Going back to 8.8 as well)

HTCondor LTS versions are retained in PyPI as well (going back to 8.8)

OSG 3.6 repositories will be available for at least a year.

Retention Policies (cont.)

The **daily** repositories and tarballs may be removed once a newer version appears.

The **rc** repositories and tarballs may be removed after two newer versions appear.

The **update** repositories and tarballs may be removed after two newer versions appear.

To clarify the roles of repositories and tarballs in HTCondor 24:

- The **daily** repository will be renamed **snapshot**
- The **rc** repository will be renamed **chtc**
- The **update** repository will be renamed **rc**

What about CentOS 7?

Many, many sites have not migrated away from CentOS 7.

We need to support our community, but still gently nudge them in the right direction.

Luckily, even though CentOS 7 repositories have been taken down, we can still build HTCondor for CentOS 7. However, external packages such as **apptainer** and **pelican** will not be updated in the tarballs.

So, CentOS 7 packages and tarballs will still be built for HTCondor 23.x and 23.0. However, CentOS 7 packages will not be available in HTCondor 24.0 and 24.x.

Upgrades: Feature vs LTS Versions

We encourage you to update to the latest feature version. We run this in production at CHTC. This is an ideal place to use new features and give us feedback.

We have LTS versions that only receive bug fixes. You should update annually with the help of the **condor_update_check** script.

Upgrades

When updating from one LTS to the next LTS, please read the section on upgrading:

<https://htcondor.readthedocs.io/en/latest/version-history/>

It enumerates new features and "gotchas"

Use the **condor_upgrade_check** script in the **condor-upgrade-checks** package. It's not dependent on a specific version of HTCCondor. It checks for issues that need to be addressed before the update.

Special Upgrade Considerations

When moving a CM to a new machine

- Bring \$ (SPOOL) /Accountantnew.log to the new CM
- If enabled, offline ads should be brought over as well

When moving an AP to a new machine

- Bring \$ (JOB_QUEUE_LOG) to the new AP
 - Otherwise, users will need to resubmit their jobs

Handling Configuration during Transition

The configuration language allows one to conditionalize the configuration based on HTCCondor version.

```
if version >= 24.0.0
    DO_X = True
else
    DO_Y = True
endif
```

Hopefully, you won't need to use this.

Preferred Upgrade Order

Is there a preferred order? Not really.

Common sense: Update the CM, update APs, then update EPs

Cautious upgrade: (I use this approach for my feature version testing)

1. Update a few test APs and EPs and run "Hello, World!" jobs on same version and cross versions (both feature and LTS).
2. Update CM and non-essential APs, and $\frac{1}{4}$ EPs (18 hr drain, via `shutdown_graceful_timeout`)
3. Update essential APs with manual restart and another $\frac{1}{4}$ EPs
4. Update $\frac{1}{4}$ EPs one day and another $\frac{1}{4}$ EPs the next day

Carefree upgrade: (I use this approach for my LTS testing)

- Update everything all at once

Questions?

Feedback welcome at tim@cs.wisc.edu or htcondor-users@cs.wisc.edu

