Pipeline Testing tracking sheet

One of the main tasks of the TSO WG on FY2020 is to perform testing of the steps made by the JWST Calibration Pipeline for TSO-like data. This tracking sheet contains all the information regarding pipeline testing by the members of the TSO WG. Nestor Espinoza and Sarah Kendrew will be populating this page heavily in the weeks of July 20 and 27, 2020, for NIRISS and MIRI TSO testing, respectively. Nikolay Nikolov and Tony Keyes will be populating at their own pace for NIRCam and NIRSpec, respectively.

1.1. General guidelines for Pipeline Testing

1. Please, test every step identified below. If for some reason you don't have time to test a step, let Nestor Espinoza and Sarah Kendrew know.
2. Use the comments in the tables to post brief comments about the status of that testing step (think of it as a quick overview for others to know what is going on). For extended commentary, or details on how you performed the testing, use the text below the tables.
3. **If a bug is found, please report it in Jira as a "Bug" Jira ticket.** Please copy the links to those tickets in the table below under the comments.
4. **For algorithm improvements/enhancements, please file an "Improvement" Jira ticket.** Please copy the links to those tickets in the table below under the comments.

In general, if bugs are found, the best would be to discuss them first in one of our meetings before going ahead and submitting them (unless this is high-priority, in which case we could schedule a special meeting for this). This is left at the discretion of each TSO WG member.

1.2. General comments

Before jumping to read the below table, some important comments to keep in mind while reading them:

- NIRISS testing is being done using awesimsoss NIRISS/SOSS simulations of K2-141 (which has a short transit duration, hence the selection). Care must be taken as this is the first end-to-end simulation of awesimsoss (which is under active development). In-progress notebooks of each stage testing can be found here.
- MIRI testing is being done using MIRISim simulations of MIRI/LRS.
- NIRCam testing is being done using Mirage simulations of imaging and grism time series (350 frames of 5 groups, i.e. 70 exposures) of a modified version (deeper transit) of WASP-79b. Notebooks for the testing are available here.
- NIRSpec testing is being done on CV3 data.

1.3. Stage 1: Detector Processing TSO testing

Pipeline step	NIRISS status	NIRISS comments	NIRCam status	NIRCam comments	MIRI status	MIRI comments	NIRSpec status	NIRSpec comments
group_scale	DONE	The step is supossed to do nothing for NIRISS /SOSS (here, a group = 1 frame). Indeed, nothing changes when running this step, except for the length of the ASDF element in the HDU list object, which is normal.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. An CRDS - ERROR - Error determining best reference for 'pars-groupscalestep' = Unknown reference type 'pars-groupscalestep'. This seems to be a warning, rather than an error.	**N/A**		DONE	Runs without issue.
dq_init	DONE	This step correctly initializes the data-quality flags, as it is supposed to do, according to the reference mask in CRDS.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. The data quality flags are initialized correctly. This step has sufficiently been tested by unit testing.	DONE		DONE	Runs without issue.
saturation	DONE	This step does what is expected to do: it captures saturated pixels and flags them in the GROUPDQ masks. The simulations used for this validation had saturated pixels and they were correctly identified (flagged) by the pipeline.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. Saturated pixels are identified and flagged in the GQ masks correctly. This step has sufficiently been tested by unit testing.	DONE	done for imager. only high level check for LRS.	DONE	Runs without issue.
superbias	DONE	Tested my own bias substraction using reference files and got exact same results.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. This step has sufficiently been tested by unit testing.	**N/A**		DONE	Runs without issue.

