MOMA

- MOMA Alerts
- MOMA C++ Mining Scripts
- MOMA Consumables
- MOMA Data View
- MOMA MAIF Voltages
- MOMA Mailing List
- MOMA Mass Calibration
- MOMA Meta Markers
- MOMA Python Packages
- MOMA Python Time Synchronization
- MOMA Script Validation Process
- MOMA SEB Voltages
- MOMA System Locale
- MOMA Trending
- XINA Data Mining

MOMA Alerts

A MOMA Alert is defined as a notification of certain parameters of interest in a single MOMA Test. This page details the logic we use to extract the data for each alert. Note that this is the master reference, which means the engineers and scientists should ensure that the logic detailed here is correct. All alert data is in an Alert Panel within a typical TID record __TOC__

General Error Check

Severity : Notice

- Message Log is parsed for the phrase 'Error' while excluding messages that contained 'Holding at stop for operator' or 'error =' <- Means a value was being set.

MAIF APV Comm Check

Severity : Warning

- Message Log is parsed for the phrase 'ldi: error'.

Spacewire Comm Error

Severity : Warning

- Spacewire Disconnect (HK 522) > 0 or Spacewire Parity (HK 523) > 0 after 30 sec.

Severity : Notice

- Spacewire Disconnect (HK 522) > 0 or Spacewire Parity (HK 523) > 0 only before 30 sec.

MOMA Safe Event

Severity : Notice

- MOMA_STATUS_SAFE (HK 507) is 1 at any time during the TID.

WRP Too Fast

Severity : Notice

- PWM Speed (HK 832) > 103000RPM at any time during the TID.

WRP Failed to Start

Severity : Warning

- PWM Speed (HK 832) surpassed 1000RPM but never reached over 99000RPM at any time during the TID.

WRP Pressure Spike

Severity : Warning

- MOT_SUP_IMON (HK 830) > 1.1A at any time during the TID.

Dark Counts

Severity : Warning

- SCAN_TELEM_DAC_ID (HK 252) == 20000 and SCAN_TELEM_DAC_VAL (HK 253) > 5 at any time during the TID.

Laser Burst Energy Variation

Severity : Info

- Standard Deviation of the Energy from all Pulses in a Single Burst > 5.0

Laser Pulses Mismatch

Severity : Notice

- Number of Main Pulses Commanded (HK 1002) != # of Pulses in the Burst

Unknown SEB Sequences

Severity : Notice

- Scan Type is LDI or EI && SEB Packet Sequence Count == 0
- Scan Mode determined using c_scan.scan_mode_as_string()
- Sequence Length determined using len(c_scan.seb_sequence_pkt_indexes())

Filament VMON Pegged

Severity : Warning

- FIL_VMON (HK 19) > 4.9 at any time during the TID.

Pulse Count | Packet Size Mismatch

Severity : Warning

- HKID 1050 (PLS_SUBPKT_CNT) != (pkt.length() - 4 / 16)

Summed Scan | TIC Mismatch

Severity : Warning

- In a sum scan, the number of summed scans != the number of TICs

Gap in Scans

Severity : Warning

- Bin Delta between Science packets is > 100

cFactor | TIC Mismatch

Severity : Warning

- Number of cFactors != # of TICs

Scan Status | TIC Mismatch

Severity : Warning

- Number of Scan Status Packets != # of TICs

Dropped Packets

Severity : Warning

- Packet Sequence Numbers

Both EMs On

Severity : Warning

- Both Electron Multipliers are on at the same time

Limit Violations

Severity : Warning

- Come Back Later

MOMA C++ Mining Scripts

Introduction

The MOMA C++ mining scripts are command-line tools built on top of MOMA Data View code.

The primary purpose of these scripts is to extract data into a XINA Online-friendly format. They currently live in mine699.gsfc.nasa.gov in the /home/mpallone/cpp-playground directory (although I intend to move them into /usr/local/bin as soon as I can get permission...):

If running these executables on mine699, be sure to be logged in as a user whose momagse directory is configured and up to date, such as the 'moma' user.

Housekeeping mining script

Absolute filename: /home/mpallone/cpp-playground/extract_hk_data_main

From the file's header comment:

```
/**
 *
 * @file   extract_hk_data_mainclass.cpp
 * @author Mark Pallone
 * @date   December, 2015
 *
 * A script to extract HK data from a TMFile in order to populate XINA online
 * databases.
 *
 * Data will be written to the specified output directory with the preferred
 * XINA format. That is, one datapoint per line as follows:
 *
 *     utc time,fsw time,relative time,hkid,science value
 *
 * Note that times will be in microseconds, not seconds, unless the --seconds
 * flag is used.
 *
 * The name of the output file will be:
 *
 *     <tid_number>{=html}_hk_data.csv
```

```
*
* Example usage:
*
*     ./extract_hk_data_main  --infile <full path to tm.mom file>
*                             --outdir <full path to output directory>{=html}
*                             --hkids 1,2,5-10,15-20  --dhkids 30-50,55
*
* HKIDs listed immediately after the --hkids flag will have *all* of their
* values stored.
*
* Delta HKIDs listed next to the --dhkids flag will have their initial value
* stored, and subsequent values will only be stored if they differ from the
* previous value for that particular HKID.
*
* One or both of --hkids, --dhkids must be specified.
*/
```

MOMA Consumables

A MOMA consumable is defined as any piece of hardware in the MOMA instrument that has a limited lifetime. What exactly "lifetime" means is dependent on the specific piece of hardware we are interested in and a single consumable may have more than one way to define its lifetime. This page details the logic we use to extract the data for each consumable. Note that this is the master reference, which means the engineers and scientists should ensure that the logic detailed here is correct. All consumable data is available on XINA Online at https://ssed.gsfc.nasa.gov/xina/xo/tool/moma-timeline.

WRP Consumable

On Time

WRP on time is determined by calculating the total time that PWM_SPEED (HKID 832) is above 500 RPM.

Cycles

The number of WRP cycles is determined by counting the number of times the PWM speed (HKID 832) goes about 30,000 RPM.

NOTE: This current logic counts some of the failed WRP Startups as a cycle. See trend for failed startups.

Filament Consumable

On Time

Filament 1 and Filament 2 on time are determined by calculating the amount of time that SEB IS_EMON_A and IS_EMON_B (HKIDs 17 and 18) are greater than 1 uA.

Cycles

Filament 1 and Filament 2 cycles are determined by counting the number of times the filaments transition from off to on. A few notes about the algorithm:

- if the digital status packet indicates that the SEB is off, then the state of the filaments is considered to be off.
- a filament is considered to be on if if its on "on bit" (HKIDs 102, 105) is set to 1, and if the FIL_VMON is above 2.5V. Otherwise, it is assumed to be off.

- if the FIL_VMON is not being sampled, then only the "on bits" (HKIDs 102, 105) are used to determine if the filaments are on.

EM Consumable

On Time

EM 1 and EM 2 on times are determined by summing the on times for all scans, and then looking for any time when the EMs were constantly on outside of scanning.

SEB sequences are examined to determine the on time for individual scans. In order to avoid double-counting scans, only sum packet (packet type 27) scans are examined when in EI mode, and only science packet (packet type 26) scans are examined when in LDI mode.

Additionally, outside of science scans, the total time that the EMs are on are summed. This is determined by summing the amount of time that EM A and EM B are below -1900 V. HKIDs 22 and 23 are examined in this phase of the algorithm.

Bradley Tse pointed out a flaw in this algorithm:

> There is a corner case where that may introduce its own problems. If the user stops the script after the Sequence was dumped but before scanning started (maybe because they noticed one of the parameters was incorrect), then that means the SebSequences will be duplicated.
>
> And so EM on time for all scans after that (until the next Sequence Dump) will have 2x the amount (or however many times the script was stopped before scans occurred)

Full Cycles and Soft Cycles

EM Cycles are determined through two passes of the data.

The first pass of EM Cycles are counted from the scan status packets. The cycles are calculated differently depending on the type of mode the test is in.

If in LDI mode [1], if HKID 222 (SCN_EM1) or HKID 223 (SCN_EM2) is below the -1900V threshold, then a full cycle is added to the corresponding detector's count. If in Dark Counts mode [2], no full cycles are added here, as typically a Dark Counts run contains only 1 full cycle total. If in EI mode [3], soft counts are added to EM1 if HKID17 (IS_EMON_A) is greater than 9, and the same procedure for EM2 and HKID18 (EMON_B).

The timestamps of the scan packets are used to determine gaps in time when scans are not occurring during the test. A second pass through but looking specifically in the gaps and the binned HK data is necessary to avoid missing any full or soft cycles.

Here, HKID47 (SEB: EM_1_DAC0_SET) and HKID48 (SEB: EM_2_DAC1_SET) are looked through individually. If the HK value is > -50V, the detector is OFF. If the HK value is < -1900 the detector is ON. Full cycles are added for each occurrence where a detector changes from the OFF state to the ON state within a gap.

[1] - LDI Mode is when Filaments are OFF ([HKIDs 17/18] < 9), and the RF (HKID 235) is ON (End Amplitude >= 500A)

[2] - Dark Counts Mode is when Filaments are OFF ([HKIDs 17/18] < 9), and the RF (HKID 235) is OFF (End Amplitude < 500A)

[3] - EI Mode is when a Filament is ON ([HKIDs 17 or 18] > 9)

EM Counts

EM 1 and EM 2 counts are calculated according to the logic below. If the most recent DAC value (HKIDs 222 and 223) is zero, then ions are not counted against the multiplier.

```
    # dummy variable to help when counting ions in science packet (packet 26) scans
    temp_counts = 0
    #
    for each packet in the tm.mom file:
        if the packet indicates that filament A or filament B is on (HKIDs 105 and 102):
            set the current scan type to EI
        else:
            set the current scan type to LDI
        #
        else if the packet type is a scan status packet (type 25):
          if we're in LDI mode:
              count temp_counts against whichever EM is on (or both, if they're both on)
              set temp_counts back to 0
        #
        else if the current mode is LDI mode and the current packet type is a science packet (type 26):
            temp_counts += packet.total_ion_count()
        #
        else if the current mode is EI mode and the current packet type is a sum packet (type 27):
            count packet.total_ion_count() against whichever EM is on (or both, if they're both on)
        #
        # special-case code for handling dark count scans:
        else if the current mode is LDI, and the current packet is a sum packet (type 27), and the packet's RF DAC v
alue (HKID 405) < 50:
            count packet.total_ion_count() against whichver EM is on (or both, if they're both on)
```

Laser Consumable

Laser Shots

Laser shots are determined by summing the number of main pulses (HKID 1050) and the number of pre pulses (HKID 1001).

Valve Consumable

MS Valve Cycles

Valve cycles are determined by examining the message log and counting the number of times a valve transitions from closed to opened. When counting valve cycles for a TID, the state of the valve cycles in the previous TID is accounted for. The exact regexes used (as of Feb 22) are:

```
VALVE_OPENED_REGEX = r'.*valve \d opened$'
VALVE_CLOSED_REGEX = r'.*valve \d closed$'
```

PCV Cycles

Counts the number of times the Pressure Control Valve (PCV) on the GC has been opened. This is found by adding all the times the PCV On bit in the GC Control register (mask 0x4000 of register 0x3A)(HKID 712) transitions from 0 to 1.

PCV On Time

This is the total powered time of the PCV. This is determined by calculating the duration of time when the PCV On bit (HKID 712) is set to 1 in a TID.

GC Flow Time is calculated the same way, but was made as a separate consumable so it could be reset when the Helium tank is refilled.

Bus Energy Consumable

Bus 1 energy is calculated by summing HKID 939, and then dividing that sum by 3600 to get watt-hours.

Bus 2 energy and Bus 3 energy are calculated in the same way, except HKIDs 940 and 941 are used.

Bus 4 energy is always assumed to be zero.

Miscellaneous

Duration

This is the sum of the durations of all TIDs. Currently, each TID's duration is calculated by simply subtracting earliest_packet.timestamp from last_packet.timestamp. However, we plan on making this calculation more accurate by ignoring rover simulation packets.

Aperture Valve

This is the number of times the "apv open" message appears in the message log. NOTE: This does not account for APV opens during commissioning since the "apv open" message is out output.

MOMA Data View

MOMA Data View is an application built with the Qt framework that is used to interpret, display, and analyze MOMA data. It can be used for both real time and post processing purposes. At this time we only support Macs.
__TOC__

Installation

Prerequisites

- A NASA AUID (agency user id)
- Access to the momaioc server. This will require your AUID account to be setup with the momaioc server. Please contact Bradley Tse if you do not have access.
- A functioning RSA token that is attached to your AUID which will be used when creating a tunnel to the momaioc server. The tunnel is used to gain access to the supplementary files and data.
- Either the `svn` command line tool or a working Subversion GUI client. See the instructions for installing Subversion. `svn` is used to download MOMA Data View, all required supplementary files, and MOMA data.
- You need to either be logged into the NASA network (not NASA guest) or VPNed in

Tunneling into momaioc

This step creates a "tunnel" to the momaioc server, which basically gives you access to all of the files on the server. The momaioc server contains our Subversion repository which is where we store all of the data, apps, and files.

