

Web-Based SAC-D/Aquarius Telemetry Monitoring Tools

I have put several perl scripts and sample files into the following FTP directory,
ftp://samoa.gsfc.nasa.gov/pub/norman/SAC-D_Aquarius_telemetry_monitoring_software/.

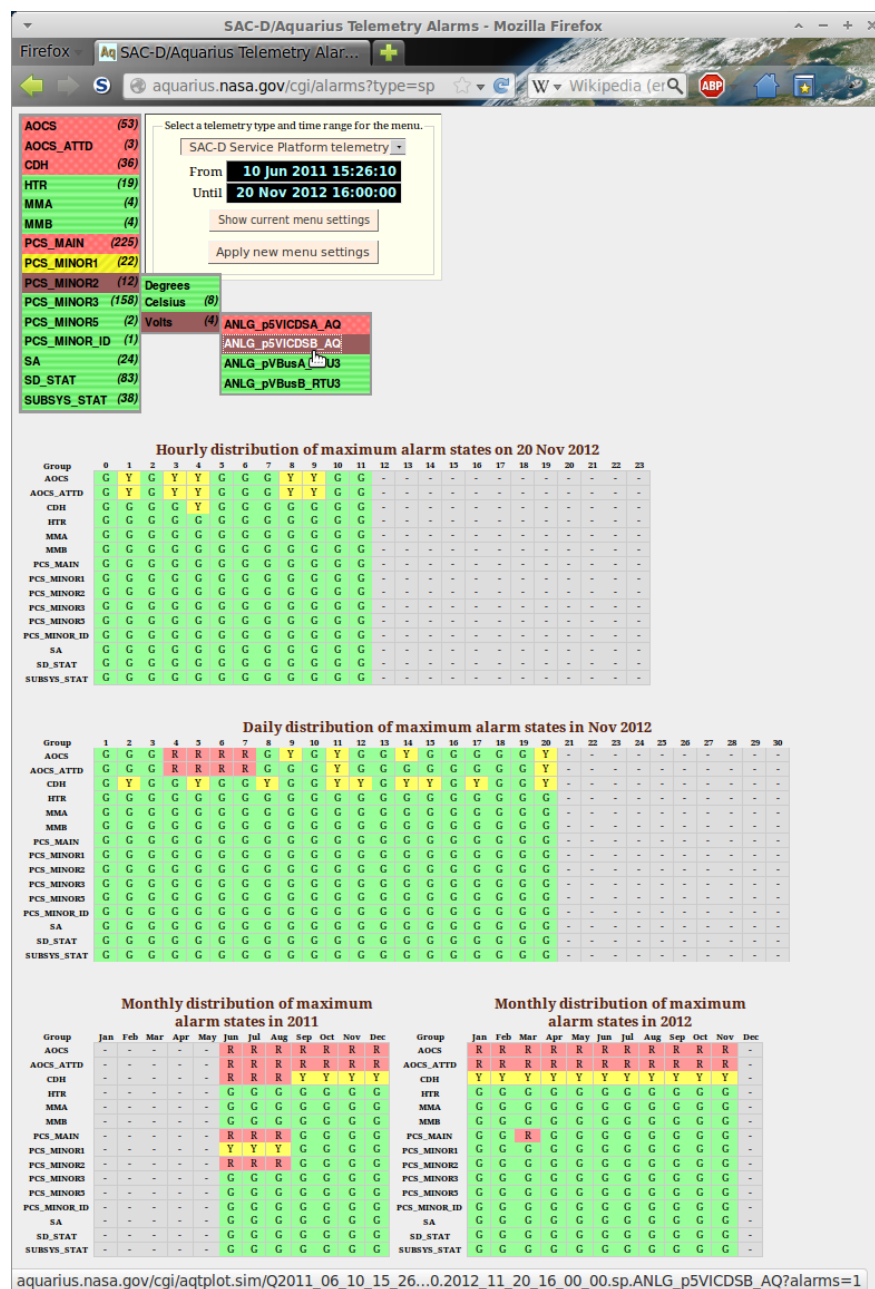
What follows is a brief explanation of which scripts are used to generate the SAC-D and Aquarius telemetry monitoring tools that I have made available on the Web. Full details about the operation of the scripts can be discovered by examining their code.