reset	N/A		N/A		DONE	This was tested on a TSO imaging file separately since this step does not currently run by default. The test passed with no issues.	N/A	
linearity	DONE	Tested linearity correction using reference files and got exact same results.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. The step performs what is expected and has sufficiently been tested by unit testing.	DONE	sufficiently covered by unit testing.	DONE	Runs without issue.
rscd	**N/A**		**N/A**		DONE	Step currently flags frames in integrations higher than the first as 'DO_NOT_USE' in the groupdq array. This has been tested successfully.	**N/A**	
dark_current	DONE	Dark frames perform expected corrections on the data.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. Dark subtraction works well.	DONE	sufficiently covered by unit testing	DONE	Runs without issue.
refpix	DONE	Currently, does not do anything to NIRISS/SOSS simulations, as reference pixels there are set to zero. When injecting a fake strain of reference pixels got some deviations that most likely have to do with DMS-detector frame rotations.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. The step performs what is supposed to do and has sufficiently been tested by unit testing. Still, I find this step to correct much less for subarrays (i.e. using half of the reference pixels) compared to a subtraction of the median value. This is not a suggestion to improve this for exoplanets, as the 1d cbc spectral extraction would do a better correction. Would be good for the imaging data though.	DONE	This step is sufficiently tested in unit tests. It is skipped for MIRI subarray data so only applies to FULL array imaging TSOs, and note that MIRISim's implementation of Refpix is different from the pipeline step, so may give unexpected results.	DONE	Runs without issue.
jump	DONE	Step produced no errors. However, currently detecting more jumps than it should — will revisit once some errors on the SOSS simulator are addressed (see below).	DONE	Runs without a crash, but identifies too many pixels as cosmic rays or bad with the default threshold of 4 sigma. Results improve if the threshold is increased to >15 sigma. NOTE: This step names files 'jump', i.e. without the root 'step'. This differs to all of the remaining steps in stage 1.	DONE	high level testing is done. Inserted cosmic rays of varying strengths were detected (based on threshold) and flagged neighboring pixels (based on level above background)	DONE	Runs without issue.
ramp_fitting	DONE	The pipeline was tested against an OLS ramp fit, and pipeline does exactly as intended.	DONE	Runs well without a crash with the run() and call() commands for both imaging and spectroscopy TS. Ramp fitting (including OLS) has sufficiently been tested by unit testing and performs as expected. The run() command accepts weighting = "unweighted", despite the optimal is the only option and produces different results. The call() command has no options for weighting.	DONE	Well covered by unit tests. May be a candidate for more detailed testing when time.	DONE	Runs without issue.

gain_scale	DONE	This step only applies to NIRSpec data. Because no GAINFACT is found in the headers, this step is properly skipped by the pipeline.	DONE	Only applies to NIRSpec and skipped for NIRCam.	DONE	Documentation states 'This currently only applies to NIRSpec exposures that are read out using a subarray pattern, in which case a gain setting of 2 is used instead of the standard setting of 1.' This step is properly skipped for MIRI data.	DONE	Runs without issue.

Extended comments by instrument branch:

- NIRISS: For the jump detection algorithm, the pipeline seems to be detecting a large fraction of jumps (~20% per integration). This is apparently a known issue in other instruments as well, and is due to aggresive parameters in the reference files. These will need to be calibrated with on-sky data.
- NIRCam
- MIRI:
- NIRSpec: **Build 7.5:** The only NIRSpec Level 1 entry for DMS Build 7.5 is that all steps are verified to run without issues. NIRSpec does not have specific validation tests for Level 1 as this pipeline is common to all instruments and there are several tests already included within the pipeline code that have been verified previously by the instrument teams.

1.4. Stage 2: Spectroscopic processing TSO testing

Pipeline step	NIRISS status	NIRISS comments	NIRCam status	NIRCam comments	MIRI status	MIRI comments	NIRSpec status	N c
assign_wcs	DONE	Assign WCS indeed adds information to the ASDF name of the files. Tested that ranges of the WCS information made sense.	DONE	Runs without a crash and assigns WCS correctly.	DONE	done	DONE **Build 7.5:** Though we do not have a validation test, this is presumably ok since FS passes the test.	This dire perf BOT we ESA inte proc com aga bloc -157 wcs pas vali O
extract_2d	DONE	This step is not supported for NIRISS/SOSS, and indeed running this steps shows the "not-supported" message. Still, saves an output equal to the input.	DONE	Runs without a crash and extracts 2d spectra.	DONE	n/a	DONE **Build 7.5:** Though at present we do not have a validation test, this is presumably ok since FS passes the test.	bloc -157 2d s spe the test
flat_field	DONE	Flat field is correctly applied to NIRISS/SOSS data.	DONE	Runs without a crash and performs flat field correction.	DONE	done for LRS B7.5, 7.6, 7.7	DONE **Build 7.5:** Validated/verified	bloc -107 c: fla cals faili vali FS BOT MO
srctype	DONE	As stated in the documentation, TSO always defines SRCTYPE to POINT. The pipeline indeed does this for NIRISS/SOSS TSO data.	DONE	Runs without a crash. SRCTYPE correctly set to POINT.	TODO		DONE **Build 7.5:** Validated/verified Logic is now correct	bloc -157 type spe corr NIR and moc Cl

