H4C RTP **Josh Dillon, 8/4/20** 4-pol Raw Diff Data 4-pol Raw Sum Data **A Priori Antenna Status Table** zen.{JD}.diff.uvh5 zen.{JD}.sum.uvh5 Chunked RTP Stage I Chunked **Initial RFI Detection** xrfi_run_data_only.py **Extract Autocorrelations:** extract_autos.py Extract autos and write to disk. **Initial XRFI Flags** zen.{JD}.stage_1_xrfi/*zen.{JD}.*.h5 **Raw 2-pol Autocorrelations Run Antenna Metrics** ant_metrics_run.py zen.{JD}.sum.autos.uvh5 Try to figure out dead/crossed **Daylong Flag Thresholding** antennas. Run twice, once xrfi_day_threshold_run.py with known good antennas **Day-Long RFI Flags** "All Ants" Antenna "Known Good" "Possibly Good" **Data Down-Selection:** zen.{JD}.total_stage_1_ **Antenna Metrics** ???.py **Antenna Metrics Metrics** threshold_flags.h5 Perform down-selection for imaging zen. zen. zen. {JD}.sum.known_goo pipeline and bispectrum/delay {JD}.sum.maybe_goo {JD}.sum.all_ants.ant_ pipelines. This is probably two d.ant_metrics.hdf5 d.ant_metrics.hdf5 metrics.hdf5 separate tasks. **Initial RFI** "All Ants" Inspect **Data Inspect** Notebook Notebook **Imaging Data for** Used to look for Used by the **Internet Transfer** times with major commissioning ???.sum.uvh5 RFI contamination ???.diff.uvh5 team. **Bispectrum/Delay** "Known Good" "Possibly Good" Redcal: **Spectrum Data for Data Inspect** redcal_run.py Data Inspect **Internet Transfer** Notebook This is run twice for each file, once with Notebook Used for looking for ???.sum.uvh5 the known good antennas and once Used for ongoing ???.diff.uvh5 antennas that died. with the possibly good ones. graduating new antennas to "Calibration OK". "Known Good" Antennas: "Possibly Good" Antennas: "Known Good" "Possibly Good" **FirstCal Calibration Solutions FirstCal Calibration Solutions Redcal Inspect Redcal Inspect Omnical Calibration Solutions Omnical Calibration Solutions** Notebook Notebook **Omnical Visibility Solutions Omnical Visibility Solutions** Used in combination Used for ongoing zen.{JD}.sum.first.calfits zen.{JD}.sum.first.calfits with the Data Inspect graduating new zen.{JD}.sum.omni.calfits zen.{JD}.sum.omni.calfits notebook to pick antennas to zen.{JD}.sum.omni_vis.uvh5 zen.{JD}.sum.omni_vis.uvh5 which antennas to "Calibration OK". calibrate and LST-bin **Nightly** Weekly **Observer** Commissioning Task Task **A Priori Antenna Status Table Conservative List of Bad** (Same as above) **Antennas (and Times???)** Update Weekly with new "Calibration OK" bad_ants/{JD}.txt Antennas. Also assign apparently broken Pushing this file triggers the antennas to digital or dish maintenance status. RTP Stage II RTP Stage II 4-pol Raw Sum Data zen.{JD}.sum.uvh5 (Same as above) Redcal: redcal_run.py **FirstCal Calibration Solutions** Run without any iterative removal of zen.{JD}.sum.first.calfits antennas. **Final Redcal Inspect** Notebook Checked to make **Omnical Calibration Solutions Omnical Visibility Solutions** sure data is OK zen.{JD}.sum.omni_vis.uvh5 zen.{JD}.sum.omni.calfits for LST binning Legend . Chunked Data with **External Origin Abscal: RIMEz Redundant Visibility** post_redcal_abscal_run.py Chunked **Simulation** Use RIMEz simulated visibilities to **Raw Visibility** ???.uvh5 calibrate redcal degeneracies. **Data Product** hera_cal process **Abscal Calibration Solutions** zen.{JD}.HH.abs.calfits **Calibration Data** Chunked **Product Abscal Inspect** Notebook See how abscal is **XRFI** hera_qm process doing and which LSTs xrfi_run.py and frequencies its Find and flag RFI based on raw data, failing at. Examined Omnical gains and chi^2. Omnical visibility before LST-binning solutions, and Abscal gains. Condenses **Metrics Data** flags to a single waterfall. **Product** casa_imaging process **Per-File Flagging Metadata** zen.{JD}.xrfi/*zen.{JD}.*.h5 **CASA Imaging** Intermediate data products from **Data Product** XRFI (flags and metrics) **Jupyter** Notebook (one per night) **XRFI Day-Long Thresholding** xrfi_day_threshold_run.py All other XRFI flags/metrics Flag entire integrations or entire from the same day. channels based on RFI statistics over a **Analogous Data or** whole day. Apply all flags to calibration. **Calibration from Other Times** Unfinished Module or Notebook **XRFI Thresholding Metadata Initial RFI Flags** zen.{int(JD)}.*_threshold_flags.h5 Flagged Absolute ??? Intermediate results from day-long **Calibration Solutions** (Same as above) XRFI thresholding zen.{JD}.HH.flagged_abs.calfits **XRFI Inspect** Notebook See how well XRFI worked. Examined before LST-binning. **Smoothcal:** smooth_cal_run.py All other absolute calibration Smooth calibration solutions on a solutions for the same day. desired calibration and frequency scale. Also selects a reference antenna. **Smoothed Absolute** smooth_cal **Calibration Solutions** Inspect Notebook zen.{JD}.HH.smooth_abs.calfits See how well smooth_cal worked. Examined before LST-binning. **TO FIGURE OUT: Reflection Fitting Noise Estimation Update Omnical Visibility Solutions** Clean Up / Librarian **Delay Filter Fringe Rate Filter Imaging Movies LST BINNING**