Norman Kuring <norman@seawifs.gsfc.nasa.gov>

23 November 2012
updated 12 January 2013

The web page, <http://aquarius.nasa.gov/cgi/alarms?type=sp> (Figure 1), is generated by the perl script named [alarms](#).

Figure 1



Information about the maximum alarm state reached by each telemetry parameter that we monitor is stored in binary files with names like the following.

[Q2011.sp.bin](#) (*Service Platform metadata for 2011*)
[Q2012.sp.bin](#) (*Service Platform metadata for 2012*)
[Q2011.aq.bin](#) (*Aquarius metadata for 2012*)
[Q2012.aq.bin](#) (*Aquarius metadata for 2012*)

The [alarms](#) script reads those files using the [ReadTelemMeta.pm](#) module to determine which colors to paint the menu and tables shown in Figure 1.

The four alarm states are:

Value	Meaning	Color
0	Missing	Gray
1	Nominal	Green
2	Warning	Yellow
3	Error	Red

These values are stored as 2-bit integers for each of the Service-Platform and Aquarius telemetry parameters that we monitor. If a given parameter has the same alarm state through an entire year, then only that single value needs to be stored in the Q*.bin file. If the alarm state of a parameter varies over time, then alarm states are broken down by month, then day, then hour, then minute, and finally second until the maximum-alarm variability is fully captured. (*Note that the Aquarius metadata files are substantially smaller than the Service-Platform files because the former currently represent much less alarm variability than the latter do.*)

The above Q*.bin files are created by the [make telem exceedance meta.pl](#) perl script which runs from our crontab every half hour to keep them up to date. This script uses two parameter index files which are included in the [SAC-D MOC ACCS Joint Configuration Management](#) tool.

[ServicePlatform_parameter_index.txt](#)
[Aquarius_parameter_index.txt](#)

These files contain the alarm threshold values for each telemetry parameter and, along with the [units.txt](#) file, give the units of each parameter. They also indicate the category that each parameter belongs to and which column in the category file the parameter values are to be found. The formats of these index files are given in the file headers.

The [make telem exceedance meta.pl](#) script creates the aforementioned Q*.bin files from outputs from Liang Hong's Analysis Tool. These Analysis Tool outputs are grouped into categories (15 for the Service-Platform and 18 for the Aquarius instrument). Sample filenames follow.

Analysis Tool Output File Names

Service-Platform files from stored HKT data:

CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_AOCS_ATTD.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_AOCS.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_CDH.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_HTR.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_MMA.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_MMB.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MAIN.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MINOR1.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MINOR2.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MINOR3.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MINOR5.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_PCS_MINOR_ID.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_SA.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_SD_STAT.txt
CGSS_20121120_102300_10020121120100705_SACD_HKTMST.bin.SP_T20121119_160654_20121120_101151.SP_SUBSYS_STAT.txt

Service-Platform files from level-1A data:

Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_AOCS_ATTD.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_AOCS.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_CDH.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_HTR.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_MMA.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_MMB.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MAIN.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MINOR1.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MINOR2.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MINOR3.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MINOR5.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_PCS_MINOR_ID.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_SA.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_SD_STAT.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.SP_SUBSYS_STAT.txt

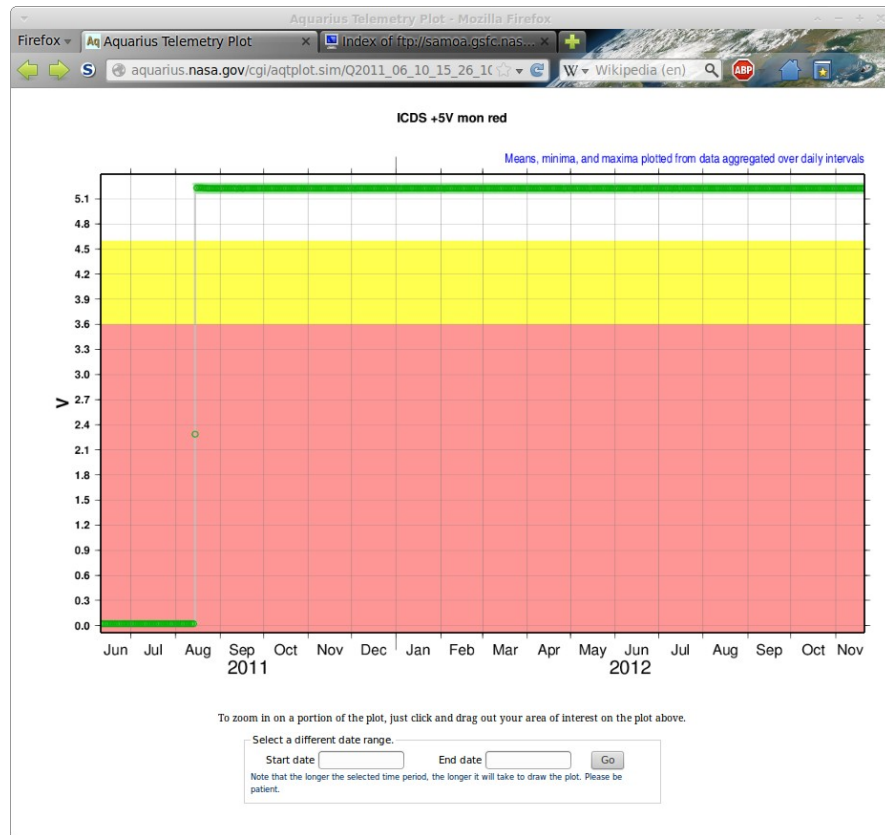
Aquarius files from level-1A data:

Q20121120_101854_T20121119_225500_20121120_005259.AQ_APDU.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ATC_OMT1.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ATC_OMT2.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ATC_OMT3.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ATC_RBE.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_DP.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_EXTT_OMT1OMT2.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_EXTT_OMT3SFE.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_EXTT_RFL.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ICDS_Eng.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_ICDS_Proc.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RD_DPURSTA.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RD_DPU.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RD_NRHK.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RFE1RBE1.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RFE2RBE2.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_RFE3RBE3.txt
Q20121120_101854_T20121119_225500_20121120_005259.AQ_SCAT.txt

Each record in all of the above files consists of a time stamp followed by a series of white-space-delimited parameter values. The number and order of parameters in each category (e.g. AOCS, PCS_MINOR3, ATC_OMT2, etc.) must match what is stored in the parameter index files mentioned above.

If you click on one of the menu items shown in Figure 1, you get a plot such as is shown in Figure 2.

Figure 2



The web page shown in Figure 2 is generated by the [aqtplot.sim](#) perl script. That script allows a user to control which time period is plotted and to drag out a region to zoom in on within the plot. The plot image itself is generated by the [aqtplot.sim.pl](#) perl script which uses the [psxy](#) program and others from the [Generic Mapping Tools](#) (GMT) package. *(Note that the “.sim” in these and a few other script names originated during pre-mission development when I was supporting both real integration and test data as well as simulated data streams. I should get rid of those vestigial name fragments some day.)*

Parameter plots are generated from three separate data sets. Plots of periods longer than 80 days, such as the one shown in Figure 2, are generated from daily aggregates of the telemetry values. These aggregates are stored in files like the following.

Daily aggregate telemetry parameter statistics for 2012

Aquarius

Q2012.day.AQ_APDU.txt
 Q2012.day.AQ_ATC_OMT1.txt
 Q2012.day.AQ_ATC_OMT2.txt
 Q2012.day.AQ_ATC_OMT3.txt
 Q2012.day.AQ_ATC_RBE.txt
 Q2012.day.AQ_DP.txt
 Q2012.day.AQ_EXTT_OMT1OMT2.txt
 Q2012.day.AQ_EXTT_OMT3SFE.txt
 Q2012.day.AQ_EXTT_RFL.txt
 Q2012.day.AQ_ICDS_Eng.txt
 Q2012.day.AQ_ICDS_Proc.txt
 Q2012.day.AQ_RD_DPURTSTA.txt
 Q2012.day.AQ_RD_DPU.txt
 Q2012.day.AQ_RD_NRHK.txt
 Q2012.day.AQ_RFE1RBE1.txt
 Q2012.day.AQ_RFE2RBE2.txt
 Q2012.day.AQ_RFE3RBE3.txt