1. Open up a terminal and paste this command into the terminal, replacing AUID with your actual AUID:

```
ssh AUID@momaioc.gsfc.nasa.gov -L 6994:localhost:3690
```

If that command complains about an unknown hostname, then use this command instead:

```
ssh AUID@128.183.20.207 -L 6994:localhost:3690
```

2. You should now be prompted for your password. Your password will have the following format (omitting the carrots when you actually enter it): `<RSA_PIN>` `<RSA Generated Numbers>` Be careful about entering your password because after 3 consecutive unsuccessful attempts you will be locked out.

3. You have now successfully created a tunnel to the momaioc server, which means we can now access the data and files! You do not need to do anything else with this terminal tab. It should be left open for the duration of your work session. All other terminal work should be done in a different tab.

Downloading momagse

The momagse directory contains various configuration files and all of our apps, including MOMA Data View.

1. Open a new terminal tab

2. Change to your home directory:

```
cd ~
```

3. Download/checkout the momagse directory:

```
svn co svn://localhost:6994/momagse)
```

4. You should now have the momagse directory!

Installing MOMA Data View

1. Mount the MOMA Data View dmg

```
open ~/momagse/Apps/momadataview.dmg
```

2. A Finder window should have opened with the MOMA Data View application in it. Drag and drop the application to your Applications directory.

3. Unmount the MOMA Data View dmg

4. You now have MOMA Data View installed!

Updating to a newer version

Note: The following instructions assume you have met all of the prerequisites.

1. Create your momaioc tunnel

2. Open a new terminal tab or window.

3. Update your momagse directory:

`svn up ~/momagse`

4. Follow the same steps in Installing MOMA Data View except this time replace the pre-existing app.

Getting the data

1. Create your momaioc tunnel

2. **Open a new tab or terminal window.** If it does not already exist, create a momadata directory in your home folder and change into it.

```
mkdir ~/momadata
cd ~/momadata
```

3. If you do not have any data, then use the following commands to check out the data:

```
svn co svn://localhost:6994/momadata/etu etu
svn co svn://localhost:6994/momadata/fm fm
svn co svn://localhost:6994/momadata/qsm qsm
```

If you already have data, then you can update your working copy as follows:

```
svn up ~/momadata/etu ~/momadata/fm ~/momadata/qsm
```

Build Environment

The following configuration is used to build MOMA Data View:

- Qt 4.8.7
- Apple LLVM version 7.0.2 (clang-700.1.81) 64bit

Version History

The changelog is ordered from most recent version to oldest. Because a changelog has not been consistently maintained for the lifetime of the application, versioning only starts at 3.0.0.

5.2.0

Revision: 9728

Release Date: 2021-04-02

5.1.0

Revision: 9588

Release Date: 2020-12-18

5.0.2

Revision: 9531

Release Date: 2020-11-23

5.0.1

Revision: 9300

Release Date: 2020-10-08

5.0.0

Revision: 8994

Release Date: 2020-07-29

4.25.0

Revision: 5528

Release Date: 2017-06-28

4.24.4

Revision: 5425

Release Date: 2017-06-02

4.24.3

Revision: 5351

Release Date: 2017-05-22

4.24.2

Revision: 5328

Release Date: 2017-05-12

4.24.1

Revision: 5249

Release Date: 2017-05-1

4.24.0

Revision: 5113

Release Date: 2017-04-19

- General
  - Supported packet types are now defined in an external config file: TMDef/tm_packet_types.txt. As a result, this version of MDV can now support the new LITMS packet types.
  - The 699config.ini dialog is now scrollable to avoid issues with it being too big and the edges being off the screen
- Limit Monitor
  - Added buttons to "Clear All" triggered limits, and "Enable/Disable All Limits"
  - Disabling all limits now automatically clears any triggered limits
- HK Viewers
  - Decreased padding around the edges
- Raw Packet Viewer
  - Horizontal scroll bar position is now restored when the selected packet changes

---

4.23.6

Revision: 5013

Release Date: 2017-04-10

- Fixed bug where if the min mass in the SIC range was less than the scan's min mass, the calculated ion count was incorrect. This impacted the DAC Scan and Chromatogram Viewer.
- Increased the max allowed mass to 2000 in the Chromatogram Viewer's SIC range
- Changed order of the splash page fields
- Added a packet count to the splash page
- Chart closest point tooltip calculations are now performed in a separate thread. This should help improve performance for high rate samples.

4.23.5

Revision: 4985

Release Date: 2017-03-28

- Mass Scan Viewer
  - Fixed values of DACs 8-12 in the Scan Info area

4.23.4

Revision: 4976

Release Date: 2017-03-24

- Sequence Viewer
  - Now correctly uses the SET_BIN_TIME command to calculate ramp durations

---

4.23.3

Revision: 4939

Release Date: 2017-03-15

- Data Rate Viewer
  - Fixed the TVAC GSE packet being included in the data rate calculation
  - Added a checkbox to toggle the inclusion of GSE packets. Unchecked by default.
  - Added a new data size series (base 10)

4.23.2

Revision: 4870

Release Date: 2017-03-02

- Performance improvements. TIDs now load ~2-4x faster.

4.23.1

Revision: 4776

Release Date: 2017-02-14

- Science Settings
  - Added new Summed Scan Timestamp Mode: Average first and last
  - Added new Scan Info Display Format Mode: Succinct

4.23.0

Revision: 4762

Release Date: 2017-02-10

- General
  - Added support for the calibrated laser pulse energy conversion. HKID 1087 will use it.
  - Added support for 4 new laser pulse energy HK items: AVG (1089), MIN (1090), MAX (1091), and STD_DEV (1092)
  - Added model and DAC table to the splash page
  - Other minor tweaks, bug fixes, and optimizations
- All Charts
  - Added ability to change a series' axes via its legend item's right-click menu. This is not currently saved between sessions.
  - Added ability to choose your own series colors from the Setup menu. This will impact all charts. Reload of viewers is required to reflect changes.
  - Tweaks to the axis settings dialog
- HK Charts
  - Added option to display the journal.txt with the message log. This feature is currently not available in the Message Log Viewer.
- Science Settings
  - Added a new "Average Summed Scan Timestamps" option, which uses the Scan Status packet timestamps
  - Improvements to the setting's status indicator that is shown on each science viewer
- Mass Scan Viewer
  - Added laser energy stats to the scan info display
  - Split up the info into two panes
  - Added a "Animate Scan Changes" chart right-click option, which is off by default.
- DAC Scan Viewer
  - Added support for science packets
- Raw Packet Viewer
  - Added ability to export the packet list's columns
  - Added a local datetime display at the top
  - The timestamp column's tooltip is now the raw timestamp converted to seconds
- TID Explorer
  - The selected TID now stays in sync with the open TID

4.22.7

Revision: 4676

Release Date: 2017-01-26

- All
  - The Settings button has been removed from the science viewers. It can now be accessed from the Setup menu
  - Plots that are opened through an HK Viewer's right-click menu are now added to the Data Viewer menu's open viewers list
  - Closing an HK Viewer no longer closes the plots that were opened through it
  - Fixed negative timestamps when raw FSW Ticks > ((2^32) / 2)
- Energy Viewer

    - New Energy Viewer to calculate and display the energy used by the instrument
    - Data Viewers-->Open-->rsim_energy.grf
    - OR Data Viewers-->Viewer Finder-->search "energy"
- Charts
    - Added in support for exporting the series data, accessible through the chart's right-click menu
- Mass Scan Viewer
    - If the "Average" setting is enabled, the selected scans will also be averaged
- Chromatogram Viewer
    - Fixed bug where the scan indicator line would be hidden if the TIC series was hidden
- DAC Scan Viewer
    - Fixed a crash when using the Opt+Left-Click feature for certain data sets

---

4.22.6

Revision: 4625

Release Date: 2017-01-14

- Fixed crash caused by the chart's line items (messages, markers, limits, etc)

4.22.5

Revision: 4623

Release Date: 2017-01-13

- Added new Duty Cycle Viewer
- Small tweaks to Mass Scan Viewer's auto-scale logic

4.22.4

Revision: 4596

Release Date: 2017-01-06

- Added support for the TVAC Packet

4.22.3

Revision: 4563

Release Date: 2016-12-02

- Bug fixes
    - Fixed crash when opening DAC Scan Viewer while summed scans are being filtered out

- Mass Scan Viewer
  - The cFactor for each scan is now applied before summing multiple scans
- Data Rate Viewer
  - Added a new Data Rate Viewer to plot the TID's data rate
  - Data Viewers->Open->data_rate.grf
  - OR Data Viewers->Viewer Finder->search "data"
- TID Explorer
  - Searching now uses the same fuzzy filtering algorithm that everything else uses
  - Cmd+E now always shows the TID Explorer and highlights the search text. Use Cmd+W to close it.
  - Now supports Down/Tab and Up/Back-tab keys to navigate the tree
  - Added more right-click options
- General
  - Minor tweak to HK graph's auto-scale logic

4.22.2

Revision: 4482

Release Date: 2016-11-04

4.22.1

Revision: 4475

Release Date: 2016-11-03

4.22.0

Revision: 4426

Release Date: 2016-10-27

- New Annotation feature
- DAC Scan Viewer
  - Fixed bug that was causing the series to be labeled in reverse order
- Message Log Viewer
  - Fixed the font not being properly saved/restored
  - Fixed bug that was adding a blank line for messages that were filtered out

---

4.21.1/2

Revision: 4391

Release Date: 2016-10-18

- Science Data
  - Fixed the incorrect parsing of summed packets with an odd number of scans. It was causing the bins to be shifted by one to the right. This affected data that was generated with FSW v3.4.6 and above.
  - Fixed some rounding inconsistencies
- Mass Scan Viewer
  - Sum region is now saved/restored
- Chart
  - Closest X tooltip is now displayed even when crosshair is visible
- General
  - Fixed problems with the SWTS packet timestamps

4.21.0

Revision: 4355

Release Date: 2016-10-12

- Charts
  - Mouse wheel now zooms in and out -- doing it over a specific axis will modify only that axis
  - Shift key now enables "Panning" mode -- doing it over a specific axis will pan only that axis
  - Added a new Zoom Mode: X Only, Scale Y -- This zooms in an x-window, but auto scales the y-axis to fit the data.
  - Zoom mode is now displayed in the bottom right corner
  - Zoom stack size is displayed in the top right corner. Clicking on it will reset the zoom.
  - Added ability to cycle the chart's zoom mode with Cmd+Ctrl+Z shortcut
  - Added shortcut text in the chart menu for the ToolTip Mode
  - Changes to an axis's settings are now immediately shown
  - Fixed some edge case bugs with the cursor shape
  - Zoom stack popping is now disabled while the mouse crosshair is visible
  - Minor fixes to the Quick Graph's search algorithm
  - Fixed a minor memory leak with the menu
  - Hovering over the legend items now changes the cursor to a pointing hand
- General
  - Timestamp resolution is now correct for TIDs with resets (Big thanks to Mark!)
  - Added in the option to average the counts of the Summed scans - accessible from either the Mass Scan Viewer's or Chromatogram Viewer's "Settings" dialog
  - All "Latest" checkboxes are now auto-unchecked when the window is scrolled up
  - Tweaks to viewer default sizes
  - Added ability to plot a HK Items conversion to the Database Info window
- Mass Scan Viewer
  - Added a "Sum" mode that allows you to sum the counts in a range
  - You can now remove individual "Kept" series
  - Auto-grow/auto-shrink now only accounts for the data in the current x-axis range
  - Closest point ToolTip now correctly updates when the scan changes
  - Minor visual tweak to the filter popup
- Chromatogram Viewer
  - Added cTIC and cFactor AGC series
  - AGC series can now be individually toggled
  - Updated the how the "Integrate" mode works
  - Added new min and max inputs for the SIC functionality
  - Fixed bug where it would be difficult to manually input values into the "Integrate" widgets
  - Fixed bug where sometimes it would not load properly the first time
  - Now remembers its zoom mode and tooltip mode
- DAC Scan Viewer