photom	DONE	Photometric calibration is performed according to the NIRISS reference files.	DONE	Runs without a crash and performs as expected.	DONE	done for LRS B7.5, 7.6, 7.7	DONE **Build 7.5:** N/A	tso_ is sl NIR Jul : for t unit cod and moc
extract_1d	DONE	This step currently extracts a big chunk of the whole subarray as a spectrum, which produced order overlap, which makes this step produce an unusable product for science right now. An algorithm is in development for this step. For details, see ⚠ JP-201 - Jira project doesn't exist or you don't have permission to view it. .	DONE	Runs without a crash and extracts spectra. Still need to enable column-by-column background subtraction with a Jira ticked issued ⚠ JP-1478 - Jira project doesn't exist or you don't have permission to view it.	DONE	done for LRS up to B7.7	TODO **Build 7.5:** methodology needs to be vetted by TSOWG	This curr cras BOT ditic com pen

Extended comments by instrument branch:

- NIRISS:
- NIRCam
- MIRI:
- NIRSpec: **Build 7.5:** general stage 2 (spec2) BOTS comment: verified BOTS (runs without issues and skips the correct steps)

1.5. Stage 2: Image Processing TSO testing

Pipeline step	NIRISS status	NIRISS comments	NIRCam status	NIRCam comments	MIRI status	MIRI comments	NIRSpec status	NIRSpec comments
assign_wcs	N/A		DONE	Runs without a crash and assigns WCS correctly.	TODO	have successfully run imaging TSO simulations through Image2 and TSO3 but no notebooks yet for individual steps. high level checks looked good.	N/A	
flat_field	N/A		DONE	Runs without a crash and performs flat field correction.	TODO		N/A	
photom	N/A		DONE	Runs without a crash and performs as expected.	TODO		N/A	

Extended comments by instrument branch:

- NIRISS:
- NIRCam
- MIRI:

- NIRSpec:

1.6. Stage 3: Time-Series Observations (TSO) Processing testing

Pipeline step	NIRISS status	NIRISS comments	NIRCam status	NIRCam comments	MIRI status	MIRI comments	NIRSpec status	N c
outlier_detection	DONE	Testing and validation done. Found some problems both with the documentation (⚠ JP-1647 - Jira project doesn't exist or you don't have permission to view it.) and with the effectiveness of the algorithm itself. On this latter point, the algorithm marks as outliers points that are predominantly on the edges of the SOSS profile (	DONE		TODO		TODO **Build 7.5:** Validation test needs to be written.	

		⚠ JP-1654 - Jira project doesn't exist or you don't have permission to view it.). The algorithm also doesn't use time-series information at all, and only uses the median image to catch outliers (⚠ JP-1655 - Jira project doesn't exist or you don't have permission to view it.). In my opinion, algorithm currently not optimal for NIRISS/SOSS TSOs at least.					

tso_photometry	N/A		DONE	Runs without a crash and performs as expected.	TODO	have successfully run TSO imaging simulations through Image2 and TSO3 but no notebooks yet for individual steps. high level checks looked good.	DONE N/A	tso_ is n perf the spe (NIF
extract_1d	DONE	This step currently extracts a big chunk of the whole subarray as a spectrum, which produced order overlap, which makes this step produce an unusable product for science right now. An algorithm is in development for this step. For details, see ⚠ JP-201 - Jira project doesn't exist or you don't have permission to view it. .	DONE	Runs without a crash and extracts spectra. Still need to enable column-by-column background subtraction with a Jira ticked issued ⚠ JP-1478 - Jira project doesn't exist or you don't have permission to view it.	TODO		TODO **Build 7.5:** Validation test needs to be written. Sept 2019: For now, use the same test as for spec2.	bloc -172 c ex testi O bloc -172 of N BOT inte O leve "x1c proc NIR BOT con sep exte eacl inte dete obta exp entl no c way disti betv exte term corr dete inte whic nec orde to b com data dete give inte

white_light	TODO		DONE	Runs without a crash. Jira ticket issued for defining min /max wavelength for white light curve: ⚠ JP-1469 - Jira project doesn't exist or you don't have permission to view it.	TODO		TODO **Build 7.5:** Sept 2019: Need to design tests; start with verification	

Extended comments by instrument branch:

- NIRISS:
- NIRCam
- MIRI:
- NIRSpec: