Backport Providers

Context: Airflow 2.0 operators, hooks, and secrets

We already have a lot of changes in the operators, transfers, hooks, sensors, secrets for many external systems, but they are not used nor tested widely because they are part of the **master/2.0** release.

As a part of Airflow 2.0, following AIP-21 “change in import paths” all the non-core interfaces to external systems of Apache Airflow have been moved to the `airflow.providers` package.

Thanks to that, the operators from Airflow 2.0 can be used in Airflow 1.10 as separately installable packages with the constraint that those packages can only be used in Python 3.6+ environment.

Installing Airflow 2.0 operators in Airflow 1.10

We released backport packages that can be installed for older Airflow versions. These backport packages will be released more frequently compared to the main Airflow 1.10.* releases.

You will not have to upgrade your Airflow version to use those packages. You can find those packages on PyPI and install them separately for each provider.

These packages are available now and can be used in the latest Airflow 1.10.* version. Most of those packages are also installable and usable in most Airflow 1.10.* releases but there is no extensive testing done beyond the latest released version, so you might expect more problems in earlier Airflow versions.

An easier migration path to 2.0

With backported providers package users can migrate their DAGs to the new providers package incrementally and once they convert to the new operators/sensors/hooks they can seamlessly migrate their environments to Airflow 2.0. The nice thing about providers backport packages is that you can use both old and new classes at the same time, even in the same DAG. So your migration can be gradual and smooth.

Note that in Airflow 2.0 old classes raise deprecation warning and redirect to the new classes wherever it is possible. In some rare cases the new operators will not be fully backwards compatible, you will find information about those cases in UPDATING.md where we explained all such cases.

Switching early to the Airflow 2.0 operators while still running Airflow 1.10.x will make your migration much easier.

Installing backport packages

Note that the backport packages might require extra dependencies. `pip` installs the required dependencies automatically when it installs the backport package but there are sometimes cross-dependencies between the backport packages. For example `google` package has cross-dependency with `amazon` package to allow transfers between those two cloud providers. You might need to install those packages in case you use cross-dependent packages. The easiest way to install them is to use “extras” when installing the package, for example the below will install both `google` and `amazon` backport packages:

```
pip install apache-airflow-backport-providers-google[amazon]
```

This is all documented in the PyPI description of the packages as well as in the README.md file available for each provider package. For example for `google` package you can find the readme in README.md.

You will also find there the summary of both - new classes and moved classes as well as requirement information.

Troubleshooting installing backport packages

Backport providers only work when they are installed in the same namespace as the ‘apache-airflow’ 1.10 package. This is majority of cases when you simply run pip install - it installs all packages in the same folder (usually in `/usr/local/lib/pythonX.Y/site-packages`). But when you install the `apache-`

`airflow` and `apache-airflow-backport-package-*` using different methods (for example using `pip install -e .` or `pip install --user` they might be installed in different namespaces.

If that's the case, the provider packages will not be importable (the error in such case is `ModuleNotFoundError: No module named 'airflow.providers'`).

If you experience the problem, you can easily fix it by creating symbolic link in your installed "airflow" folder to the "providers" folder where you installed your backport packages.

If you installed it with -e, this link should be created in your airflow sources, if you installed it with the `--user` flag it should be from the `~/.local/lib/pythonX.Y/site-packages/airflow/` folder.

Previous Next

Was this entry helpful?

☆ ☆ ☆ ☆ ☆