# Data Setup and Processing for JASON

See Data Processing Order of Operations document for SOP.

* Documentation
  + Jason\_Metadata.xls
    - Lowerings “dives” – enter dive specifics
    - Enter Dates and Times:
      * Start Launch = In water
      * Start Ops = on bottom
      * End Ops = off bottom
      * End/On Deck = on deck
        + Good resource for this is Sealog can also call the bridge for launch/on deck times.
    - Extract some data for spreadsheet from…
      * cd /Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest
      * head Dive#\_renav.ppi
      * tail Dive#\_renav.ppi
      * First record of head is first time on bottom last line in tail is leaving the bottom. Enter bounds far right from MatLab output.
      * Verify lat/lon of origin and gather UTM
        + /Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest
        + grep ^DVZ \*.DAT |cut –d’ ‘ –f 23,24 (lat/lon)
        + grep ^DVZ \*.DAT |cut –d’ ‘ –f 22 (utm)
        + Webb’s version of using awk vs cut has worked much better for me

grep ^DVZ \*.DAT | awk ‘{print $22, $23, $24}’

* + - * Get max depth of each dive
        + /Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/
        + ~/bin\_scotty/max\_jason\_depth.pl Dive#.DEP.raw
    - Highlights Tab
      * A1 – file > import > csv file > select the log\_dive#.csv file from Highlights folder
      * Delimited, comma, general
  + JasonDataPkgSummary.doc
  + NDSF\_receipt.doc
  + Highlights\_log.xls
    - Pull off the data logger station.
      * [datalogger@198.17.154.67](mailto:datalogger@198.17.154.67) pass:dsldata1
      * scp <filename> tina@198.17.154.184:/Volumes/Jason\_Data
  + Config Pics – Per dive number; get 360 views on Jason and close up’s of sensor additions and any unique setup.
  + Dive Reports – EL prepares. Will be asked for 4 time points, max depth, possibly weather and formatting.
    - Dive reports get PDF’d and emailed to rmunier@whoi, ssoule@whoi and DSL-Sea
* H264Recordings
  + H264rec3
    - ~/src/CamDisplayStable201710$ ./cam\_display.py
    - runs the Cam Display to start recording/growing
    - pulling data
      * ssh [installer@198.17.154.198](mailto:installer@198.17.154.198) pass: dsl!inst
      * cd src/CamDisplayStable201710
      * scp \*yyyymmdd\*.ts [tina@198.17.154.184:/Volumes/Jason\_Data/CruiseID/h264/Dive#](mailto:tina@198.17.154.184:/Volumes/Jason_Data/CruiseID/h264/Dive)
      * need to do this for .ts (Videos), .txt (Metadata), and .srt (Subtitles)
  + H264rec4
    - Rec 4 is acting as backup in case something goes down on rec 3. You do not need to download these files unless something is wrong with rec3. These files can be deleted post dive/cruise as space is needed.
    - ~/src/CamDisplay3$ ./cam\_display.py
    - runs the Cam Display to start recording/growing
    - pulling data
      * ssh [scotty@198.17.154.189](mailto:scotty@198.17.154.189) pass: dsl!scot
      * from h264rec4 computer (can access via KVM)
      * scp \*yyyymmdd\*.ts [tina@198.17.154.184:/Volumes/Jason\_Data/CruiseID/h264/Dive#](mailto:tina@198.17.154.184:/Volumes/Jason_Data/CruiseID/h264/Dive)
      * need to do this for .ts (Videos), .txt (Metadata), and .srt (Subtitles)
      * each 15min clip, 1-2 merges per minute, 15 minute per hour on bottom (can be lengthy process)
  + Data Processing for both h264’s
    - Organize data into folders
      * .srt = Subtitles mv \*.srt Subtitles
      * .txt = Metadata mv \*.txt Metadata
      * .ts = Videos mv \*ts Videos
      * will end up with .ts file +5subtitles per video
    - Creation of 5th subtitle that incorporates the .ppi renav
      * cd /Subtitles
      * vi (or nano) ~/bin\_scotty/make\_st5.pl
        + set proper dive number/path
      * ~/bin\_scotty/make\_st5.pl \*st1.srt
      * cd ..
      * ~/bin\_scotty/make\_J2\_mkv.sh
      * will see videos.mkv created in main h264/Dive#
  + Backing up H264 Recorders
    - Talk to Scotty and better understand /scratch1