    - You can now use the chart crosshair (option+left-click) to select the closest scan in the Mass Scan Viewer!
    - Added in an auto-scale option -- mainly relevant to auto-scale when the SIC range changes
    - Fixed bug where the current scan's tooltip would be in the wrong place if a SIC range is applied
    - Fixed data point visiblity being off
    - SIC range better validates its input
    - SIC range is now saved between sessions
    - SIC range is colored green when it is valid and being used
- HK Viewers
    - Added in right-click options to change all HK items to a specific value mode
- HK Graphs
    - Added a new "Real-time" tooltip option that displays the most recent values for each series
    - When adding new series, the new series is scaled to the current zoom window
    - Fixed bug where sometimes the auto-scaling would not work when opening an active TID until you manually reset the zoom
    - Fixed bug where the selected marker's ID was not being displayed on the chart indicator
    - Fixed bug where messages and markers were already selected the first time the panel is displayed
    - Decreased the font size of the message and markers
    - Replaced the row column from the messages and markers with a label at the bottom
- Raw Viewer
    - Added in HKID and Science value columns
    - All HK items are now displayed, but some are still incorrect
    - You can now right-click on an HK item to access its database info
    - You can now click on the address area in the hex display to toggle between Hex and Decimal
    - Fixed bug where sometimes selecting the first packet would not update the display
    - Fixed bug where sometimes the display would not be cleared when no packet is selected
- Message Log Viewer
    - Fixed extremely long load times
    - Redesigned the search feature

4.20.0

Revision: 4023

Release Date: 2016-08-16

- All Charts
    - Double clicking on a y-axis now scales to only the data contained in the current x-axis range - no longer pops a range off the axis' zoom stack
    - Double clicking on a x-axis now sets its range to fit all data - no longer pops a range off the axis' zoom stack
    - Replaced the mouse coordinates in the bottom left corner with an indicator directly on the axis - this has the added benefit of showing the coordinates for all axes instead of only one
    - A "closest point" tooltip is now on by default
    - Added a new "Closest X" tooltip mode which displays a combined tooltip for all series
    - Zooming can now be initiated outside of the axis rectangle
    - Delta x and y are now displayed along the zoom rectangle
    - Slight tweaks to the Cyan and Orange series colors
    - Changed the "All series data points visible" options to hopefully make it more intuitive
- Mass Scan Viewer
    - Default zoom mode is now x-axis only -- zoom mode is saved between sessions
    - Added in a right-click menu option to control the chart's anti-aliasing
    - The zoom level is no longer auto reset when the mass cal is changed or the cFactor option is applied

- Chromatogram Viewer
  - Added an "Interpolate" option to the integration functionality that will interpolate between points to calculate the area of the full window
  - Added an "AGC" option that will show series for the AGC parameters - the series are also available to export
  - You can now export the visible HK series - this required a separate "Time" column for each exported series since their times differ
  - Changed it's real-time auto scaling logic to match the HK Graphs
  - When series are checked/unchecked, their axis is automatically scaled now
  - Fixed bug where adding a HK series for the first time would cause the graph's zoom level to be reset
  - Minor changes to hover tooltips
  - Integration functionality's area is now rounded to the nearest whole number
  - Added ability to copy the area's value with Cmd+C; the selection must have focus since the chart also uses Cmd+C
- Message log Viewer
  - Remembers its font between sessions
- HK Graphs
  - Using the mouse crosshair will automatically open the msg/marker panel
- General
  - The list of open viewers in the "Data Viewers" menu is now sorted alphabetically to make it easier to find a viewer
  - Fixed the J2k epoch - this shifts the UTC time of data by 64 seconds, relative time is unchanged
  - Minor tweaks to the "HK Selection" dialog

4.19.1

Revision: 3859

Release Date: 2016-07-15

- Chromatogram Viewer
  - Fixed the area calculation
  - Improved the functionality of the integration feature

4.19.0

Revision: 3845

Release Date: 2016-07-14

- Chromatogram Viewer
  - Added new integration functionality. You can toggle it by using the "Integrate" checkbox. The chart crosshair (option+click or middle-click) can be used to move the integration region. The "Move Mode" determines which part of the region is moved.
- Mass Scan Viewer
  - Added support for 2nd order polynomial curve fits
  - Improved the auto shrink/grow functionality
  - The Mass Cal Dialog's curve fit now has a range that makes sense for the control mode
  - Fixed a bug where the MIC series was getting reloaded as a SIC series when the Mass Cal changed

4.18.6

Revision: 3822

Release Date: 2016-07-08

- Fixed the checked markers not being displayed

4.18.5

Revision: 3818

Release Date: 2016-07-07

- Fixed TICs not being extracted from the new sum packet format
- Fixed bugs caused by Tom's 4-byte padding
- Added sum of TICs

4.18.4

Revision: 3812

Release Date: 2016-07-05

- Added support for the new sum packet format that contains the cTIC factor and MBC/SIC for each scan (added in FSW 3.4.6)
- Added display of the new sum packet fields for each scan to the Mass Scan Viewer
- cTIC factor calculation now properly accounts for the TIC weight of each scan
- The text of the currently selected message or marker is now auto displayed on the chart
- Fixed bug where Laser Pulse Packets with no pulses were being dropped
- HK Viewers now save/restore their font size
- Minor changes to tooltips

4.18.3

Revision: 3773

Release Date: 2016-06-22

- Reverted accidental change to RF Mon and RF Dac conversions - this slightly shifted RF Mon and RF Dac conversions by ~0.001V.

4.18.2

Revision: 3765

Release Date: 2016-06-20

- Added ability to export visible series in Mass Scan Viewer - useful to export kept and averaged series
- Added plot of mass cal curve fit in mass cal dialog
- Fixes/improvements to real-time auto-scaling logic

4.18.1

Revision: 3726

Release Date: 2016-06-06

- General
  - New app icon to easily differentiate between MOMA applications
  - Fixed bug where MDV would pop up if hidden
  - All GSE created packets are never used for t0 now - impacts old ETU TIDs < 7300
- Charts
  - All charts now support animations. Can be disabled/enabled from Setup->Chart Animations. Requires a restart of the application.
  - Double-click on an axis to rescale only that axis
  - Axes now display the attached series' legend
  - Changed the default series colors
  - Added ability to select zoom mode in the right-click menu
  - Slight tweak to the default title padding
- HK Graphs
  - Added in Central Moving Average option when selecting HK items
  - Axes are no longer rescaled when modifying/adding HK items
  - Improved auto-scale logic. Now only auto-scales the currently visible series
  - Existing HK series' visibility and data point visibility are not reset when modifying/adding HK items
  - Message and Markers now have basic filtering functionality
- Mass Scan Viewer
  - Ported to the new chart library, which adds a legend, and copy/save functionality
  - Added averaging option
  - Fixed bug where using the "View Sequence" right-click option would not display the sequence with the correct scan data if the Sequence Viewer had never been opened
  - No longer resets zoom when changing graph modes
- DAC Scan Viewer
  - Now has chart copy/save functionality
- Raw Packet Viewer
  - Added in relative time column
  - Attempts to reselect the previous HK item when changing the selected packet

---

4.17.3

Revision: 3540

Release Date: 2016-04-27

- Chromatogram Viewer
  - Fixed the crashing when enabling the cTIC and MIC after adding a HK item
  - Fixed the incorrect calculation of the cTIC. Now uses the Src A Focus A register readback value instead of the last set DAC value
- DAC Scan Viewer
  - Ported to the new chart library
  - Added in an "auto-select" option that will automatically display the DAC Scan of the selected scan in the Mass Scan Viewer - NOTE: This does not work in real-time yet
- HK Selection Dialog
  - Fixed the Tab and Backtab functionality to scroll through the filtered HK items
  - Changed the shortcut to graph from Shift+Return to Cmd+Return (Shift+Return still works for now)
- TID Explorer
  - Fixed the expanded and selection state not being properly restored between sessions
- DAC Scan
  - Fixed the averaged DAC Scan values being truncated to an int
- Other
  - Quick Graphs are now centered when created

4.17.2

Revision: 3442

Release Date: 2016-04-7

- Fixed TID Explorer crashing when opened
- You can now double-click on files in the TID Explorer to open them in their default application
- When saving or copying a chart's image, the TID is now displayed in the bottom right corner instead of the top left to avoid overlapping the legend or title

---

4.17.1

Revision: 3434

Release Date: 2016-04-7

- New charts now have the right-click option to copy directly to the clipboard
- Added option to TID Explorer to use instead of the Open File dialog

---

4.17.0

Revision: 3403

Release Date: 2016-04-4

- Sequence Viewer:
  - Fixed the Sequence Timeline not properly updating when the selected scan changes
  - Fixed the hex display not automatically scrolling to the selected cmd

    - When you select multiple cmds, only each cmd's region will be highlighted. Before it would highlight everything from the start of the first cmd to the last one.
    - You can now zoom with the mouse - double-click to reset zoom
    - DACs that are set outside of the sequence now use the closest scan status packet's values
    - Different cmds have different colors
    - Added in support for SEB_RAMP
    - Option+Click or middle button click on the Sequence Timeline to automatically go to the closest scan
    - Fixed the RAMP_IMM so that it now uses the scan status packet's bin time. Before it was hard coded to 10us
    - Visible DACs can now be toggled via the right-click menu
    - Added in option to auto-select the selected scan's sequence
    - Other minor visual changes
- Chromatogram Viewer:
    - Beta version is now the default version
    - Now properly saves and restores any HK series that were added
- Graph's Message and Markers:
    - Added in a "Latest" checkbox that sticks to the end
    - Added in useful right-click menu options
    - Added in ability to filter out 0-id markers
- Graphs:
    - Added in a menu option to save the graphs
    - Graph window titles and menu names are properly updated when HK series are added or removed
    - When saving the windows, the added HK series are now properly saved
    - Added in the ability to plot moving averages of the data
    - Redesigned the HK selection dialog
    - Fixed the issue with the series disappearing when zooming in on unordered data such as resets
- TID Explorer Beta:
    - Accessed through the File menu
    - Provides an easier way of quickly navigating through TIDs

---

4.16.0

Revision: 3295

Release Date: 2016-03-22

- Sequence Viewer:
    - Sequences now use the selected scan in the Mass Scan Viewer to extract register values for the SET_DAC and PAUSE SEB commands.
    - Still unsupported:
        - RAMP command that uses the register values
        - No SET_*_REG commands within the sequence are being used. Right now it only uses the Scan's scan status packet values.
    - A green line is now used to indicate the beginning of a segment (i.e when pause or ramp commands are selected)
    - General visual improvements to the sequence timeline
    - Fixed the rollover issue when the sequence's timestamp is greater than t0
    - Uses the new hex display
- New and improved raw packet viewer
- Minor changes to the "About" dialog and menu ordering

4.15.3

Revision: 3265

Release Date: 2016-03-19

- Added in support for FSW v3.4.2 which changes how sum packets are timestamped. They are now timestamped when the last scan's science data ack is received, as opposed to being timestamped when the first science packet is received. The significant difference is that the sum packet's timestamp is now greater than the last corresponding scan status packet's timestamp. Before the sum packet's timestamp was less than the first corresponding scan status packet's timestamp.
- Chart axis grids are now gray for better visibility

4.15.2

Revision: 3254

Release Date: 2016-03-15

- Bug fix for very large science data timestamps

4.15.1

Revision: 3250

Release Date: 2016-03-15

- Added in a slight margin for sides with no axis
- Fixed how MAIF sample timestamps were being calculated for old packets with no timestamp

---

4.15.0

Revision: 3235

Release Date: 2016-03-10

- Added in display of the TID to the menu bar so you can easily tell which TID is opened
- Changed the limit monitor's popup dialog to only tell how many HKIDs were triggered instead of how many values
- Decreased the size of the point indicator so that it matches the size of the series points
- Optimizations to the plot times for large series
- Axis settings are now accessed through right-clicks instead of double clicking
- Added back in limit monitor's limit value indicator and point indcator
- Added in the ability to disable the limit monitor popup for 5 minutes
- Slightly increased the minimum zoom rect size
- When creating quick graphs using only the HKID, the graph's title now displays the key instead of just the HKID

- Added in shift functionality to graph that will now display a point indicator for each series

---

4.14.0

Revision: 3187

Release Date: 2016-02-29

4.13.5

Revision: 3158

Release Date: 2016-02-25

- Minor change to SWTS packet size

4.13.4

Revision: 3147

Release Date: 2016-02-24

- When adding a new housekeeping item to the Chromatogram Viewer, there is a little icon on the right side that has search tips
- Added in ability to toggle the legend
- Added in ability to show only the selected series
- Added in ability to display message log and marker text on the charts
- Added in ability to display the "max ion counts" on the Chromatogram viewer

---

4.13.3

Revision: 3122

Release Date: 2016-02-22

- This version contains a pretty stable version of the new Chromatogram Viewer. After some more usage in the lab, the old one will be fully replaced with the new one.
- New features:
  - Uses the brand new charting library that adds a lot of nice functionality:
    - Full multi-axis support
    - Series using anti-aliasing to make them look nicer
    - Ability to display data point labels
    - Legend is displayed above the chart and allows the visibility of series to be toggled
  - All cSIC and cTIC traces now have their own axis
  - Added ability to plot housekeeping items on their own axis

    - All SIC traces are now saved between sessions

4.13.2

Revision: 3097

Release Date: 2016-02-18

- See changelog for v4.13.3 for a full feature list of the new Chromatogram Viewer Beta
- Bug fixes to Chromatogram Viewer Beta
    - Fixed the zoom not being properly set to full scale if the series min value was greater than 0.
    - If a series has no data or the series has all the same y values, the axis now keeps a range that makes sense instead of setting the min and max of range to the same value
    - Corrected counts axis is now correctly hidden on startup
    - Fixed SIC widgets enabled state on startup
    - Scan indicator is now shown properly on first open
    - Gave it its own settings key so it doesn't conflict with the regular Chromatogram Viewer