Q2012.day.AQ_SCAT.txt

Service Platform

Q2012.day.SP_AOCS_ATTD.txt
 Q2012.day.SP_AOCS.txt
 Q2012.day.SP_CDH.txt
 Q2012.day.SP_HTR.txt
 Q2012.day.SP_MMA.txt
 Q2012.day.SP_MMB.txt
 Q2012.day.SP_PCS_MAIN.txt
 Q2012.day.SP_PCS_MINOR1.txt
 Q2012.day.SP_PCS_MINOR2.txt
 Q2012.day.SP_PCS_MINOR3.txt
 Q2012.day.SP_PCS_MINOR5.txt
 Q2012.day.SP_PCS_MINOR_ID.txt
 Q2012.day.SP_SA.txt
 Q2012.day.SP_SD_STAT.txt
 Q2012.day.SP_SUBSYS_STAT.txt

Each of the above files contains 1830 records (366 days in 2012 times 5 statistics [n, mean, median, minimum, maximum]) with one column in each record for each parameter in the category (CDH, HTR, SD_STAT, etc.) . The records in these aggregate files do not contain timestamps because each group of five records represents a known day of the year.

The daily aggregate files are generated from the Analysis Tool output files by the [condense_hkt_day.pl](#) and the [combine_condensed_dailies.pl](#) perl scripts which are run periodically by cron.

If a plot of less than 80 days is requested, then the [aqtplot.sim.pl](#) script reads 5-minute aggregates of the telemetry parameters. These are stored in files with names like the following.

Five-minute aggregate telemetry parameter statistics for 15 November 2012

Aquarius

Q2012320.5min.AQ_APDU.txt
Q2012320.5min.AQ_ATC_OMT1.txt
Q2012320.5min.AQ_ATC_OMT2.txt
Q2012320.5min.AQ_ATC_OMT3.txt
Q2012320.5min.AQ_ATC_RBE.txt
Q2012320.5min.AQ_DP.txt
Q2012320.5min.AQ_EXTT_OMT1OMT2.txt
Q2012320.5min.AQ_EXTT_OMT3SFE.txt
Q2012320.5min.AQ_EXTT_RFL.txt
Q2012320.5min.AQ_ICDS_Eng.txt
Q2012320.5min.AQ_ICDS_Proc.txt
Q2012320.5min.AQ_RD_DPU.txt
Q2012320.5min.AQ_RD_DPURTSTA.txt
Q2012320.5min.AQ_RD_NRHK.txt
Q2012320.5min.AQ_RFE1RBE1.txt
Q2012320.5min.AQ_RFE2RBE2.txt
Q2012320.5min.AQ_RFE3RBE3.txt
Q2012320.5min.AQ_SCAT.txt
Q2012320.5min.AQ_SUBSYS_STAT.txt