    - Definitely want to clean out/back up main directory in between dives
    - For RR1902 created a dive folder in main directory to main contents to.
  + Troubleshooting
    - If recorders start recording but don’t switch to growing, try rebooting the machine.
    - Pink tint to videos is likely sign of bad HDMI splitter
    - Check that the videos being logged actually have data
      * ls -la
    - Use totem to view the video and check quality
      * Totem <filename>
* HDgrabs
  + Camera feed is selected through the Smart Hub Controller. Need to select video feed and hit “Make It So”. Won’t see camera selection update until you hit the grab button.
  + ssh [data@198.17.154.194](mailto:data@198.17.154.194)
    - user: data pass: dsldata1
    - cd /data
    - ls
    - shows you folders grouped by day which contain all the grabs of that day
    - cd capture\_(year month day time)
    - scp \*.GMTif [tina@198.17.154.184:/Volumes/Jason\_Data/Cruise](mailto:tina@198.17.154.184:/Volumes/Jason_Data/Cruise) Name/HDgrabs/Dive Number
    - pass: dsl!tina
  + Now convert extension from .GMTif to .tif
    - Using rename\_gmtif.sh in ~/bin\_scotty
    - Navigate to directory with files you want to rename
      * Cd /Volumes/Jason\_Data/Cruise\_Name/HDgrabs/Dive#
      * ~/bin\_scotty/rename\_gmtif.sh
  + Linking Renav to HDgrabs:
    - Same as linking renav for sulis photos
    - cd /Volume/Jason\_Data/CruiseID/HDgrabs
    - ~scotty/bin/photoinfov2.py –x img=Dive# out=divenumber\_HDgrabs.ppfx nav =/Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest/dive#\_renav.ppi
    - processing occurs and Dive#\_HDgrabs.ppfx is created
* Heat Flow Probe
  + Connection:
    - Prism port 3 (Baud of either 9600 or 38400)
      * Depending on probe
    - RS 232/3 – black serial cable (moxa 144)
      * Runs through Jason system
    - RS 232/3 – red Ethernet to serial
      * Runs through Rob (scientist) computer
  + Data
    - Incorporated in main Jason lowering data but if science wants it raw follow next steps.
    - ssh jason@198.17.154.201 dsl!jason
    - cd hg/dsHeatflow/build\_5ch/data or /build\_9ch/data
    - scp hfp\*.txt [tina@198.17.154.184:/Volumes/Jason\_Data](mailto:tina@198.17.154.184:/Volumes/Jason_Data) dsl!tina
* Highlights 4k and 1080i
  + 4k Captured by the KiPro AJA drives. 4k Sulis footage. Only video feed for 4k is the Sulis SciCam. Folder (highlights4k)
  + 1080i is captured by the KiPro Rack. Can toggle between different cameras (Brow, Pilot, Sci). Folder (highlights1080i)
  + Eject drive and plug into data station to download footage. Files can be large stay on top of this after each dive.
  + Monitor drive space try to not delete things until after the cruise has ended. Drives fill from left to right, make sure to have a backup in place.
  + Eject – hit slot.
  + Monitor drive space
  + Changing SC.mov videos to proper name: works for both highlight types.
    - Go to directory
      * cd /Volumes/Jason\_Data/CruiseID/Highlights/Dive Number
      * ls
      * /usr/local/bin/extract\_mov\_md.sh
        + creates the batch files with timestamp info for renaming
      * vi ~/bin\_scotty2/make\_rename\_hilites\_script.pl
      * opens the text editor or use nano (way easier!)
        + x = delete i = insert shift zz = save
        + use arrow keys to scroll to where you want
        + need to adjust the cruise id
      * cat batch\_list.csv | ~/bin\_scotty2/make\_rename\_hilites\_script.pl
      * more rename\_dive-number\_hilites.sh
      * ./rename\_dive-number\_hilites.sh
* Lowering Data – Populating the Vehicle Folder
  + How to pull sensor data to raw data folder and then process to proc data folder
    - cd ~/bin\_scotty2
    - ls
    - more get\_data\_byday\_osx2016 (to view)
    - ~/bin\_scotty2/get\_data\_byday\_osx2016 (space) /Volumes/Jason\_Data/Cruise-ID/Vehicle/ (space) Cruise-ID yyyymmdd
    - repeat for each day Jason is in the water
