# CRDS

# Procedure for

# Routine Reference File Submissions

## Todd Miller

## 2015-10-05

# Nominal Procedure for New References

This document describes the process of delivering new references to CRDS, archiving them, activating new default CRDS rules, and synchronizing the rules, references, and defaults to the pipeline’s CRDS cache. This procedure encompasses the process for nominal cases where CRDS is capable of performing automatic ruels updates. Related but different processes are required to perform type additions or custom rules updates.

## File Submission

This section describes the steps performed by a file submitter which result in files being checked, added to the CRDS catalog and rules, and finally linked to the CRDS file delivery directory for pickup by the CRDS pipeline and eventual addition to the STScI archive.

#### Deliverer Login to Server

The file deliverer browses to <https://jwst-crds-b5it.stsci.edu> and clinks on the green doorway arrow image in the top right of the web page. On the login page, the submitter selects the instrument for which files are being delivered and enters username “submitter” with password “submit\_files”. After authenticating the submitter is redirected to a restricted version of the top-level page that lists available restricted functions in the right column of services. Next to “Restricted Services” the user will see their username “submitter” and the name of the instrument they have locked. At the top right of the page the lock count down timer is displayed under the login icons, nominally 4 hours. If the count down timer expires, any pending submissions are cancelled, the user is logged out, and the lock is released.

#### Batch Submit References

Under “Restricted Services” the submitter clicks on link (6), “Batch Submit References. This service is used for routine reference file deliveries with corresponding automatic rules updates.

##### Upload Files

The submitter opens the “Upload Files” accordion, clicks “Add Files” to select and add files from the submitter’s file system to the upload list. After selecting files, the submitter MUST click “Start Upload” prior to completing and submitting the remainder of the form.

##### Derive From Context

The submitter chooses the set of existing rules to which the submitted reference files will be added to create a new set of rules. The default selection of “Editing Context” tracks the last routine delivery and is the nominal place for future submissions. After each submission, the Editing Context is adjusted to the newly created set of rules to serve as the basis for any further submissions.

##### Change Level

The submitter selects a change level, with SEVERE indicating that reprocessing is required for datasets for which the references apply, and TRIVIAL indicating that reprocessing is not required.

##### Creator

The submitter types in the name of the person who originally created the references, typically not the person submitting them. The “Creator” field is nominally the same as the AUTHOR field in reference files, but can be adjusted for multiple authors or other special circumstances.

##### Description

The submitter types in a brief description of the references that will be saved in the CRDS catalog and visible in the “Recent Activity” log. For single deliveries, the description is also automatically propagated to the Set Context page’s description field that will appear in the context history.

#### Review and Confirm

The output of the reference submission consists of results from the CRDS certifier for each files followed by the differences between the original “derive from” rules and the rules generated by the submission. Certifier errors result in automatic cancellation of the delivery. Certifier and difference warnings should be carefully reviewed prior to confirming the submission.

#### Final Results

After the submission is confirmed and files are made available to the CRDS pipeline for delivery to the archive, a final results page is displayed which shows information about the submission and resulting files.

#### Delivery Output

The output of the submission process from CRDS is the creation of file links in the CRDS file delivery directory that is shared by the CRDS pipeline. Each delivered reference and generated rules file is linked to the delivery directory. A manifest file called a “catalog” that has name extension “.cat” lists the base name of each delivered file, one file name per line.

#### Logout

The submitter clicks the red “out door” icon at the top right of the page to log out and release the instrument lock to other team members.

#### Limitations

The CRDS file submission system is limited to deliveries of 25G and/or 100 files or less.

## CRDS Pipeline Delivery

If the archive’s CRDS Pipeline’s Poller process is running, the CRDS pipeline checks for a new .cat file roughly once every 30 minutes. .cat files (and their associated rules and references) are delivered to the archive according to the order of their serial numbers. Deletion of the “.cat” file (as well as all corresponding file links) is the acknowledgement from the CRDS pipeline to CRDS that the delivered files have been archived and made available for download from the archive’s online cache and reference file web service. The CRDS Pipeline processes exactly one .cat file per activation, so at most one .cat file per polling cycle, nominally 30 minutes.

## Pipeline Context Activation

To set the global default operational context shared by all CRDS users, the I&T operator logs into the CRDS server and activates the new context using the “Set Context” restricted service web page.

#### Operator Login to Server

The file deliverer browses to <https://jwst-crds-b5it.stsci.edu> and clinks on the green doorway arrow image in the top right of the web page. The operator logs into the CRDS server as username “operator” with password “set\_context. The operator should select instrument “ none” to avoid blocking submitter inputs on an instrument.

#### Set Context

From the top-level page, the operator clicks the restricted services (5) Set Context link to open the Set Context page. Under routine circumstances, the operator leaves the selected Context Type as “operational” and does not adjust the value of the Set To context. With lock-step submissions and activations, the Reason For Change text field defaults to the description given by the file submitters. The operator can adjust the“Reason For Change” as necessary. The operator clicks the “Set Default Context” button to effect the change in centralized CRDS server default context.

