

Profiles & The AAT/PPI Systems

- [Introductory Comments](#)
- [Methods of Using AAT/PPI Commands](#)
 - [The vlapipe and almapipe Accounts](#)
 - [Running A Few Commands From Another Account](#)
 - [Specifying the Profile](#)
 - [Setting A Window Up For a Single Environment](#)
- [Caveats and Details](#)
 - [Shared Resources Among Environments](#)
 - [NGAS](#)
 - [Lustre](#)
 - [Cluster](#)
 - [NAASC Metadata Database](#)
 - [Site Differences \(NM vs CV\)](#)
 - [CAPO Files](#)
 - [What is that source `~pipeaccount/workflows/myprofile/bin/activate` command doing?](#)

Introductory Comments

The AAT/PPI has 3 parallel 'environments'. Each of these environments consists of a set of software which is deployed to dedicated servers in both NM and CV, and a set of configuration parameters (also called CAPO Profiles) for each deployment location for that environment. The 3 environments (and their associated profiles) are:

- [Production \(dsoc-prod, naasc-prod\)](#)
 - The current release of the AAT/PPI
 - No redeployments without an announcement to the `archive_issues` list
 - Web interface: `archive-new.nrao.edu`
- [Test \(dsoc-test, naasc-test\)](#)
 - Used for integration testing & validation of changes (features or bug fixes)
 - Also the home for pre-release candidates for release to production
 - Typically stable on day+ timescales
 - Major redeployments are announced on the `archive_test` list
 - Web interface: `archive-test.nrao.edu`
- [Development \(dsoc-dev, naasc-dev\)](#)
 - Developers' basic verification testing ground
 - Often subject to sudden redeployments of subsystems
 - Can have components from multiple branches of development deployed at one time
 - Web interface: `webtest.aoc.nrao.edu/portal/#/`
 - (Yes, I know, that name is unfortunate)

However, when it comes to the basic utilities for Data Analysts, the separation between these environments can be somewhat blurred: the tools are all installed next to each other within the vlapipe & almapipe account's home directories. This makes it easy to potentially use the wrong set of software (or mix software and settings from multiple environments) by mistake, and that's without accounting for bugs in the software itself. This document is meant to demonstrate a few suggested methods of accessing those tools in ways which might reduce the chance of confusion among the installations.

Methods of Using AAT/PPI Commands

The vlapipe and almapipe Accounts

As a convenience feature, these two accounts have had the `activate_profile` & `deactivate_profile` commands added to their Bash configuration. When given the name of a CAPO profile (`dsoc-prod`, `dsoc-test`, etc) `activate_profile` sets up a few environment variables and sets up your shell so that the corresponding set of utilities (and the correct supporting libraries) in your `$PATH`, so that they're easy to use.

You can `activate_profile` with a new argument to update to a new configuration (for instance, testing a new feature before release), or `deactivate_profile` to clean up what was done and go back to the basic account settings.

Running A Few Commands From Another Account

When running just a few commands, it might be easier to simply directly call the appropriate utility without resorting to the relevant 'pipe' account. In that case, pay particular attention to the directory from which you run the command (either by changing directory there or typing out the path). The differences between `/users/vlapipe/workflows/dsoc-prod/bin/myCommand` and `/users/vlapipe/workflows/dsoc-test/bin/myCommand` can (potentially) be not just the details of the software itself, but also any differences in the profile's parameters.

Specifying the Profile

You can provide most (if not all) the AAT/PPI utilities with a `-P` option to define the CAPO profile you want to use. It's typically a good idea to do so, as you likely do not have the `CAPO_PROFILE` environment variable set, and not all the utilities are capable of guessing the profile from their location.

Often, you'll want to ensure that the profile you provide matches that of the utility you're running, but there is one exception: In version 3.7.0 and later, `restoreToCache` can be provided one of the VLASS profiles (`vlass.test`, `vlass.w7`, etc) to run the restore workflow with appropriate settings for that project.

Setting A Window Up For a Single Environment

For running a larger set of commands for a single environment, there is an alternative to using the `vlpipe`/`almapipe` accounts (which is particularly helpful if you need access to directories where those accounts don't have permissions). This is not as comprehensive as using `activate_profile`, so it is recommended that you don't switch between environments using this method.

It is possible to approximate what is done via `activate_profile` with:

1. `export CAPO_PROFILE=myprofile`
2. `source ~pipeaccount/workflows/myprofile/bin/activate`

Where **pipeaccount** is `vlpipe` or `almapipe`, depending on where you're working, and **myprofile** is the appropriate name from the discussion of the 3 environments/installations in the introduction.

Caveats and Details

Shared Resources Among Environments

Each deployment environment has a copy of the AAT/PPI code (with underlying software infrastructure), and its own copy of the AAT/PPI metadata database. That separation reduces concerns about the size of the connection pool for the metadata database, and each environment having a separate messaging infrastructure insulates them from one another. However, some resources are shared among all environments.

NGAS

There is only one official file-storage system in each location. Thus all 3 environments draw from the same set of servers. If multiple environments are in heavy use, it's possible to overload the system.

Lustre

The high speed shared filesystem is another shared resource. Each environment has a particular sub-area in which it performs its processing. Each of the areas is parallel to each other (much like the software installations underneath the `*pipe` accounts) on the associated site's lustre system (for instance, `/lustre/naasc/web/almapipe/pipeline/naasc-test` is where the Test environment works in CV).

Cluster

The computing resources are shared among environments, as each site only has one set of dedicated processing nodes. While they can (and sometimes are) configured to submit to separate subsets of these clusters, the entire set of computers to which jobs are submitted is limited. This is often particularly noticeable in CV during periods of heavy ALMA data processing (particularly around pipeline validation testing).

NAASC Metadata Database

There is only a single database about ALMA data in CV. All 3 of the AAT/PPI environments draw information from it using the same set of Read-Only credentials. So far, however, there not been severe issues with connectivity.

Site Differences (NM vs CV)

The entirety of the software making up the AAT/PPI is deployed upon that environment's server in NM, with only a subset deployed to its companion machine in CV. There is a separate copy of the utilities deployed to the `almapipe` account, and there is an independent workflow system on the CV server. However, as most of the AAT/PPI utilities are just sending system messages to provoke action, the fact that the message processing system (`amygdala`) is located in NM may cause additional delay for some commands.

In addition, some of the services which facilitate processing are deployed only to NM, so that information requests (like those made for image ingestion) may be slower.

CAPO Files

The configuration details for the AAT/PPI are kept in text files in `/home/casa/capo`. This directory is replicated to CV via `rsync` on a cron job, so it is not immediately updated after an edit. There are a set of properties for each site (NM vs CV) and environment (Development, Test, Production). Included in that directory are a set of files for the VLASS project, containing configuration specific to their processing needs.

These properties define information like server names, where to process data, and what version of CASA to use as a default. They are editable by the `vlpipe` account, so those values which are of interest can be readily modified by the Data Analysts. It is possible (even likely) that some of those variable names may change as the system evolves. It is important to remember that the copy of the files in NM take precedence and will periodically overwrite their counterparts in CV. If you don't have access to the `vlpipe` account, contact one of your fellow DAs or the SSA team.

What is that `source ~pipeaccount/workflows/myprofile/bin/activate` command doing?

In order to function properly, the AAT/PPI utilities (both internal and external) require a stable, defined group of python libraries. In order to provide that, these tools are deployed within [Python Virtual Environments](#). These environments evolve over time both as new tools are added, but also as the interactions (either internally or with external libraries) change. There will be times when two separate environments will be entirely incompatible, which is why we try to provide simplifications for accessing a coherent set of utilities.