    - A Rawdata folder will be created that has a folder for each yyyymmdd as well as a Navest folder.
    - vi (or nano) parse\_daily\_data.sh
      * add date (day) yyyymmdd
      * modify sensor list
      * add cruise ID
      * edit destination path
    - vi commands
      * i = insert left
      * a = insert right
      * x = delete \*use esc to toggle between these three
      * arrow keys to navigate
      * shift zz = quit with saving
      * :q! = quit without saving
      * dd = delete a line
      * shift g = end of file
    - create procdata folder first in vehicle
    - ~/bin\_scotty2/parse\_daily\_data.sh
      * creation of CruiseID folder in procdata occurs
    - Navigate to Cruises folder
      * Make a new folder for current cruise
      * Add a mk\_low\_diveid for each lowering
        + ie: mk\_low\_1113
    - Make changes to mk\_low\_diveid
      * Use vi, nano, or text editor (Xcode)
      * need to add in water, on bottom, off bottom, on deck date/times.
    - ./mk\_low\_1113 (runs it)
    - create new one for next dive
      * cp mk\_low\_1113 mk\_low\_1114
      * vi mk\_low\_1114 – setup correctly
* Nav / SciNav Setup
  + Cruises > rr1901 (cruise name)
    - Maps and targets that science provides us. Should scp these from Nav to SciNav.
  + Doppler Reset – sets the starting point for the DVL. Integrate from this point forward to improve navigation.
  + Gathering Origin info for each dive
    - ssh [jason@198.17.154.225](mailto:Jason@198.17.154.225)
      * User: jason password: dsl!jason
    - /cruises/CruiseID/origins
  + Science using our USBL for tracking.
    - If this occurs will need to provide them with the .DAT files as well as telling them which VPR line is logging the data.
      * Example – Torres 1901 used USBL to track their heat flow measurements sans Jason.
    - ssh [jason@198.17.154.225](mailto:Jason@198.17.154.225)
      * User: jason password: dsl!jason
      * /data/nav
      * scp 2019\*.DAT tina@198.17.154.184:/Volumes/Jason\_Data/RR1901
* Renav
  + cd /Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest
  + ls
  + cp ../../../CruiseID/Dive#/navest/ jason1102\_renav.m jason1101\_renav.m
    - Copy a x\_renav.m script into the folder you want to renav
  + Scotty now has two different renav scripts
    - jasonDive#\_2dvlrenav.m
    - jasonDive#\_renav\_dev.m (usbl only)
  + now open MatLab and make necessary changes to the script
    - dive number
    - start launch, start ops, end ops, on deck (times/dates)
    - origin – located on nav comp
    - path
  + run the script jason1101\_renav.m
  + enter complementary cutoff
    - cutoff freq – proportion of usbl to dvl
      * smaller # increases dvl
      * bigger # increases usbl
    - Important for things like photomosaic’s. The usbl will destroy mosaics with the “cloud” while dvl gives you greater precision.
  + save nav data (n)
  + save raw (n)
  + Make sure you save the cords outstr = it goes in metadata spreadsheet. All the way to right of lowering’s tab.
* SeaLog
  + Grabbing SeaLog-framegrabs:
    - ssh [jason@198.17.154.223](mailto:jason@198.17.154.223) dsl!jason
      * cd sealog-images
      * scp [tina@198.17.154.184:/Volumes/Jason\_Data/Cruise](mailto:tina@198.17.154.184:/Volumes/Jason_Data/Cruise) ID/sealog-framegrabs/Dive#
    - 369byte images are garbage, placeholder by system.
  + Post Dive Data Management
    - ssh [jason@198.17.154.223](mailto:jason@198.17.154.223) dsl!jason
    - cd /mnt
      * mounts SciShare which is where you should put the sulis photos and renav .ppi files.
    - cd sealog-server-jason/misc
      * Sulis Cam Stills
        + python3 sealog-sulisCam-stills-import.py /mnt/SciShare/sulis\_pics/J2-1113 /mnt/SciShare/sulis\_pics/J2-1113\_sulis.ppfx
      * Sealog Renav
        + python3 sealog-apply-renav-to-events.py J2-1113 /mnt/SciShare/Renav/J2-1113\_renav.ppi
      * Sealog Postdive
        + ./sealog\_postdive.sh –c rr1902 J2-1113