#### Logout

The operator clicks the red “out door” icon at the top right of the page to log out.

## Pipeline CRDS Cache Synchronization

Delivering files or running Set Context on the web server only changes the state of the centralized CRDS server and STScI archive. Typically pipelines are run in CRDS “local” mode fully decoupled from the CRDS server during pipeline calibrations. Consequently, when new files are delivered and/or the default CRDS context is changed on the server, the pipeline’s CRDS cache must be synchronized to the CRDS server. Synchronizing the CRDS cache effectively performs three things: (1) it distributes new reference files to pipeline local storage. (2) It distributes CRDS rules to pipeline local storage, which is in effect a kind of database replication. (3) It updates various pieces of CRDS configuration information including the default context used by the pipeline.

#### File Availability and Default Context

Submitted files are not available for download until after the CRDS pipeline and archive have finished delivering the files to the archive’s online cache. During the interim, the CRDS server considers the files to be “unavailable” and will not list them as part of any abstract groups of files such as the context specifications –-all or --last 20. Similarly, until the files are fully available, they cannot be chosen for the default context. Once files are fully archived and available for download, the CRDS sync tool includes them in abstract groups and can download them prior to the update of the CRDS server’s default context. Downloading files in advance of Set Context ensures that the cron\_sync following the Set Context will complete as rapidly as possible without performing related file downloads.

#### Pipeline cron\_sync setup

The cron\_sync tool is configured with knowledge about a specific CRDS server and the location of the pipeline’s CRDS cache. It is configured in the shell roughly as:

% setenv CRDS\_SERVER\_URL <https://jwst-crds-b5it.stsci.edu>

% setenv CRDS\_PATH <location of pipeline’s private CRDS cache>

#### Operator runs cron\_sync tool

CRDS provides the command line cron\_sync wrapper script intended to configure crds.sync specifically for pipeline operation. Since there are a variety of situations for running the cron\_sync tool, the cron\_sync script requires additional parameters that are provided by the pipeline in a pipeline wrapper script. cron\_sync is typically run in the pipeline as follows:

% cron\_sync –-last 20 –-fetch-references

Run in this fashion cron\_sync assures that the last 20 contexts defined on the CRDS server are synchronized to the local cache, including all references referred to by those contexts. Files already in the CRDS cache are not downloaded again.

#### Contexts Specifier

The context specifier (e.g. --last 20) nominally determines the rules which are downloaded. The switch --last 20 specifies that the most recent 20 CRDS contexts should be synchronized to the pipeline’s CRDS cache. Typically, only files from the most recent submission are missing and –-last 1 would suffice. Using --last 20 provides margin for a backlog of file submissions between running Set Context while maintaining a more stable runtime for cron\_sync. Using –-all instead of --last 20 is more comprehensive and syncs all available rules, but has an increasing runtime with each context added. Then intent of –-last 20 is to maintain a pipeline cache that contains all available rules while retaining a reasonable and fairly constant runtime.

#### References Specifier

By default the cron\_sync wrapper does not synchronize reference files. Adding the switch --fetch-references causes cron\_sync to download all reference files referred to under the contexts defined by the contexts specifer.

#### cron\_sync locking

Because cron\_sync is designed to be run periodically as a cron\_job, and because synchronization duration is driven by the number of files and deliveries being processed, cron\_sync uses a lock to prevent multiple instances of cron\_sync from running simultaneously; running a second instance of cron\_sync while a first instance is still running will result in the second instance immediately exiting. After downloading the requested files, cron\_sync updates the local CRDS cache with a record of the default context.

#### cron\_sync customizations

The purpose of cron\_sync is to provide an abstract encapsulated configuration of crds.sync intended to support pipeline’s. Nervertheless, the properties of CRDS which are customized by the wrapper are currently:

% setenv CRDS\_LOG\_TIME 1 # add time to log messages

% setenv CRDS\_READONLY\_CACHE 0 # override CRDS cache write protection

% setenv CRDS\_DOWNLOAD\_MODE plugin # use default CRDS plugin file downloader

% setenv CRDS\_CLIENT\_RETRY\_COUNT 60 # retry failed network calls 60 times

% setenv CRDS\_CLIENT\_RETRY\_DELAY\_SECONDS 10 # wait 10 seconds per retry

#### Reference Download File Server

cron\_sync downloads rules and references from a particular file server, nominally one provided by the archive. The download server utilized by cron\_sync is defined on the CRDS server and messaged to the CRDS client and cron\_sync tool. Until recently the CRDS server itself has been used as a static web file server to support cron\_sync reference and rules downloads. Starting with build-5 I&T the archive has provided a static file server (DMS-667) which supports downloading CRDS rules and references from the archive’s online cache. The reference and rules server in actual use can be verified by running cron\_sync with the switch –verbose. Use of the archive file server requires successful delivery of rules and references not just to the CRDS server, but through the entire delivery and archive ingest process. After delivery to the CRDS server, but prior to successful archiving, CRDS references and rules are nominally unavailable for download.