---

4.13.1

Revision: 3090

Release Date: 2016-02-18

- Fixed the Chromatogram Viewer Beta not being properly reloaded when new TIDs are opened and also not properly updated when used in real-time

---

4.13.0

Revision: 3088

Release Date: 2016-02-18

- If the tm file indicated in the metadata file does not exist, it falls back on the old logic of searching the current TID directory for any file containing "tm.mom".
- ChromViewer Beta!
    - Uses the brand new graph library with lots of changes and new features
    - See changelog for v4.13.3 for a full feature list

4.12.0

Revision: 3083

Release Date: 2016-02-17

- Mass Scan Viewer
  - Added display of filament state for each scan
- Miscellaneous
  - Fixed restoration of saved quick graphs
  - Updated splash page to include info about the status of the metadata file
  - Still attempts to parse and use unsupported metadata file versions (i.e. newer versions that momdataview doesn't explicitly support yet)
  - Fixed skipping of packets larger than 41k bytes. Now supports the max size of 65k.
  - Added in support for EIS packet type 40
  - Added in support for SWTS status packet type 45

4.11.2

Revision: 2980

Release Date: 2016-01-19

- Corrected the J2k offset

4.11.1

Revision: 2937

Release Date: 2016-01-05

- Science data
  - Scan's are no longer sorted by time (reverted from a previous version) and instead are again in the order they exist in the TID. This was causing issues in real-time.
- Mass Scan Viewer
  - Added in graph options to auto grow and shrink the y-axis when viewing scans
- Sequence Viewer
  - Fixed a crash when multiple sequences are selected at once
- Chromatogram Viewer
  - Now uses the same auto-scaling implementation as the HK Graphs
- DAC Scan Viewer
  - Fixed the axis not being automatically updated
  - Empty scan's are now skipped which fixes some scaling issues if a DAC Scan has empty scans. A DAC Scan may have empty scans if the required "END" marker is missing.
- HK Graphs
  - Fixed the static x-window implementation. Before the window would slowly increase on each refresh.

4.11.0

Revision: 2888

Release Date: 2015-12-17

Updated to support version 2 of the tm.meta

- Graphing
  - Optimization to graph zooming. Now zooming should always be smooth, even for very large data sets like the Bus 1 current sample.
- Decoding
  - Added in support for decode type 217 which calculates the laser UV energy
- Mass Calibration
  - Fixed loss of precision when saving and loading the mcal file
  - Now determines the mass cal to use from the tm.meta file
  - Saving the mcal to the TID updates the tm.meta's default mcal field
  - Prompts the user that they're overwriting a file
- Science
  - Added in some extra error handling for corrupted science data
  - Fixed a few corner cases for cTIC calculation
- Marker Viewer
  - Now remembers its "show zero marker" state

4.10.0

Revision: 2823

Release Date: 2015-12-3

- Scan data
  - Added in option in the global science settings to select which science packet types are used
  - Scan's are now ordered by time instead of their index in the telemetry file
- Mass Calibration
  - The mass cal dialog now works directly with the global mass cal. This means that any changes you make to the mass calibration are reflected right away. When closing the dialog, you will be prompted to save the changes. If you do not save, then the mass cal will be reverted back to the original one.
  - Fixed bug where negative intercepts were being rounded to an integer
  - Fixed the m/z -> RF DAC transformation
  - The mass cal dialog's is automatically updated to reflect the currently selected scan's phase cycle
  - Added option to save using the current mass cal's filename
- Sequence timeline
  - The DAC values are cut off at 3 decimal places for now. Eventually an option to set the number of decimal places will be added.

4.9.3

Revision: 2809

Release Date: 2015-11-23

- Beta version of the new sequence viewer timeline view

4.9.2

Revision: 2786

Release Date: 2015-11-17

Bug Fixes

- Fixed the message missing whitespace in the message log
- Fixed how the "sci from eng" column in the database was being applied by patch files

---

4.9.1

Revision: 2768

Release Date: 2015-11-11

Bug Fixes

- Made the Preferences dialog smaller so it fits on the screen better
- DAC Scan Viewer
  - Fixed the DAC scan counts being less than the expected TIC - This was fixed by hardcoding it to only use Summed packets
  - Fixed the number of traces for a DAC Scan changing by switching to a different Scan type and then back - This was due to the "End" marker missing for some data; added in some extra logic to handle these cases
  - Fixed the mass range functionality so that it properly uses each scan's unique transform when calculating the counts

---

4.9.0

Revision: 2749

Release Date: 2015-11-5

Bug Fixes

- Fixed divide by zeros that were occurring when calculating the cTIC

New Features

- Now supports the ability to differentiate between models!
  - This was achieved by adding in full support for the tm.meta file, which details which model a tm file was created with
  - This allows you to select different default paths and files for the different models via the 699config.INI file
  - If you attempt to open a tm file that has a model that does not currently exist in the 699config.INI file, you will be prompted with a dialog to set the key-value pairs

    - This dialog can be opened at any time from the momadataview->Preferences menu (or use the shortcut CMD+,)

Removed

- SciTmViewer
- Open archive file menu option

4.8.1

Revision: 2744

Release Date: 2015-11-5

Bug Fixes

- Now correctly handles Scan Status packets that are out of order with respect to their corresponding sum packet
- No longer incorrectly associates Scan Status packets to scans that are not related
- Using the slider in the Mass Scan Viewer now properly remembers the Scan list's horizontal scroll position

New Features

- Added a "# SS" column to the Scan list, which is the number of scan status packets associated with this scan
- Added the packet indexes for the Scan Status and Science/Sum packets at the bottom of the Scan Info display

---

4.8.0

Revision: 2732

Release Date: 2015-11-3

Bug Fixes

- Mass Scan Viewer:
    - Fixed the "cmd" key causing the latest scan to be deselected
    - Fixed the total scan count not updating if "latest" isn't checked
    - Fixed the packet type filtering capability
    - When summing multiple scans, the "latest" checkbox will be automatically unchecked
- Chromatogram Viewer:
    - The Mass option when using the SIC functionality now works as expected. Before this did not exactly work correctly. It performed a conversion from mass -> bin once, and then used that corresponding bin to determine the ion counts. However, this was wrong because the conversion from mass -> bin is scan dependent! Furthermore, it previously did not support the two step conversion that was required if the MassCal was done with either RF MON or RF DAC.
- DAC Scan Viewer:
    - The scan indicator is now only drawn if the selected scan is currently displayed

- The scan indicator is now properly shown when using the "View DAC Scan" functionality from the Mass Scan Viewer

New Features

- Added in basic support for the new tm.meta file
  - Now searches for the tm.meta file to determine the name of the tm.mom file
  - Full support of all other features coming in the next release
- Added in new Science Settings feature
  - Control the GC AGC parameters from this new settings dialog
- Mass Scan Viewer:
  - Added in a cTIC column to the scan list
  - Added in an Average TIC (TIC/#Sum) column to the scan list
  - Added in option to apply the cTIC factor to the spectrum
  - The phase cycle and beta value are now displayed at the top of the scan's info display
- Mass Calibration:
  - The Mass Calibration dialog now inserts the correct control value irregardless of what the Mass Scan Viewer's current display mode is
  - If a phase cycle does not exist, the user will now be automatically prompted to create a new entry
- Chromatogram Viewer:
  - Improved the Export functionality, now lets you choose which traces to export
  - Added in support for cSIC traces
  - Added in support to have multiple cSIC/SIC traces displayed at once
  - Switching between the Mass and Bin option no longer attempts to do a conversion - The reason for this is because the mass <-> bin conversion is scan dependent
  - Improved the scan indicator
- DAC Scan Viewer:
  - Improved the scan indicator
  - Added in the "Show Data Points" option
- Message Log Viewer:
  - Now automatically fixes messages that were separated between packets, i.e. no more partial messages on their own line
- Graphs:
  - When the graph is set to full scale, the scale stack is automatically cleared now
  - Added in option to display an indicator of the scale stack - useful to know how far zoomed in you are
  - Multiple value graphs now initially load much faster
- Quick graph:
  - Greatly improved its functionality
  - Now supports fuzzy searching of the HK Items in the search bar. For instance, in order to plot MAIF:MAIN_IMON, you can simply start typing "mainim" and the option should pop up. Very useful!
  - Can now be opened via CMD+G shortcut
  - Now defaults to Sci mode
  - Remembers its state between sessions
- Raw Packet Viewer:
  - Editing the current packet index field now pauses the viewer automatically
- Marker Viewer:
  - Columns are now properly sorted by their double value instead of their string representation
- Data interpretation:
  - Equation type 9 now uses the Science K Constants to apply a polynomial to the converted value
- General:
  - Now prompts the user about which TM File to open if a TID has more than one
  - When opening viewers, you can now select more than one at a time
  - When re-opening the TID that is currently opened the tm file will actually be re-opened and parsed

4.7.0

Revision: 2584

Release Date: 2015-09-24

Bug Fixes

- Fixed issue where HK graph did not properly rescale itself when initially blank and then data arrives at a later time
- Fixed graph "blinking" that occurs in the Mass Scan Viewer when the "latest" checkbox is checked and the Command key is held down
- Fixed/improved how the scan summing functionality works. Now the first selected scan is always the one that scans get summed on top of and also the mass calibration that is used
- Fixed crash that occurs when you have 1 or more trace "kept" and then open a new TID

New Features

- Completely redesigned how auto scaling works for HK graphs
  - When zoomed in on data, it will no longer automatically zoom you back out
  - If data exists outside of the current window, then the appropriate axis side will be bolded
  - If new data arrives while zoomed in, the right axis will be colored green to indicate this
  - Added capability to "freeze" the current x axis window
- Added shortcuts for the graph right-click options
- The slider in the Mass Scan Viewer now reflects the currently selected scan

---

4.6.3

Revision: 2555

Release Date: 2015-09-18

Bug Fixes

- Fixed crashing when the Scan Status data does not exist for a scan
- Fixed timestamp calculation for various decoders

New Features

- Added cTIC column to Mass Scan Viewer

4.6.2

Revision: 2491

Release Date: 2015-09-03

Bug Fixes

- Actually fixed the cTIC this time
- Fixed decoder 223, upirani pressure low

New Features

- Added in capability to filter by packet type in the Raw Packet Viewer

---

4.6.1

Revision: 2477

Release Date: 2015-09-02

Bug Fixes

- cTIC now works for Science Packets
- The Mass Calibration Dialog now opens with the phase cycle of the currently selected scan (if one exists)

4.6.0

Revision: 2473

Release Date: 2015-09-01

Significant changes

- New v2 mass calibration format that supports different mass calibrations for different phase cycles and also calibration via the RF values

New Features

- Chromatograph Viewer :
  - now supports displaying the cTIC
  - improved SIC functionality
  - graph crosshair now displays the closest mass spec in the Mass Scan Viewer
  - you can now display the messages and markers on the graph
  - "scan indicator" is now a straight line instead
  - improved export capabilities
- Raw Packet Viewer index input is slightly modified to save space
- Added in a "Load Global" button to the Mass Calibration Dialog
- You can now remove multiple Mass Calibration data points at once
- DAC Scan Viewer scan list is now ordered by the order of the scantypeinfo.json config file
- Mass Scan Viewer's Scan Info now uses the scantypeinfo.json config file for the Scan Id and Scan Value interpretations
- Slightly changed the format of the scantypeinfo.json file and also the default location of it
- Graph crosshair is now triggered via option+left-click instead of the previous shift+left-click because the shift modifier was causing issues with other widgets
- Supports the new Science Summed Packet structure that includes the TIC for each summed scan
- All graphs now support Adaptive Sampling and the ability to display the data points
- "AS" is now drawn to the top left corner to indicate that Adaptive Sampling is being used

- Tooltips in the message and markers left panel are now immediately shown on hover

Bug fixes

- Fixed the incorrect RF Monitor conversion for the last bin segment

Bug fixes

- Fixed the Scan Status packet data being associated with the wrong scan (only for single scans).

4.5.1

Revision: 2321

Release Date: 2015-07-30

Bug fixes

- Fixed the Scan Status packet data being associated with the wrong scan (only for single scans).

4.5.0

Revision: 2303

Release Date: 2015-07-21

Significant change

- Added in support for Equation Type 12: Lookup tables! This means that data conversions can now be done via lookup tables. All lookup tables should exist in the "momagse/TMDef/lookupTables" directory.

Bug fixes

- Changed the name of the "Limit Viewer" to "Expected Value Viewer"
- Fixed a crash when the science data packets have no actual science data. The culprit was the Mass Scan Viewer, and an example TID was 01277.