Service Platform

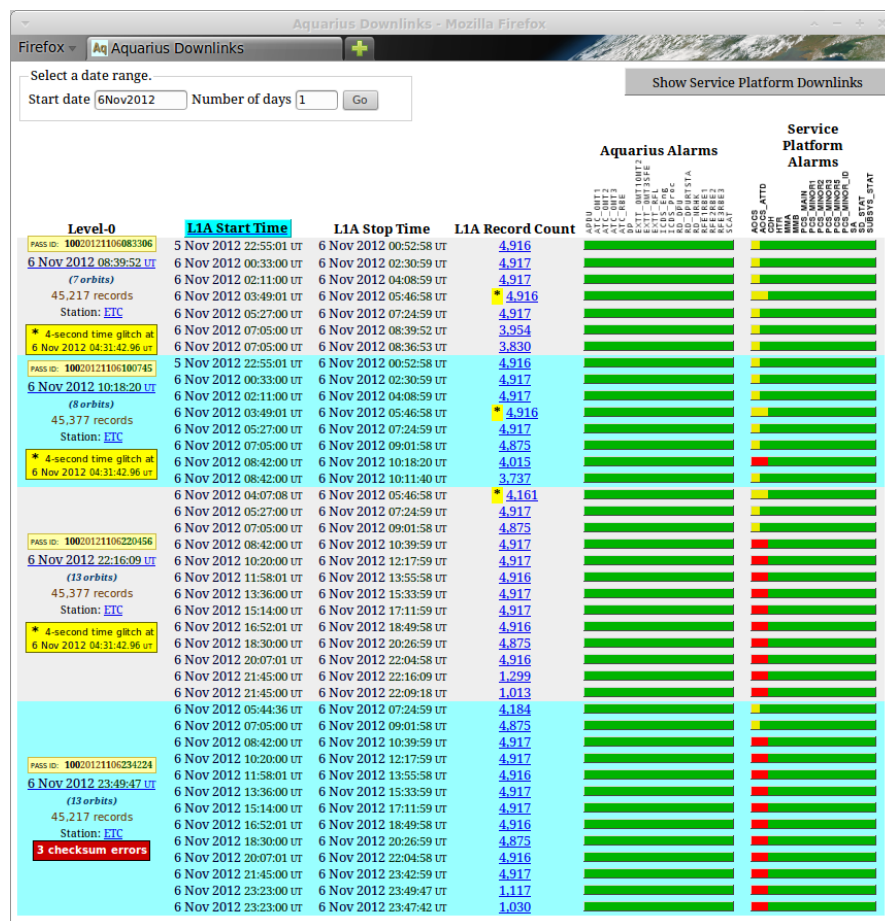
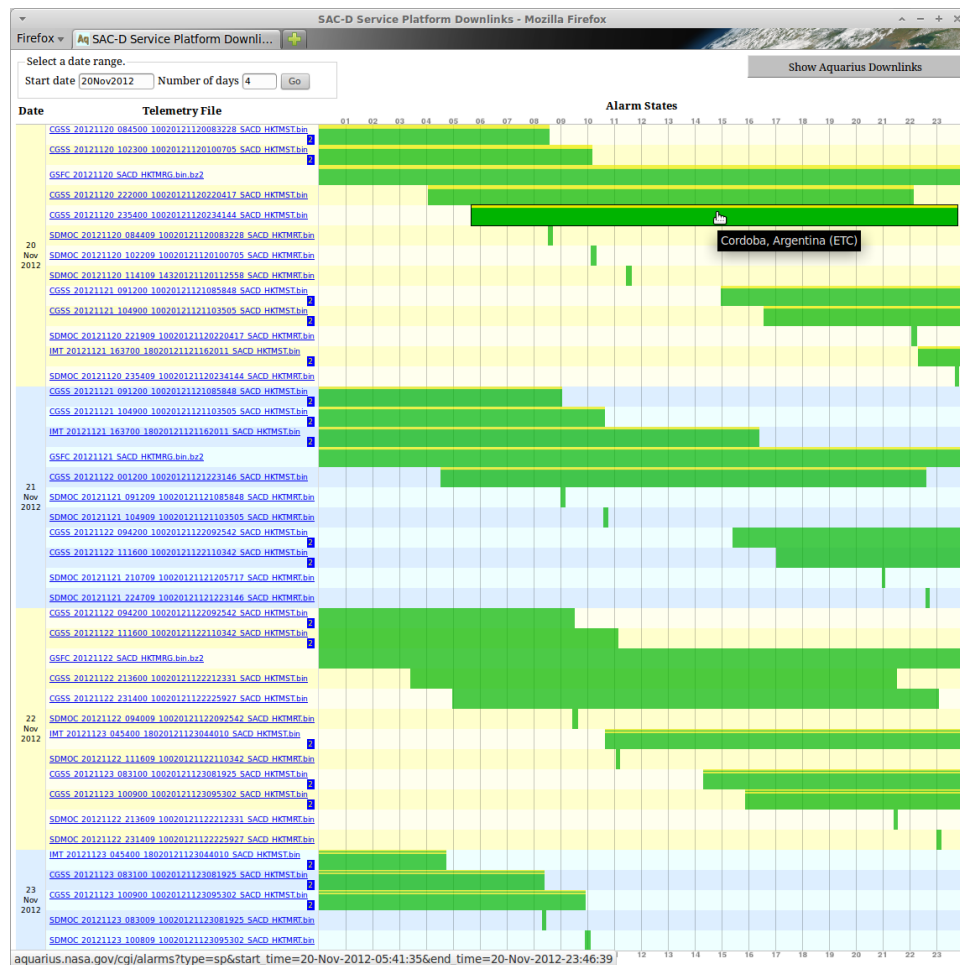
Q2012320.5min.SP_AOCS.txt
Q2012320.5min.SP_AOCS_ATTD.txt
Q2012320.5min.SP_CDH.txt
Q2012320.5min.SP_HTR.txt
Q2012320.5min.SP_MMA.txt
Q2012320.5min.SP_MMB.txt
Q2012320.5min.SP_PCS_MAIN.txt
Q2012320.5min.SP_PCS_MINOR1.txt
Q2012320.5min.SP_PCS_MINOR2.txt
Q2012320.5min.SP_PCS_MINOR3.txt
Q2012320.5min.SP_PCS_MINOR5.txt
Q2012320.5min.SP_PCS_MINOR_ID.txt
Q2012320.5min.SP_SA.txt
Q2012320.5min.SP_SD_STAT.txt
Q2012320.5min.SP_SUBSYS_STAT.txt

Each of the above files contains 1440 records (288 5-minute periods per day times 5 statistics [n, mean, median, minimum, maximum]) with one column in each record for each parameter in the category. Once again, no timestamps are needed because each 5-record group represents a fixed 5-minute period.