enter y to continue and y to create directory

* + - * + backups go to /sealog-backup/cruiseID
        + Provide science with copies of each dives backup “J2-1140\_eventOnlyExport.json”
        + Provide science with their cruise event template. If changes made to templates happened per dive can also provide the event templates per dive. “cruiseID\_eventTemplates.json”
* SulisPhotos
  + Switch Sulis to Download mode. Use watch leader control box, misc screen, download mode on.
  + Camera mounts on data station computer. Pull off the photos from that dive leg.
  + If Jason is back on deck before download is done will need to put Jason back on DC deck cable.
  + Renaming of Sulis Photos:
    - Cd /Volumes/Hardrive Name/CruiseID/sulis\_pics/Dive#
    - ls
    - ~/bin\_scotty/rename\_sulis.sh
    - cd
  + Linking renav to the renamed Sulis photos:
    - Cd /Volumes/Hardrive Name/CruiseID/SulisPhotos
    - ~scotty/bin/photoinfov2.py –x img=J2-1101 out=J2-1101\_sulis.ppfx nav=/Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest/Dive#\_renav.ppi
    - processing occurs – only photos taken near the bottom will get nav info appended.
    - More Dive#\_sulis.ppfx shows what’s been added to each file.
* VirtualVan
  + User: vv-admin password: admin!!
  + Applying Renav to Virtual Van: do at end of each dive!
    - Need to create location on virtual van for renav.
      * ssh [root@198.17.154.221](mailto:root@198.17.154.221) pass: dsl!seadata1
      * cd /webdata/DAQ/CruiseID/Jason
      * mkdir Renav
      * chmod 777 Renav
    - Transfer renav.vvan.txt
      * cd /Volumes/Jason\_Data/CruiseID/Vehicle/Procdata/CruiseID/Dive#/navest
      * scp Dive#\_renav\_vvan.txt root@198.17.154.221:/webdata/DAQ/CruiseID/Jason/Renav
    - Adding renav to Virtual Van
      * <http://198.17.154.221/virtualvan/tools.html>
      * Re-Merge Jason VVan Navigation
      * User: vv-admin
      * Select Nav file from drop down and click Merge Navigation. Do for each dive.
    - Virtual Van Export (Make for each dive)
      * Go into Virtual Van
      * Export, pick dive csv, copy everything and put in a text file. Save as test.rtf
      * Cat ~/Desktop/test.rtf | grep –v ASNAP > ~/Desktop/test\_filter.txt
      * Rename the test\_filter.txt to dive#.txt
    - Make Jason HTML Tree (at end of cruise)
      * On VVan tools page
      * User – vv-admin
      * Click Make HTML Tree (sealog can’t do this)
    - Backing up of Virtual Van
      * Make Jason HTML Tree
        + DO AT END OF CRUISE

On vvan tools page click Make HTML tree

User: vv-admin

The big thing that SeaLog currently can’t do.

* + - * Backup for Data Package
        + Create folder on data drive (virtual\_van)
        + ssh [root@198.17.154.221](mailto:root@198.17.154.221) (dsl!seadata1)
        + cd /webdata/DAQ
        + scp –r rr1902 tina@198.17.154.184:/Volumes/Jason\_Data/RR1902/virtual\_van
      * On VirtualVan machine (access through KVM)
        + Login as Scotty (dsl!scot)
        + Open a Terminal and run rsync2seadata2.sh

Need to switch over to root

Su dsl!seadata1

As root cd /webdata/virtualvan/misc\_scripts

./rsync2seadata2.sh

This script will keep a running backup of virtual van (.221) to vvan 2 (.222)

* Data Stations – password for science: dsldata1
  + Event Logger – SeaLog and VirtualVan access
  + SciNav – passive view of NavG
  + Video Logger
    - Access the KVM and view 5 computers with recorders
      * H264rec3
      * H264rec4
      * Dlog
        + When vehicle comes online looking for green lights of various vehicle components