4.4.1

Revision: 2288

Release Date: 2015-07-16

Bug fixes

- No longer crashes on close

4.4.0

Revision: 2275

Release Date: 2015-07-13

Significant change

- Added in a new Limit Monitor that is constantly watching incoming values.

New Features

- Added in tooltips to all items on the splash page
- Turned off auto coloring of old traces

Bug fixes

- Made a fix to how markers were being extracted by the old limit viewer (which is really the expected values viewer) so that it actually works now

---

4.3.0

Revision: 2238

Release Date: 2015-06-23

Significant change

- v1.0 of the database patch file spec detailed here has been fully implemented. This means that the HKID (data id) is now the unique identifier. One immediate impact this has is that HK Viewers will automatically update to display the names in a patch file. For example, there is no longer a need for a separate QSM MAIF housekeeping viewer because the regular one will automatically replace it's names with the QSM naming as detailed in the QSM patch file.

New Features

- Added in option to color old graph traces a different color.
- Added in t0 in seconds to the splash page display. This is the value that is used to calculate all relative times throughout the application.
- The "Get Info" display now displays the "Mux Channel #" field
- The "Get Info" display now displays the equation type name and also the equation
- The HK Graph's now have a much more useful initial size

Bug fixes

- In the DAC Scan Viewer, when a scan is selected that now has scan data, the icon indicator will now be shown in the scan type drop-down list
- Fixed a memory leak - the viewer's we're not properly destroying their TelemetryFile instance
- The Marker Viewer now initially loads significantly faster for large amounts of markers

---

4.2.3

Revision: 2218

Release Date: 2015-06-15

Bug fixes

- Fixed the DAC Scan viewer not being properly reset between TIDs
- Added back in the accidental removal of the more descriptive DAC Scan trace labels

---

4.2.2

Revision: 2189

Release Date: 2015-06-04

Bug fixes

- Fixed the "Conversion calculator" not converting sci or eng values to the correct raw value when there should be an exact conversion. This was a result of comparing two doubles for equality, and was fixed by doing a fuzzy comparison.

4.2.1

Revision: 2181

Release Date: 2015-06-03

Bug fixes

- Fixed the crashes when plots had no data. The cause was the formatting of very large axis values.
- Fixed the graph marker list rows not updating their tooltips properly

---

4.2.0

Revision: 2178

Release Date: 2015-06-02

New Features

- **Housekeeping Graphing**
  - Added in the ability to display a plot's data points with a circular indicator. This can be accessed by right clicking on a graph and selecting "Show data points"
  - Graph cursor position in lower left corner now always has 4 decimal points

    - You can now plot data of any size, including the 100Hz data, and even the 1000Hz RSIM Bus1 current
    - Added in the ability to display both the message log and markers on a graph. This can be access by right clicking on a graph and selecting "Messages and Markers".
        - You can find the closest message or marker on the graph by either middle clicking or shift clicking on the graph
        - You can select a range of messages or markers to highlight a portion of the graph
- **Other**
    - The Sequence Viewer now fully supports loading a sequence from a file (given it's in the correct format). This can currently be accessed by going to the "File" menu, and then "Open Sequence".
    - The Sequence Viewer now prompts for a sequence ID when saving a sequence to a file
    - Added in a "Converter Calculator" to Hk Item's "Get Info" menu. This allows you to easily convert between raw, eng, and sci values.
    - Changed the MOMA msg log viewer to use "Menlo", a monospace font, by default to help with viewing hex values and tables. "Menlo" is similar to Deja Vu Sans Mono and is shipped with all Macs
    - Minor visual modifications to the Raw Packet viewer
    - Added in a relative time column to the Marker Viewer so that it can be correlated to the other data

Bug fixes

- Added in proper handling of the pre-boot RSIM packets, i.e. now it never uses the pre-boot RSIM packet's timestamp as t0 since that timestamp can be very incorrect, causing all other timestamps to be wacky.
- Fixed bug where a scan's total ion count was not being calculated correctly when it was larger than 65535 per 100 bins (including summed scan data)
- Fixed the seconds counter that is displayed in the toolbar so that it displays the correct seconds even if the packet's timestamp is before the time sync's tick counts.
- Made the Marker Viewer's pause functionality more sane
- Fixed a bug where raw values were not being displayed correctly in the Hk Viewer if they were greater than a signed int's max value
- Fixed the MAIF decoders so that they use the MAIF 1-sec timestamp and they now also account for the ADC sample rate
- "Copy Raw/Eng/Sci" functionality now correctly converts time to relative seconds
- Fixed a bug where using either "Make Hk" or "Quick Graph" was wiping out any loaded patch files
- Fixed the RSIM current decoders so that they properly (or as close as possible) calculate the first sample's timestamp

---

4.1.1

Revision: 2021

Release Date: 2015-04-02

- Added in handling of the pre-boot RSIM packet
- Fixed bug with sequence viewer sometimes displaying the packet #s in the packet count column

---

4.1.0

Revision: 2021

Release Date: 2015-03-31

- Added in the ability to view the spectrum using RF Amp Monitor and RF Amp DAC
- Added in the ability to select multiple scans to view their summed spectrum
- Added in exporting capability for the selected scans
- You can now view a scan's related DAC Scan by right clicking on the scan and also display the scan's indicator on the DAC Scan plot
- I added in a slider to the Mass Scan Viewer in order to quickly scroll through the scans. It's important to note that this purposely does not update when selecting a scan in the list.
- Added in the option to "Verify before saving" a mass cal
- Removed "Journal" functionality in Mass Scan Viewer because it should no longer be needed
- More minor improvements and bug fixes to the scan info display
- Other minor bug fixes
- Tweaks to how the msg log is highlighted when viewing a scan's related msg log

---

4.0.2

Revision: 1931

Release Date: 2015-03-12

- The related portion of the message log is now highlighted
- Added in the RF/Aux start and end values
- Fixed the rod bias DAC not being properly extracted and displayed
- Fixed the pause register values not being properly extracted
- Improved the accuracy of the "Counter per second" value
- Changed the DDS Stats values to be the "science" conversion
- The Scan Info scroll bar now stays at its previous position
- Now properly indicates that a corresponding Scan Status packet was not found, and also properly clears the previous Scan Status state when a new TID is opened
- Other general improvements and more accurate information in the scan info display
- The mass cal is properly handled when a new TID is opened, particularly when the dialog is still open
- Added in usage of a status bar in the mass cal dialog
- Removed "Modify Viewer Params" from the menus

4.0.1

Revision: 1931

Release Date: 2015-03-11

- Fixed a bug that was causing some scans to not display properly. It wasn't interpreting the custom scan values (aux phase, ramp time, etc.) correctly which was resulting in an exception.

4.0.0

Revision: 1929

Release Date: 2015-03-10

- The auto switching of TIDs when a new TID is created will only work with momagse v3.8.3 and up. Communication with the GSE app is now accomplished using a local socket instead of shared memory. This will hopefully fix the issue where sometimes the data viewer doesn't change when a new TID is created.
- Improved the Mass calibration functionality and made it easier to use
- I changed the config file format, so this change will break backwards compatibility with the mass cal files we currently have, which is like 2 I think
- Mass Cal's will be copied from the global directory (momagse/mcals) if none exist. Otherwise, the most recent mass cal file will be automatically loaded.
- Saving a mass cal automatically timestamps it, so you never have to worry about overwriting an old mass cal
- The default masses are now configurable in a file called default-masses.amu which can also be found in momagse/mcal.
- This was really the most important change out of all the changes, so please provide as much feedback on this as possible.
- The Science Data Cache now uses the 0x86 Science Packet ACK ID. This fixes the delay of the Mass Scan Viewer when displaying Science packets
- Added in decode type 210 for the MAIF 1-min packet
- Added in decode type 226 for the RF Frequency
- Added in decode type 203 which can be used to extract the RF/AUX values from the sum and combined packet
- Added in a display of all Scans to the mass scan viewer. This allows you to also view things related to the scan, along with display the currently selected scan on the other custom viewers.
- You can now view a scan's related message logs by right clicking on the scan
- Changed the mass scan viewer to display which mass cal file is loaded.
- Added in an equation type to convert the RF Step and AUX Step sizes
- You can now resize the HK Viewer's font size using cmd- +,-,0
- You can now view HK value's meta data info in the HK Viewer by right clicking
- When a HK View's value is being displayed as raw, the tooltip will now display an integer conversion. I figured we might as well make it used for something instead of displaying "raw"
- Added in the ability to save a sequence to a file.
- Added in ability to close the current window with cmd-w
- TextEdit's search dialogs can now be triggered with ctrl-f and "wrap" is also checked by default
- Reverted Message Log viewer font and font size to the default one
- The splash page now correctly displays if the patch file is not actually found
- Filtered messages in the Message log viewer now correctly display the relative timestamp
- Quick Graph's can now be properly saved and restored, including ones with multiple HKIDs
- Fixed the Marker Viewer slowing down the application when more than ~5000 markers exist
- Minor tweaks to the Config699Dialog
- Moved change log into the 699wiki and added a link to the About Dialog
- The custom viewers can now be properly saved and restored
- Changed the loading of new tm files. It's now smoother and properly displays which step is actually occurring.
- The hex bytes of the selecting sequence's command are now highlighted
- Better usage of the status bar to indicate any events
- Added a slider to the raw packet viewer
- Almost every "text area" now has the search capability which is really useful if you're interested in a specific value or perhaps a sequence of bytes.
- Better time resolution since we have 4 decimals
- Changed indicator of a scan in the DAC Scan viewer to a plot icon
- Everything save's their state much better now between sessions
- Added mass cal file to the splash page

3.3.0

Revision: 1792

Release Date: 2015-02-06

- When an error occurs while opening a viewer, an informative dialog is displayed instead of silently failing
- Added in ability to copy raw values to clipboard
- Added in ability to export DAC scan data to a file
- Added in ability to toggle auto scaling of individual HK graphs (including quick graphs) by right clicking on the graph
- ChromViewer now has a default file name when exporting data
- ChromViewer fat lines are fixed
- ChromViewer has better performance
- ChromViewer now always updates but has same auto scaling toggling as the HK graphs by right clicking on graph
- New spiffy icons

3.2.2

Revision: 1776

Release Date: 2015-02-03

- Added in ability to convert message log timestamps to relative time. The relative time is the same relative time that all other viewers use.

3.2.1

Revision: 1770

Release Date: 2015-02-02

- Fixed the sequence viewer not correctly converting the RF_AMP DAC value
- Minor improvements to the Config699Dialog
- Improved "Make Hk Viewer" and "Quick Graph" initial size, position, and resizing of columns
- Updated versioning to use `<MAJOR>`{=html}.`<MINOR>`{=html}.`<PATCH>`{=html}. Versioning now follows the conventions detailed here
- Minor naming changes to the menus

3.2.0

Revision: 1736

Release Date: 2015-01-26

- Sequence Viewer is now fully functioning. It has a config file that determines how the various values are extracted and interpreted
- The open file dialog now uses the native dialog. The native dialog has a few annoying quirks, but it does allow searching which is a very useful feature, particularly to filter TIDs based upon their name.
- Slight visual changes to the splash page

3.1.0

Revision: 1684

Release Date: 2015-01-07

- The TICs is the default trace in the Chromatogram Viewer, and the range trace is off by default but can be toggled on
- Added in the ability to export the TICs trace of the Chromatogram Viewer
- Added in the ability to update the Chromatogram Viewer during realtime data
- Chromatogram Viewer now auto scales and refreshes the plot when a new Telemetry File is opened
- Chromatogram Viewer now has a total scan count indicator
- Reworked the interface of the ChromatogramViewer so it's a little more user friendly
- Fixed the bug where the Mass Scan viewer wouldn't show the first scan if only 1 existed
- DAC Scan viewer now supports science conversions, and is fully configurable from a config file located at momagse/viewerConfigs/dacscanviewer.json. New scan values can be added this way, and labels, default conversions, etc, can be configured from this file
- DAC Scan viewer now displays the first DAC trace (if one exists) when a new Telemetry File is opened
- Fixed crashes in both the Chromatogram and DAC Scan viewer
- Science Viewer defaults to line mode
- Moved the y axis label slightly off the edge because it made it hard to read
- Fixed the RSIM bus 1 current sample decoding. Previously it was not calculating the timestamp for each sample correctly.
- SequenceViewer exists but is not fully functioning

3.0.0

Revision: 1629

Release Date: 2014-12-09

- Telemetry files now load significantly faster. Files that used to take 5 or more minutes now load in under 30 seconds.
- The splash page now displays some useful meta data about the currently open telemetry file
- A progress dialog pops up when opening a new telemetry file
- The database patches are now correctly loaded when required by the telemetry file, as indicated by the telemetry file extensions
- The Housekeeping Viewers now contain a "latest" checkbox that will make the viewer display the latest value on each update interval. This is now controllable on a per viewer basis and is automatically unchecked when viewing previous data.
- The Housekeeping Viewers now contain a packet index indicator. Note that the total is the total number of all packet types, but the index will only stay on the packet types that are relevant for that specific viewer.
- No more random whitespace in the Message Log viewer! You may still see some messages continued on a new line with no timestamp, but this has to do with the fact that some messages are split up

between packets.
- The Raw Packet Viewer now uses a monospace font! A bunch of other changes to it. Also now contains the ability to seek to any index.
- All viewers now probably reset their state when a new telemetry file is opened and also provide better indicators that no relevant data exists.
- Fixed some corner case crashes
- Mass Scan Viewer now contains a "latest" capability!