The five-minute aggregate files are generated from the Analysis Tool output files by the [condense_hkt.pl](#) perl script which is run periodically by cron.

If a plot of one day's worth of data or less is requested, then the [aqtplot.sim.pl](#) script reads the Analysis Tool output files directly.

Other web interfaces that lead to telemetry plots via the [alarms](#) interface of Figure 1 (and the similar [aqtalarms.sim](#) interface) are coded in the [service_platform_downlinks](#) perl script (whose output renders as shown in Figure 3) and the [aquarius_downlinks](#) perl script (whose output renders as shown in Figure 4).



Automated Alarm Notification

Running in parallel with, but somewhat independent from, the metadata-generating scripts mentioned above are two scripts that work in concert to notify – via email and text message – a select group of recipients whenever any monitored telemetry parameter strays outside of yellow or red thresholds for more than one measurement per input file. The input files for these two scripts are the same Analysis-Tool output files (*from level-1A data*) mentioned above.

The [latest_telem.pl](#) perl scrip runs every 20 minutes from the crontab and calls the [aqtalarmtally.pl](#) script to generate alarm reports like [Q20121120_101854_T20121119_225500_20121120_005259.AQ_L1AQL.txt](#) for each new level-1A file that produced Analysis-Tool outputs. The reports contain both a plain text and an HTML representation of the information, and they look like Figure 5 in my email reader.

To aqops@seawifs.gsfc.nasa.gov

Link to L1A file
=====

http://oceandata.sci.gsfc.nasa.gov/cgi/getfile/Q20121107_091006_T20121107_073300_20121107_090430.AQ_L1AQL.bz2

Link to Service Platform alarm interface
=====

http://aquarius.nasa.gov/cgi/alarms?type=sp&start_time=7-Nov-2012-07:33:00&end_time=7-Nov-2012-09:04:30

Group/Parameter	>= Red Hi	<= Red Lo	>= Yellow Hi	<= Yellow Lo	Missing
AOCS	12(0%)	695(2%)	449(1.3%)	619(1.8%)	
Raw_crnt_mode		679(98.8%)			
real_dot_qds_0			8(1.2%)	11(1.6%)	
real_dot_qds_1	2(0.3%)	2(0.3%)	16(2.3%)	14(2%)	
real_dot_qds_2			22(3.2%)	23(3.3%)	
real_qds_0	6(0.9%)		90(13.1%)	178(25.9%)	
real_qds_1	4(0.6%)	11(1.6%)	64(9.3%)	69(10%)	
real_qds_2			184(26.8%)	234(34.1%)	
real_ts_wheels_0			14(2%)	18(2.6%)	
real_ts_wheels_1			5(0.7%)	12(1.7%)	
real_ts_wheels_2			46(6.7%)	60(8.7%)	
RW_Pz_speed		3(0.4%)			
AOCS_ATTDD	550(26.7%)	917(44.5%)	18(0.9%)	63(3.1%)	9(0.4%)
pitch	201(29.3%)	386(56.2%)	18(2.6%)	63(9.2%)	3(0.4%)
roll	167(24.3%)	517(75.3%)			3(0.4%)
yaw	182(26.5%)	14(2%)			3(0.4%)

—Q20121107_091006_T20121107_073300_20121107_090430.html—

Figure 5

Service Platform Telemetry Alarms

Group/Parameter	Red Low	Yellow Low	Yellow High	Red High	Missing	Total
AOCS	695	619	449	12		34350
Raw_crnt_mode	679					687
real_dot_qds_0		11	8			687
real_dot_qds_1	2	14	16	2		687
real_dot_qds_2		23	22			687
real_qds_0		178	90	6		687
real_qds_1	11	69	64	4		687
real_qds_2		234	184			687
real_ts_wheels_0		18	14			687
real_ts_wheels_1		12	5			687
real_ts_wheels_2		60	46			687
RW_Pz_speed	3					687
AOCS_ATTDD	917	63	18	550	9	2061
pitch	386	63	18	201	3	687
roll	517			167	3	687
yaw	14			182	3	687

The data presented above were gathered from:

- [Q20121107_091006_T20121107_073300_20121107_090430.AQ_L1AQL.bz2](#)

SAC-D/Aquarius Location in Orbit

The interface shown in Figure 6 is generated by the [whereis_aquarius](#) perl script.

