# AAT/PPI Automated ALMA updates

The AAT/PPI uses a set of processes to handle keeping the system up-to-date with ALMA observations and calibrations. The software processes are part of the amygdala package, and run largely without developer interaction.

## General Pattern:

This system was the initial example of this pattern in the AAT/PPI, and has since been re-used for ingestion of VLBA data and extended to also work with ALMA calibrations. Using a tracking table within the metadata database (alma_reingestion_queue) and a dedicated rabbitmq queue, there are a pair of programs in amgydala which do the following:

1. QueueLoader:
    a. look at the latest item we have in the tracking table
    b. find any newer observations
    c. update the tracking table with newer items
    d. find everything that is waiting to run
    e. send a rabbitmq message for each item
    f. wait as long as your told (CAPO setting)
2. QueueRunner:
    a. set up a listener to archive.events for completion notices
        i. when one of our jobs completes, remove it from the list & free the thread
    b. wait for a free 'thread' (there's a limit we can control with CAPO)
    c. get the next 'run this' message from rabbitmq
    d. launch the appropriate workflow for that message
    e. record what we just started

The VLBA version of the system limits what will be run concurrently (to avoid self-interference in creating the basic project information), but it follows the same general idea. For VLBA the system uses a separate tracking table and a modified QueueRunner, but follow the same basic philosophy.

## Execution Block Basics:

Newer observations are found via queries to the AQUA_V_EXECBLOCK view, comparing the ENDTIME value to what the latest observation in the tracking table.

When a new ASDM comes up for reingestion, a workflow is launched in CV where the ASDM XML files are extracted from NGAS at the NAASC and parsed via the ingest script to populate AAT/PPI execution_blocks & associated tables.

## Calibrations:

Newer calibrations are found via queries to the ASA_PRODUCT_FILES table, comparing the CREATION_DATE to the latest calibration listed in the tracking table. Once a new or updated MOUS is identified, we perform the following checks:

- Does this MOUS have any archived science products?
    - ( select count(*) from ALMA.ASA_PRODUCT_FILES where FILE_CLASS='science' and ASA_OUS_ID=?; ) > 0
    - This condition indicates that the calibration has been accepted, and can be treated as official
    - If not:
        - Defer this MOUS for later evaluation (next time the system does a check)
- Does this calibration contain scripts to perform custom recalibration?
    - ( select count(*) from ASA_PRODUCT_FILES where ASA_OUS_ID='.....' and FILE_CLASS='script' and NGAS_FILE_ID LIKE '%scriptFor%Cal%'; ) == 0
    - This condition indicates that the DAs had to take an active hand in the calibration of this data.
        - If that's the case, this calibration is unsuitable for the AAT/PPI automated restore process. The execution blocks for this MOUS are marked 'Do Not Calibrate' to avoid further attempts to pipeline-calibrate this data.
            - Note: This should not be a terribly common case. The NAASC is able to accept pipeline-calibrated results roughly 85% of the time
        - If this is a purely pipeline-generated calibration, then it is placed in the queue for ingestion.

Note: Because the ATT/PPI did not initially track any of the ALMA structure information, early on after the release of version 3.6 it is possible for a calibration to be evaluated before a constituent ASDMs of the MOUS is reingested to populate the structure. In that case, the processing of the calibration information is deferred until the MOUS is recognized in the alma_ouses table.

When a MOUS comes up for reingestion, a workflow is launched in NM which performs queries to populate calibrations & associated tables. The ASDMs for the calibrated MOUS are then marked 'Calibrated' to indicate that a restore may be performed.