MOMA MAIF Voltages

`L 802 -m 50 -s -l 1.4800 -h 1.5000 \# MAIF:CDH_1P5V` https://goo.gl/r6Fd9Q

`L 803 -m 50 -s -l 2.0900 -h 2.1000 \# MAIF:2P7V` https://goo.gl/1XKlk2

`L 812 -m 50 -s -l 1.4700 -h 1.5100 \# MAIF:MAIF_1P5VMON` https://goo.gl/GDR2QN

```
L 813 -m 50 -s -l 3.2500 -h 3.3300 \# MAIF:MAIF_3P3VMON
L 814 -m 50 -s -l 5.000 -h 5.100 \# MAIF:MAIF_5VMON
L 815 -m 50 -s -l 14.500 -h 14.700 \# MAIF:MAIF_P13VMON
L 816 -m 50 -s -l -14.700 -h -14.500 \# MAIF:MAIF_M13VMON
L 817 -m 50 -s -l 16.000 -h 16.500 \# MAIF:MAIF_MOT_VMON
L 818 -m 50 -s -l 3.5000 -h 5.0000 \# MAIF:WRP1_SUM
L 824 -m 50 -s -l 0.2000 -h 0.2500 \# MAIF:HEATER_IMON
L 827 -m 50 -s -l 1.0000 -h 1.2000 \# MAIF:MAIN_IMON
L 828 -m 50 -s -l 0.5000 -h 0.7000 \# MAIF:SEB_50VAC_IMON
L 830 -m 50 -s -l 0.4500 -h 0.7500 \# MAIF:MOTOR_SUP_IMON
```

MOMA Mailing List

The MOMA Mailing list is used for any general announcements pertaining to MOMA data and the data analysis software. If you interested in either of those, it is highly recommended you sign up for the mailing list so you can keep up to date. Sign up at https://lists.nasa.gov/mailman/listinfo/moma-data-and-software

MOMA Mass Calibration

This page details how mass calibration is accomplished for MOMA's mass spectrometer. The mass calibration config file is currently being used by MOMA Data View, our python tools, and XINA Online.

Mass Cal Config File Format

All Versions

Conventions

- Any line that starts with '#' should be treated as a comment and ignored

Fields

- **version** - The version that this config file's format adheres to

Version 1

Fields

- **control_mode** - Defines the control mode that the data was calibrated with. The mapping is as follows:

```
0 = BIN
1 = RF_MON
2 = RF_DAC
```

- **entries** - Each entry defines a mass calibration. Entries are unique by their phase_cycle. A scan's phase cycle should be used to determine which mass calibration is applied.
- **constants** - The constants/coefficients for the mass transform polynomial. index 0 is the constant and so forth. Example:

`[1.0, 2.0, 3.0] --> m/z = 3x^2 + 2x + 1`

- **order** - The polynomials order
- **data_points** - The data points that were used to generate the curve fit. The control value is manually selected and mapped to a mass to achieve the necessary mass calibration.
- **rms** - The root mean square of the curve fit to the data_points
- **phase_cycle** - The phase cycle for this calibration entry. You should use a scan's phase cycle to determine which calibration is applied.

Example

```
{
  "control_mode":2,
  "entries":[
    {
      "constants":[
        0.0,
        1.0,
        0.0
      ],
      "data_points":[
      ],
      "order":0,
      "phase_cycle":-1,
      "rms":0.0
    },
    {
      "constants":[
        3.04691,
        0.568583,
        0.0
      ],
      "data_points":[
        {
          "amu":132.905,
          "control":228.389
        },
        {
          "amu":392.715,
          "control":685.332
        }
      ],
      "order":1,
      "phase_cycle":3,
      "rms":1.0
    },
    {
      "constants":[
        6.34889,
        0.992971,
        0.0
```

```
      ],
      "data_points":[
        {
          "amu":132.905,
          "control":127.214
        },
        {
          "amu":392.715,
          "control":389.253
        },
        {
          "amu":652.525,
          "control":651.162
        },
        {
          "amu":912.335,
          "control":912.074
        }
      ],
      "order":1,
      "phase_cycle":6,
      "rms":0.999999
    }
  ],
  "version":2
}
```

MOMA Meta Markers

This page defines the specification that software should adhere to in order to support **MOMA Meta Markers**. Meta Markers were created to improve our post-processing capabilities by allowing us to add markers *after* the telemetry has already been generated. This functionality may be used for any number of purposes:

- An existing marker is not properly positioned (with respect to the time domain) and does not correctly capture the desired data
- An analysis of the data reveals that more markers would improve data trending
- Markers to indicate "astronaut" events (see also Annotations)
- Markers to indicate anomalous events (see also Annotations)

A Meta Marker consists of

- marker ID (an integer in the range [50,000, 100,000))
- marker text
- start time
- end time

Unlike normal telemetry markers, which are packets inside of a tm.mom file, Meta Markers are generated by post-processing software based on predefined rules. Despite their simple format, human beings do not define Meta Markers in terms of these four fields. Instead, humans define rules which software uses to generate Meta Markers. The format and other details of these rules are described below.

Note that Meta Markers are different from Annotations. Meta Markers are generated by software to augment XINA Online's trending capability, whereas Annotations are generated by human beings using MOMA Data View in order to describe events within a single TID.

Storing Markers

Markers should be stored in a .json file in the momagse/MetaMarkerRules directory.

Example: 50050.json

Meta Marker 50050 is intended to adjust marker 50 by 10 seconds. These fields are defined in a following section.

```
{
 "meta_markers": [
  {
   "tids": "0-39999",
   "meta_marker_id": 50050,
   "meta_marker_text": "Adjusted Background Collection",
   "start_conditions": [
```

```
      {
       "type": "marker",
       "marker": 50,
       "offset_in_seconds": 10
      }
     ],
     "end_conditions": [
      {
       "type": "next_marker"
      }
     ]
    }
  ]
}
```

In this example, any time marker 50 is encountered, a corresponding marker 50050 will also be created.

Meta Marker Rule Format

Note that all .json files in the momagse/MetaMarkerRules directory will be used. So, Meta Marker Rule filenames can be named anything, as long as they have a .json extension.

Root Key

- `meta_markers` - The root node that contains an array of Meta Marker objects.

Meta Marker Object Fields

Each Meta Marker object has the following fields:

- `tids` - **String - Number Range** - Defines inclusive selection using ranges such as "0-29999" or using commas to identify
- `meta_marker_id` - **Integer**. Example: 50050
- `meta_marker_text` - **String**. Example: "Adjusted Background Collection"
- `start_conditions` - **Object** - When any of these conditions are satisfied, the marker starts (i.e., the start timestamp field is set).
- `end_conditions` - **Object** - When any of these conditions are satisfied, and when a Meta Marker is active (meaning that the Meta Marker has started but not yet ended) the marker ends (i.e., the end timestamp field is set).

The `start_conditions` array is a set of trigger objects.

The `end_conditions` array is a set of trigger objects.

Supported Trigger Types

The trigger `type` dictates how the marker's timestamps should be calculated. The different trigger types and their fields are described below.

Marker

The meta marker is defined relative to another marker ID.

- `type` : "marker"
- `marker` - **Number** - The marker ID that the marker is defined relative to.
- `offset_in_seconds` - **Number** - The offset in seconds from the `id`. Negative values offset to the left, and positive values offset to the right.
- `seb_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "SEB Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `high_voltage_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "High Voltage Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `laser_ebt_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "Laser EBT" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `gc_ebt_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "GC EBT" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.
- `heater_test_enabled`: - **boolean** - Optional. If this field is specified, then, when the marker packet arrives, the metamarker will only be generated if the "Heater Test" script configuration parameter is set to the specified value. This field is ignored when the "marker" trigger type is used as an end condition.

Message

The meta marker starts when a line in the message log matches a given regular expression.

- `type`: "message"
- `regex`: - **String** - The regular expression to look for
- `case_sensitive` - **boolean** - true if the regular expression mapping should be case sensitive, and false otherwise
- `offset_in_seconds`: - **number** - The timestamp of the message log line is used as the meta marker start time (if this trigger is used as a start condition) or the meta marker end time. This value will be added to the message log line timestamp before it is assigned as the meta marker timestamp. Negative values are allowed.

Duration

This trigger only applies to end_conditions.

- `type` - "duration"

- `number_of_seconds` - **Number** - the amount of time between the start of the meta marker and the end of the meta marker. If this value is positive, then generated meta markers will end number_of_seconds after the time when their start_conditions are satisfied. If this value is negative, then generated meta markers will *end* when their start_conditions are satisfied, and will start number_of_seconds before that end time.

Next Marker

This trigger only applies to end_conditions. If this trigger is set, then the Meta Marker end timestamp will be equal to the timestamp of the next telemetry marker.

- `type` - "next_marker"

Future Trigger Types

These trigger types are not currently supported, but might be in the future if the need arises.

Time

The marker is defined with a timestamp.

- `type` : "time"
- `mode` - **String** - "tid_relative", "absolute"
- `timestamp` - **Number** - The timestamp of the marker

Housekeeping Value

The marker is triggered by a HK value.

- `type` : "hk"
- `id` - **Number** - The HKID
- `comparison` - The comparison type: "<"
- `value_mode` - **String** - The value's mode: "raw", "eng", or "sci"
- `value` - **Number** - The housekeeping value
- `offset_in_seconds` - **Number** - The offset in seconds from the when the HK value comparison is satisfied. Negative values offset to the left, and positive values offset to the right.

Conventions

If a Meta Marker is relative to a normal telemetry marker (i.e., a marker packet in a tm.mom file), define the Meta Marker ID as `original_mkid + 50000`. This is convention only. If more than one Meta Marker is relative to the same marker, a different offset will be necessary.

Meta Markers cannot be relative to other Meta Markers.

A separate file should be used for each Meta Marker Rule.

Dependencies

This section details the dependencies that the Meta Markers have on the telemetry data. Ideally, any of the telemetry data that Meta Markers depend on should not change, but if it must, then whoever created the Meta Marker should be notified so that the Meta Marker can be updated.

Messages

- "engaging open loop mode"
- "engaging closed loop mode"
- "wrp startup complete"

MOMA Python Packages

This page lists useful Python (and some shell) scripts that facilitate data extraction and analysis, and links to install packages containing these scripts. A more detailed description of library capabilities can be found here.

To determine which version of 699util you have installed, you can type `pip list` in your terminal.

Installation Instructions

For installation instructions, go to the Python Tool Installation page, or contact MOMA software developer.

If you already have a package installed and just want to upgrade to a new package, you can:

- download the latest DMG file from this page
- double click the DMG file to mount it
- Open a terminal
- type `cd /Volumes/699util_Installer`
- type `./install699util.sh`. If you are on MacOS 14+, run with `bash ./install699util.sh` to avoid security warnings.
- restart your terminal.

List of Scripts

Script Name	Description
c_apv_ops.py	count the number of APV operations
c_emcount.py	count ions against the multipliers
c_emruntime.py	determine how long EMs were on
c_expectedvaluechk.py	report actual vs. expected housekeeping values
c_extract_hkid_928.py	stream HKID 928 (the 1 KHz RSIM current) data to a file (this HKID is too dense for c_tmfields.py)
c_filament_cycles.py	count filament cycles
c_ioncount.py	performs various ion counting calculations for MOMA scans
c_lasercount.py	count laser pulses
c_look_for_missing_sci_data.py	print errors for missing science data
c_maifruntime.py	returns the amount of time the WRP was on for the telemetry file
c_momaconsume.py	run various consumable scripts
c_momafilTime.py	determine how long filaments were on
c_momascan.py	plot voltage vs. ion counts

Script Name	Description
c_pktsummary.py	print a table of packet types and number of occurrences
c_plotdatarate.py	plot the instrument's data rate from the Rover's perspective
c_print_meta_markers.py	print metamarkers that apply to the tm.mom file
c_pwm_cycles.py	count PWM cycles
c_sync_moma_model_data.py	sync new lab data to SVN
c_t0.py	print t0 for various time systems
c_tid_to_mzml.py	output science scans in MZML format
c_tidsummary.py	output a JSON file summarizing TID data
c_tmfields.py	output housekeeping values
c_tmmarker.py	print markers in the tmfile
c_tmmsg.py	print the message log
c_tmplot.py	plot housekeeping data
c_tmsequence.py	check the packet sequence numbers and report anomalies
c_tmsummary.py	print a summary of each packet in the telemetry file
c_valve_2_open_time.py	determine how long valve 2 was open
c_valve_cycles.py	count the number of valve cycles across a range of TIDs
add_gse_pkts.py	add or modify the GSE packets at the beginning of telemetry files
boxcar.py	determine boxcar avaerage of two columns of data
checksum.py	calculate the fletcher checksum of files or input from stdin
evref	copy an expected value reference file to current directory and run expectedvaluechk.py
extract_science_data.py	write science data to files
filterExpectedValues.py	filter out ExpectedValue.txt lines for which there are no markers in the tm file
fixPacket1.py	creates tm.mom.fix such that packet 1 is modified to match the cwd
iniget.py	parse an INI file and print a specific value
make_metafile.py	make a tm.meta file for a TID
output_patched_database.py	apply patch files, and print out the resulting database
playtm.py	replay a telemetry file packet by packet
plotref	copy a plot reference file to the CWD and execute it
print_sequence_numbers_and_timestamps.py	what a descriptive name
pump_down_trend.py	trend APV pressure pulse information in LDI mode
rf_chk.py	output RF monitor values for a given bin
runall.py	execute a command across all (or a range of) TIDs
threshold_timestamp.py	print the time that an HKID went above (or below) a threshold
tids.sh	defines convenient functions to jump to TID folders

Script Name	Description
time_between_msgs.py	print the amount of time between messages in a message log (or between two occurrences of the same message)
tmdump.py	generate a hex dump of a telemetry file, separated by packets
tmexcerpt.py	filter data from a telemetry file to a smaller telemetry file
tmkeys.py	parse and print the telemetry database

What's with all this "c_" stuff?