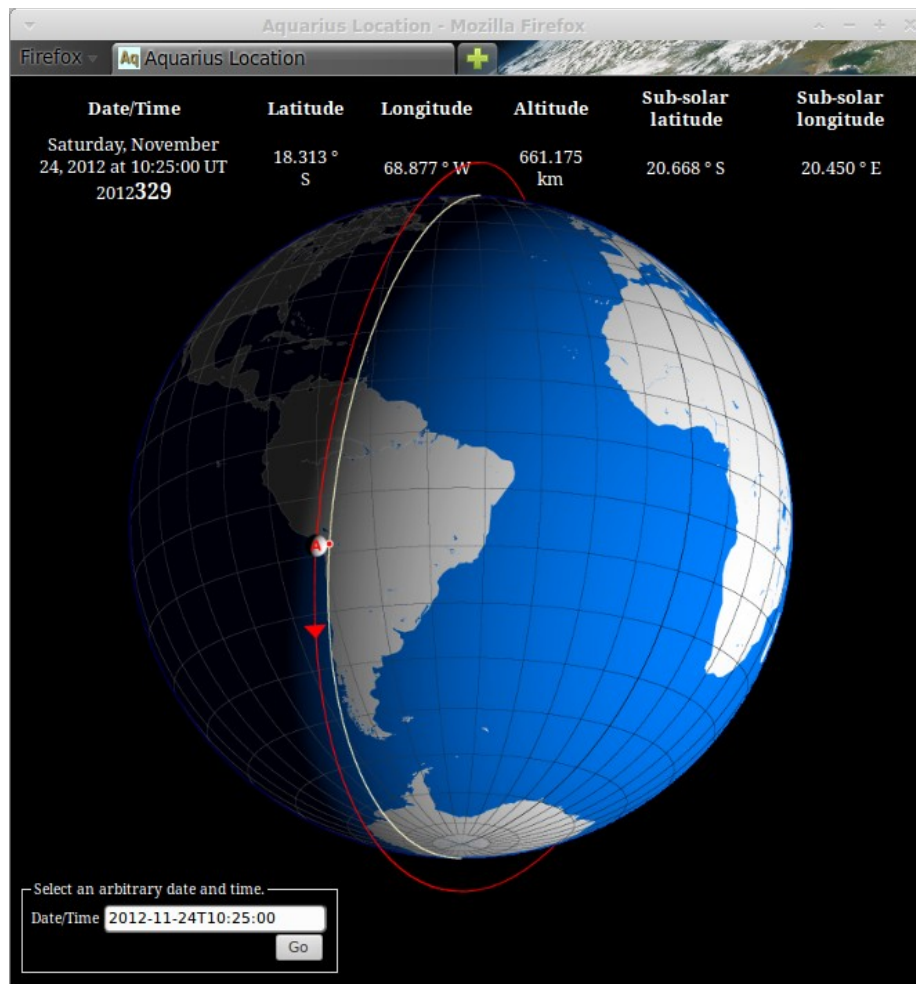


Figure 6

The [whereis_aquarius](#) script reads satellite position files with names like the following.

Q2012325.pos.txt
Q2012326.pos.txt
Q2012327.pos.txt
Q2012328.pos.txt
Q2012329.pos.txt
Q2012330.pos.txt
Q2012331.pos.txt

Each of these files contains one day's worth of minute-by-minute SAC-D positions. They are created by the periodically executing [regroup_cods_records.pl](#) perl script which reads CODS files like [CODS_20121119_135723_SACD_PREDEPHEM_GEOD_LLH_O.TXT](#) as input. (I reorganize the original CODS files to make it easier for my software to locate any arbitrary time period.)

The [whereis_aquarius](#) script calls several [GMT](#) programs like [pscoast](#) to draw its maps.