VEH

CSV

CTM - temp probe

SVP - sound velocity probe

MRU – part of winch (single body only)

OOS - optode

MAG - magnitometer

AFX

CT2 - CTD

HFP - heat flow probe

MTS

WWB - single body only

Logger comms

* + - * + To modify what is in dlog look at the jasonlog.ini file
        + jason@dlog1 > cd config
      * Frame Grabber
      * Video logger
        + Location of Scotty’s Annotator

Videologger computer

Can access from KVM

User: scotty pass: dsl!scot

Src > SDIannotMinimal

./annot\_control.py (runs it)

change color of font in annot.ini

white =0 blue =1 red =2 magenta =3 green =4 black =7

* Ship Setup
  + Talk to Ship about getting a phone ran into command van.
  + Winch Van Setup (when needed – single body)
    - 1 SDI cable ran from winch van
      * winch tension display to DIC 1-6
    - 3 cat 5 cables ran from winch van
      * winch van to jason net
      * winch van to ship net
      * winch tension KVM to DIC 2-8
  + Ship Cable Runs
    - Command Van to switch on Ship to activate ShipNet (cat5)
    - Command Van to gator box in lab to activate JasonNet (cat5)
    - Command Van to Galley (Not all ships? See what has been done in past.)
    - Gator box in lab to ship switch so SeaLog runs on both JasonNet and ShipNet (cat5)
    - Command Van to Ship (SDI)
    - KVM in Van to Lab for lab KVM access (1 CAT5)
    - Bridge to Command Van x2 cat5 1 SDI
      * 1 for Rasp Pi Bridge Display
      * 1 for NavG
      * 1 SDI for DP to Van will need some type of DAQ
  + Video Routers (40x40 and 20x20)
    - Annotated Science
      * 20x20 only
      * Destination – 9 Annotation
      * Source – 1 Sulis
      * Destination – 20 open output
      * Source – 11 output annotation
    - Sonardyne
      * 40x40
        + Destination – user 38
        + Source – Sonardyne 35
      * 20x20
        + Destination – 20
        + Source – 4 (direct feed of user 38)
  + Winch Van
    - Ewon VPN
    - MRU on dlog display
    - Creating Winch report and downloading CSV
      * Open pgAdmin III (blue elephant icon)
      * Look at left hand "Object Browser" pane
      * Double click "PostgreSQL 9.5..." to turn the red X icon into a happy server box icon, starting up the server
      * Expand "Databases (2)
      * Double click "dbroker" to turn that red X icon into a happy data cylinder icon, starting up that database
      * Once in the "dbroker" database, look up at the top ribbon of tools for the now colored "SQL" magnifying glass icon. Click it to open a command line window.
      * Paste the following into the "SQL Editor" box, except change the times and dates to those that you want data for:
        + SELECT tstamp, data FROM dbroker WHERE tstamp > '2019-08-26 12:23:00' AND tstamp < '2019-08-26 12:25'
      * Click the green arrow to the RHS of the magnifying glass to apply the query against the database.
        + Query is working appears bottom left corner
      * Once the query is completed, select the 'tstamp' and 'data/jsonb' sections in the "Data Output" tab
      * File->Export, export .csv file to D:\dbrokerQueryResults2018\ (or similar). Name file with metadata like cruiseID, timestamp, etc.
      * Add the results to the data pkg for WHOI (under "Engineering"). Apply "winch\_csv\_filter" to create file of simpler format.
  + Ship Net (IP’s Change per ship)
    - You need to identify what the Ship is assigning as an IP address. Can do this by SSHing into the computer and doing an ifconfig.
    - Ahi – SciShare
      * User: Guest Password: dsldata1
      * For Jason group: user: ahi-cv password: dsl!jason
      * Mac
        + Smb://137.110.149.65/SciShare
      * Windows
        + [\\137.110.149.65\SciShare](file://137.110.149.65/SciShare)
    - VirtualVan webpage with ship IP
      * 137.110.149.65/virtualvan/
        + \*you have to include the last /
    - SeaLog webpage with ship IP
      * 137.110.149.36