Scripts that begin with "c_" are actually using MOMA Data Viewer code to examine MOMA telemetry files. In many cases, you can omit the "c_" (for example, by running tmfields.py instead of c_tmfields.py) to instead use the pure-Python version of the script. However, we no longer support the pure-Python telemetry extraction library, so its output will become less reliable as time goes on.

I want to write my own script!

MOMA Python packages will create the directory `"${HOME}/custom_699_utils"` and add it to your PATH. So, users can write a Python script that uses the MOMA Python telemetry extraction library, stick it in this directory, make it executable, and then use it like any other script. Future package installs will not modify the contents of this directory, so your scripts will persist.

Bug Reports

Please do not report bugs on this wiki. Instead, use the Bugzilla server we have set up for bug tracking and reporting. Once you have set up an account, you may file a bug for the python software at the page for Command-line Utils.

Releases

5.05.02

DMG (Mac) Installation Package: https://smce-xina.s3.us-east-1.amazonaws.com/dist/699util/699util-5.05.02.dmg

Linux Installation Package: https://smce-xina.s3.us-east-1.amazonaws.com/dist/699util/699util-5.05.02-installer.tar.gz

Release Date: 2026-04-22

- Packaging fixes
- If you already have v5.05.00 installed, you don't need to install this version

3.03.12

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.12.dmg

C++ Revision: 4942

Python Revision: 2278

Release Date: 2017-03-16

- `import c699util` now resolves _c699util.so's shared library dependencies

3.03.11

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.11.dmg

C++ Revision: 4934

Python Revision: 2274

Release Date: 2017-03-15

- Added support for new laser decode types
- Adds new marker_relative_timestamp() method to PythonStyleMomaScan()

3.03.09

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.09.dmg

C++ Revision: 4693

Python Revision: 2252

Release Date: 2017-01-30

- Fixed a bug where MAIF datapoint timestamps were incorrect if they had a smaller timestamp than their packet's timestamp.
- Implemented c699util.get_tmfile()
- Implemented c699util.get_science_data_cache()

3.03.08

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.08.dmg

C++ Revision: 4666

Python Revision: 2247

Release Date: 2017-01-24

- Fixed a bug where the NEXT_MARKER and MARKER end conditions could randomly fail.

3.03.06

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.06.dmg

C++ Revision: 4608

Python Revision: 2239

Release Date: 2017-01-12

- Added support for TVAC Status packet (type 42).

3.03.05

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.05.dmg

C++ Revision: 4592

Python Revision: 2234

Release Date: 2017-01-10

- c_expectedvaluechk.py now supports temperature dependent values.

3.03.04

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.04.dmg

C++ Revision: 4592

Python Revision: 2233

Release Date: 2016-12-16

- Released c_bus_energy.py.

3.03.03

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.03.dmg

C++ Revision: 4592

Python Revision: 2231

Release Date: 2016-12-15

- Fixed an issue found by Brad where c_wrp_cycles.py was the wrong name, causing install699util.sh to fail.
- Included c_print_meta_markers.py in the package release.

3.03.02

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.02.dmg

C++ Revision: 4592

Python Revision: 2229

Release Date: 2016-12-15

- Fixed an issue found by Brad where regular expression matching in the message log (which happens with the new meta marker "message" trigger) wasn't stripping away the timestamp and whitespace before checking for a match.

3.03.01

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.01.dmg

C++ Revision: 4591

Python Revision: 2227

Release Date: 2016-12-14

- The "message" trigger now has an offset_in_seconds field
- The "message" trigger can now be used as an end condition

3.03.00

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.03.00.dmg

C++ Revision: 4589

Python Revision: 2222

Release Date: 2016-12-14

- c699util objects now have docstrings, which are available through the help() function.
- meta markers are no longer generated when TMFile and Science Caches are constructed, which speeds things up. Now, meta markers are only generated if one of the get_meta_markers() methods is called.

- implemented "message" meta marker start condition
- added optional script configuration fields to the "marker" start condition
- the "marker" and "next_marker" end conditions will now only end a meta marker if the timestamp of the marker packet is greater than or equal to the meta marker's start time

3.02.03

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.02.03.dmg

C++ Revision: 4509

Python Revision: 2215

Release Date: 2016-11-15

- Implemented decode type 230, to support HKID 408 -- SUM PKT TIC

3.02.02

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.02.02.dmg

C++ Revision: 4483

Python Revision: 2200

Release Date: 2016-11-04

- MOMA Python tools are now aware of the metamarkers defined in momagse/MetaMarkerRules.
- `c_print_meta_markers.py` can be used to see what metamarkers apply to a tm.mom file, and to print highly specific metamarker rule file errors
- Metamarkers can be used for filtering wherever a normal marker would be used. For example: `c_tmfields.py 811 --mkid 50050`
- bogus SWTS1 and SWTS2 packet timestamps are now corrected

3.01.01

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.01.01.dmg

Revision: 2175

Release Date: 2016-10-14

- Markers are now assigned to packets based on timestamp, rather than packet index. **Note that many scripts still extract datapoints from packets, and assume that those datapoints always have the same markers as their parent packets.** MOMA marker assignment is a work in progress.

3.01.00

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.01.00.dmg

Revision: 2167

Release Date: 2016-10-03

- This release includes the new MOMA Data View timestamp resolver, which properly handles timestamp resolution when resets occur. Using the same time resolution code for MOMA Data View, Python scripts, and XINA should eliminate time consistency issues.
- evref output is now sorted first by MKID, and next by HKID

3.00.02

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util/699util-3.00.02.dmg

Revision: 2156

Release Date: 2016-08-30

- Fixed a bug where (c_)sync_moma_model_data.py didn't have access to (c_)emcycles.py.

3.00.01

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util-3.00.01.dmg

Revision: 2155

Release Date: 2016-08-30

- c_expectedvaluechk.py and evref now read XINA exported limits. They are no longer compatible with old limits files.

3.00.00

DMG (Mac) Installation Package: https://s3.amazonaws.com/699_bin/699util/699util-3.0.0.dmg

Revision: 2151

Release Date: 2016-08-22

- We are now releasing MOMA Python packages.

MOMA Python Time Synchronization

This applies to r1920.

The code responsible for packet timestamp resolution is the BufferedMomaPacketList class inside of momapacketlist.py.

- Terminology:
  - **FSW timestamp**: the 4 byte integer at the end of every packet header. This integer represents the number of 100us ticks since the last reset near the time when the packet was created. (This article is going to pretend that the FSW has no other kinds of timestamps.)
  - **time packet**: a FSW time sync packet (type 2) or a digital status packet with non-zero time fields (type 7) which can be used to convert FSW timestamps to UTC. Time packets contain a FSW timestamp along with an equivalent J2000 timestamp, so time packets can be used to go from FSW timestamp -> J2000 -> UTC.
    - The "equivalent J2000 timestamp" I mentioned is actually Rover Elapsed Time (RET), which does not have a final definition (meaning, the UTC time equivalent to RET = 0 has not been defined). When MOMA is integrated with the rover, RET may no longer be J2000.
    - Marker packets should be considered time packets, too, however when I wrote momapacketlist.py, I thought that packet types 2 and 7 were the only packets that could be used to resolve time fields. I need to fix this after I'm finished working on more important stuff.
  - **earliest packet timestamp**: the packet timestamp with the smallest UTC value. Packets in a tmfile are not in timestamped order, so this is not always the first packet's timestamp. (By "first", I'm referring to the first packet inside of a tm.mom file.)
    - Note that the timestamps of lines in a message log are not used when determining the earliest timestamp in a tm.mom file. Because of this, and because message log line timestamps can be earlier than the timestamp of the message log packet that contains the message log, lines in a message log can have negative relative times. This is normal and expected behavior.
  - **first packet timestamp**: the timestamp of the first non-GSE packet in a tm.mom file.
- Converting a FSW timestamp to UTC requires pairing that packet with a **time packet**.
  - For **packets that occur before the first time packet**, that first time packet is used to convert the FSW timestamps of those packets.
  - For **packets that occur after the last time packet**, that last time packet is used to convert FSW timestamps of those packets.
  - **All other packets** are either themselves time packets (so they can just use themselves to convert their FSW timestamps), or non-time packets that have both a preceding time packet and a following time packet. (Digital status packets are a type of time packet that come out every few seconds, so there's almost always a time packet close to any given packet).
    - If the two time packets that precede and follow a packet have identical time synchronization fields (remember that "time synchronization fields" refers to a FSW timestamp paired with an equivalent J2000 value), then either one may be selected to convert the packet's FSW timestamps to UTC.
    - However, if the preceeding and following time packets are different, a reset probably occurred, and each middle packet needs to be paired with either the preceeding time packet or the following time packet:
      - If a packet's FSW timestamp is smaller than the FSW timestamp seen for the same packet type, then that packet, and subsequent packets of the same type, are

considered post-reset packets and are paired with the time packet that follows them.
    - If a packet's FSW timestamp is larger than the last FSW timestamp seen for the same packet type, then that packet is considered a pre-reset packet, and is paired with the preceding time packet.
    - If a packet is the first of its type in the tm.mom file, then that packet is paired with a time packet based on the following logic:
        - If the packet is a message log packet containing "MOMA CDH Boot Loader", then that packet (and subsequent message log packets) are paired with the following time packet.
        - If the packet's FSW timestamp is smaller than the following time packet's FSW timestamp, then that packet is paired with the following time packet.
        - If the packet's FSW timestamp is larger than the preceding time packet's FSW timestamp, then that packet is paired with the preceding time packet.
        - Otherwise, the "closer" time packet is selected. For example:
            - time packet t1 has a header with an FSW timestamp of 1,000,000, and
            - time packet t2 has a header with an FSW timestamp 14,000,
            - and packet p has a header with an FSW timestamp of 20,000, then p will be paired with t2, because ABS(20,000 - 14,000) < ABS(20,000 - 1,000,000)
            - Note that the FSW timestamps in the **packet headers** are the only things considered when making the decision in this edge case; time synchronization fields outside of the packet headers are not used to make this decision.
- Rover Simulation (RSIM) timestamp correction
    - **The problem**: Before MOMA is turned on, a tm.mom file may be accumulating RSIM packets, which could have large arbitrary headers. A common situation is to have RSIM packet FSW timestamps of 0, 10000, 20000, ..., 1040000, 1050000, etc. Then, when MOMA is turned on, the first time packet might have a FSW timestamp of, say, 20000. When that time packet is applied to the previous RSIM packet that had a FSW timestamp of 1050000, that RSIM packet will appear to have occurred much later than the time packet, even though the RSIM packet was created **before** the time packet.
    - When the first digital status packet with non-zero time synchronization fields is seen, the RSIM will alter the FSW timestamps in subsequent packets to have FSW timestamps that are close to the packets that MOMA is outputting. These new FSW timestamps are considered accurate by the Python tools.
    - **The solution**: If the FSW timestamps of "bad" RSIM packets are all 1 second apart, then the Python tools will change the interpretation of their FSW timestamps. The last "bad" RSIM packet will be set to 1 second before the first "good" RSIM packet, the second-to-last "bad" RSIM packet will be set to 2 seconds before the first "good" RSIM packet, etc.
    - If the FSW timestamps of "bad" RSIM packets are not all 1 second apart from each other, the Python tools will not correct the timestamps, and will output a warning.
- Relative time
    - After FSW timestamps are converted to UTC and RSIM timestamp correction is applied, the MOMA Python tools will examine all packets and select the timestamp of the **~~earliest~~ first** packet as t0. Before February 2016, the earliest packet was used. However, this was changed to the first packet to be consistent with momadataview, among other reasons.
- GSE packets
    - The first two packets in a tm.mom file are "fake", GSE generated message log packets. The timestamps of these two packets are set to match the timestamp of the earliest packet.

MOMA Script Validation Process

MOMA Scripts are thoroughly reviewed and tested before being approved for flight following the MOMA Script Validation Process. The validation process involves one or more (sometimes many more) meetings with representatives from the science, systems, electrical and software teams review each line of the script as well as one or more runs on the software testbed and engineering test unit. The intent of the process is to be thorough while still reacting quickly within the surface operations strategic and tactical flow.

MOMAScriptProcess.png

Scripts that are in the review process or that have already been approved are identified as "Activities" and listed in XINA. See link below.

TODO Fix Link > XINA Payload Uplink Activities

The activity on XINA maintains the all the notes, versions, durations, energy, etc. that go along with the script (or in some cases scripts). During surface operations as well as integration and test activities with the ExoMars rover, the XINA activities will be accessible to the ExoMars team for access to all the support data (primarily duration, energy and data volume) and restrictions (such a environmental temperature and pressure) associated with a scipt.

For surface operations, at a minimum, the following three documents will be attached to the activity: 1. Meeting Notes. Includes attendees names, action items, etc. 2. Safety Critical Check List. Spreadsheet contain specific checks for safety critical items the will be completed by the reviewers. 3. Restrictions Document. List any environmental, electrical or other states that restrict the state MOMA must be in to run the script.

In addition, fields in the XINA contain the duration, energy, data volume, filename and version, file checksum, review status (Approved for Flight, Approved for ETU, Under Development) and other keys items.

The scripts themselves are text files and they are maintained in the NASA GSFC Subversion Version Control system (SVN). SVN maintains each checked in version and automatically replaces fields within the file such that the script can identify its SVN version number when it runs. After a script is approved for flight, it is "blessed". In this process, the filename is changed to contain the revision ID number of the script and placed in another SVN folder reserved for flight approved scripts.

Note that as of this writing in August 2016, the method of delivering an approved script to surface operations team or the integration and test team at TAS-I has not been defined.

MOMA SEB Voltages

MOMA Trending Links to XINA pages trending the telemetry tracked by the expected values tables.

L 001 -m 50 -s -l 4.950 -h 5.0550 # SEB:CTL_+5D_VMON https://goo.gl/Qk2mZF

L 002 -m 50 -s -l 13.000 -h 13.900 # SEB:CTL_+13A_VMON https://goo.gl/QoFHZH

L 003 -m 50 -s -l -13.600 -h -13.250 # SEB:CTL_-13A_VMON https://goo.gl/FzEuwX

L 004 -m 50 -s -l 1.5000 -h 1.5080 # SEB:CTL_+1.5D_VMON https://goo.gl/nfXBiK

L 005 -m 50 -s -l 3.3000 -h 3.3500 # SEB:CTL_+3.3_VMON https://goo.gl/k1EHS8

L 006 -m 50 -e -l -10.00 -h 10.00 # SEB:N_-5KV_MON https://goo.gl/32S5nR

---

https://goo.gl/nSofs2 Note that on TID 30165, the delay after turning on SEB was short, so during background collection, it was still stabilizing to 0. That is why the min/max are much bigger on that TID than others.

L 008 -m 50 -s -l -5.000 -h 5.0000 # SEB:RF_AGC_MON https://goo.gl/gU2cvi

---

L 009 -m 50 -s -l -1.800 -h 0.5000 # SEB:GC_EC(HV1-1)_VMON https://goo.gl/Q3J6nd

---

L 017 -m 50 -e -l -0.1000 -h 0.1000 # SEB:IS_EMON_A https://goo.gl/TG0CSg

---

L 018 -m 50 -e -l -0.1000 -h 0.1000 # SEB:IS_EMON_B https://goo.gl/Kk28je

---

L 019 -m 50 -s -l 0.900 -h 1.0000 # SEB:FIL_VMON https://goo.gl/i9Xu0a

---

L 020 -m 50 -s -l 0.0100 -h 0.0150 # SEB:FIL_IMON https://goo.gl/XA1tlv

---

L 022 -m 50 -e -l -0.100 -h 1.0000 # SEB:EM-1(HV2-3)_VMON https://goo.gl/fYZG7d This plot shows the same sudden change in value on the same TID as did HK 009. Believe this indicates a change in the database not

reflected in previously mined data.

L 023 -m 50 -e -l -0.100 -h 1.0000 # SEB:EM-2(HV2-4)_VMON https://goo.gl/No44D3 Also shows a shift between TIDs 30339 and 30342.

L 024 -m 50 -e -l -1.000 -h 1.0000 # SEB:SRC_A_FOC_A(-100_HV1)_VMON https://goo.gl/8Gcd14 Also shows a shift between TIDs 30339 and 30342.

L 025 -m 50 -e -l -1.000 -h 1.0000 # SEB:SRC_B_FOC_A(-100_HV2)_VMON https://goo.gl/8QjIwa Also shows a shift between TIDs 30339 and 30342.

L 026 -m 50 -e -l -0.100 -h 1.0000 # SEB:LDI_EC(HV1-2)_VMON https://goo.gl/b77OsC Also shows a shift between TIDs 30339 and 30342.

L 027 -m 50 -e -l -0.100 -h 0.5000 # SEB:ROD_BIAS(HV1-3)_VMON https://goo.gl/1C3GcX

L 028 -m 50 -e -l -0.100 -h 0.5000 # SEB:OUTPUT_A(EI_ACCEL)_VMON https://goo.gl/T1t1E6

L 029 -m 50 -s -l 4.9500 -h 5.0500 # SEB:CTL_+5VREF_VMON https://goo.gl/8aNgcH

L 030 -m 50 -s -l -5.0500 -h -4.950 # SEB:CTL_-5VREF_VMON https://goo.gl/pcl63Y

L 031 -m 50 -s -l 2.5000 -h 2.8500 # SEB:RF_PRES_MON https://goo.gl/v3pqdb

Noted that the RF Pressure changes with ambient barometric pressure at GSFC and MI

L 062 -m 50 -e -l -0.100 -h 0.1000 # SEB:RF_AMP_MON https://goo.gl/Diunps

---

MOMA System Locale

The system locale is a FSW variable that can be set, and the scripts can use to determine what model we are using.

- 0 or 1 = Flight
- 2 = ETU
- 3 = GB2
- 3 = QSM
- 5 = GSE Lab (F109)
- 6 = FSW lab (Tom's lab)
- 7 = EIS

MOMA Trending

SEB Voltages

MAIF Voltages

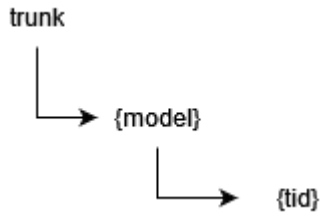
Temperatures

XINA Data Mining

Overview

Telemetry archives (also called "TID's") are imported into XINA by the process described here. "TID" is short-hand for "Telemetry Identifier", a text string that uniquely describes the experiment contained in the telemetry archive.

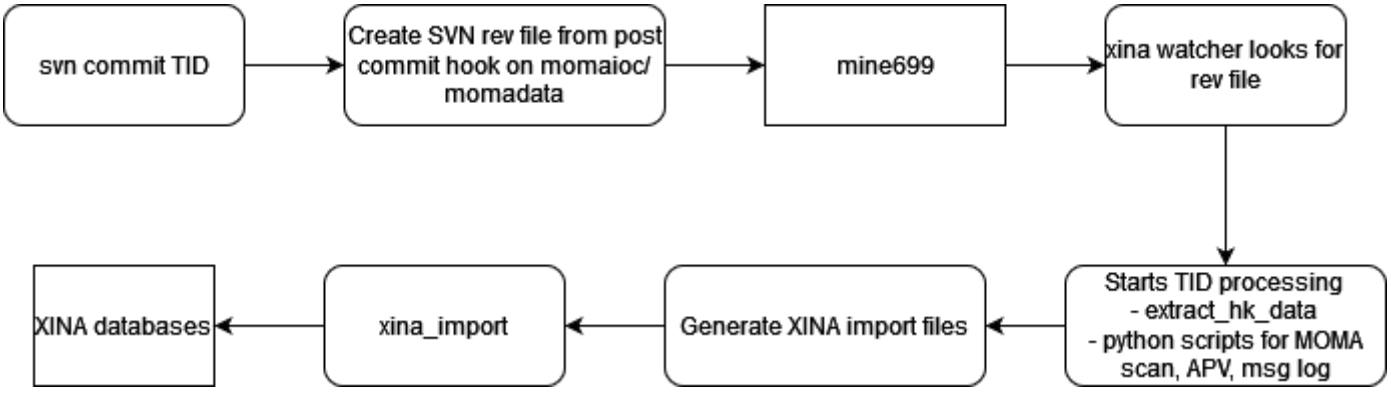
The SVN repository is typically laid out using the following heirarchy:



where {model} matches the XINA database name and {tid} is a unique description of the telemetry archive.

The XINA Data Mining process uses a shared network drive from the mining server (mine699.gsfc.nasa.gov) to the IOC server (e.g., MOMAIOC/DRAMSIOC/etc). Each mission uses a mission-specific folder on this shared drive to perform the data mining processes described below (e.g., /mine699/missions/moma/).

This flow chart shows the major steps taken during the data mining process:



Importing files to XINA

1. The user commits a TID to Subversion (SVN).
2. On MOMAIOC/DRAMSIOC/etc):

- An SVN post-commit hook calls the "svnlook" utility to generate a list of all revision changes in standard format
- "svnlook" output is saved to `/mine699/missions/*/rev/{rev#}.rev`.

3. The **XINA Commit Watch** utility running on MINE699 detects new rev file. Each mission has a separate running instance of this utility.

- Runs command(s) as specified in configuration file: `/mine699/missions/*/config.watch.json`
- Standard setup runs two commands: *XINA Commit* and *XINA Import*

- Once completed successfully, the rev file is moved to the rev archive: `/mine699/missions/*/rev/archive/`

4. The **XINA Commit** utility is run by *XINA Commit Watch* (above)

- Runs command(s) as specified in configuration file: `/mine699/missions/*/config.json`
- Reads in rev file generated by svnlook
- For each file listed in the rev file, the file is checked out from that revision into a local directory: `/mine699/missions/*/local/{file}`
- An additional directory is provided for temporary storage for mining processes if required: `/mine699/missions/*/temp`
- Files can only be processed individually
- After single file mining operations are finished, the local and temp directories are emptied

5. The **XINA Import** utility is run by *XINA Commit Watch* (above)

- Requires the **XINA Tunnel** utility to be running in order to connect to the XINA server (this is always kept running in the background on mine699)
- Reads and imports files in alphanumeric order from `/mine699/missions/*/import/{timestamp}_{rev#}`
- Moves the files as they are completed to the current revision archive directory: `/mine699/mission/*/archive/`
- Because the import process relies on operations happening sequentially, any errors shut down the import process until they are repaired

Handling Import Errors

XINA Commit Watch will not move on to the next rev file until the current file has been completed successfully

- If any operation in *XINA Commit* or *XINA Import* fails, the process returns an error code
- When *XINA Commit Watch* reads an error code from any sub-process it writes an empty file into the mission revision directory: `/mine699/mission/*/rev/rev.lock`
- As long as a **rev.lock** file is present *XINA Commit Watch* will not attempt to process any rev files

Debugging

Log output from *XINA Commit Watch*, *XINA Commit*, and *XINA Import* is stored in ```/mine699/mission/*/log```. - Errors generated during the "commit phase" are generally easier to fix, and my indicate a problem with a configuration file or script. After a commit phase bug is fixed, the rev.lock file can simply be deleted. *XINA Commit Watch* will perform a clean attempt to re-process the entire revision. - Errors generated during the "import phase" can be more complicated to fix because some operations may have already been imported before the failure, and running them again may cause new errors. It is necessary to check the archive directory for completed operations before attempting to re-process the revision. It may be preferable to perform corrections on the mined files depending on the error (for example, a typo in a field name), import the files manually, transfer the rev file to the rev archive manually, and then remove the rev.lock file.

Preload

For mining previously committed data, each mission has directory ```/mine699/mission/*/preload```

- Full checkouts of each mission are kept in `/mine699/mission/*/data`. This checkout is not related to the *XINA Commit Watch* process.
- Scripts are provided in each preload directory to create new task directories: `setup.sh task_name`
- Generated directories require a **config.json** file
- Provide scripts to process and import the backlog:
- **commit.sh, commit_bg.sh** - run across all files (in foreground or background)
- **commit_from.sh, commit_from_bg.sh** - run starting at a TID (in foreground or background)
- **import.sh, import_bg.sh** - import all generated files (in foreground or background)