Adapting to the Challenges of Extended Mission: How Chandra Changed High Radiation Safing

Sabina B. Hurley[1], Eric Martin[2], Paul Viens[3], and Brent Williams[4]
Northrop Grumman Aerospace Systems, Redondo Beach CA, 90278

and

Michael Juda[5], *Smithsonian Astrophysical Observatory, Cambridge MA, 02138*

With over twelve years of on-orbit experience, the Chandra X-Ray Observatory operations teams have become adept at addressing new challenges within fixed resources and capabilities. Looking ahead, the team saw that in 2012 a combination of changes in the orbit, elevated solar activity, increasing thermal concerns, and reduced resources would put significant stress on the processes that have kept Chandra safe and highly effective for many years. To address this risk head on, the program undertook an initiative to change how Chandra protected itself from elevated radiation levels. Instead of stopping all scheduled on-board mission commanding, including attitude maneuvers, the vehicle would now only stop commanding to the sensitive science instruments. This would allow management of spacecraft constraints (e.g., those related to unit temperatures or angular momentum accumulation) to continue while the science instruments were protected, and make recovery to normal science operations more efficient. While philosophically simple, the change touched all elements of the program including the planning and commanding systems, flight load verification tools, on-board flight software, and ground-based data processing. Implementing such a system wide change with a staff sized for maintenance and deeply entrenched processes and tools was challenging, but with strong collaboration and shared commitment, ultimately achievable. This paper will walk through implementation of the new radiation safing paradigm from recognition of a risk item, through design, review, development and final implementation. This paper will also highlight challenges encountered and the solutions found and focus on areas of best practice.

I. BACKGROUND

The Chandra X-ray Observatory, one of NASA's Great Observatories, is a space-based telescope designed to observe X-ray sources. Like any telescope, Chandra must be able to point at many locations in the sky and set up its instruments to correctly observe a wide variety of science targets. Chandra has two science instruments (SIs) that can be placed at the focal point of the telescope, the High Resolution Camera (HRC) and the Advanced CCD Imaging Spectrometer (ACIS), and two transmission grating arrays which can be inserted into the path of the X-rays to provide spectroscopy data. The SIs are housed in the Science Instrument Module (SIM), which contains a translation table and motor to move the SIs into and out of the observatory focal plane.

---

[1] Flight Operations Manager, Chandra X-ray Observatory, 60 Garden St, CAMA/MS33, Cambridge MA, 02138
[2] Pointing Control and Aspect Determination Engineer, Chandra X-ray Observatory, 60 Garden St, CAMA/MS33, Cambridge MA, 02138
[3] Lead Systems Engineer, Chandra X-ray Observatory, 60 Garden St, CAMA/MS33, Cambridge MA, 02138
[4] Mission Planning Manager, Chandra X-ray Observatory, 60 Garden St, CAMA/MS33, Cambridge MA, 02138
[5] Flight Director, Chandra X-ray Observatory, 60 Garden St, CAMA/MS33, Cambridge MA, 02138

Pointing of the observatory is provided by the Pointing Control and Aspect Determination (PCAD) subsystem. The PCAD subsystem includes gyros, an aspect camera, sun sensors, and reaction wheels to monitor and control where the telescope is pointing at any given moment. During observations the Aspect Camera Assembly (ACA) is used to track stars and fiducial lights (point sources fixed in the spacecraft body frame), which are used to provide a highly accurate pointing solution. The reaction wheels are used to orient the vehicle, maintain pointing and store accumulated angular momentum. Small thrusters are used to unload angular momentum from the reaction wheels as required.

Chandra's electrical power is provided by two solar array wings. The energy generated is distributed by the Electrical Power Subsystem (EPS), which also includes a bank of three batteries to provide power during eclipses. Chandra goes through two earth eclipse seasons annually and also experiences occasional lunar eclipses. Each eclipse is analyzed weeks before it occurs and ground commanding is prepared to configure the EPS subsystem to best manage the transition from solar arrays to batteries and back.

To keep the instruments and spacecraft components operating correctly and the telescope alignment sufficiently accurate, temperatures on Chandra must be well controlled. The thermal subsystem uses several methods to provide this control. Cooling radiators, insulation, heaters, thermostats, and reflective surfaces are used to keep each component within its specified thermal limits. As the spacecraft has aged the efficacy of the multi-layer insulation (MLI) for keeping unit temperatures stable and within limits has degraded. More specifically, the sun side of the vehicle has warmed over time, causing exposed units on the sun side to near or exceed their maximum operating temperatures. To control temperatures, the sun exposure of these units must now be limited. As the angle between the Sun and the spacecraft (pitch angle) changes, units receive more or less direct sunlight, depending on their position. So thermal control of several units on Chandra is now closely tied to sun angle. Figure 1 depicts the thermal sensitivities by pitch angle that Chandra mission scheduling and operations must now account for.

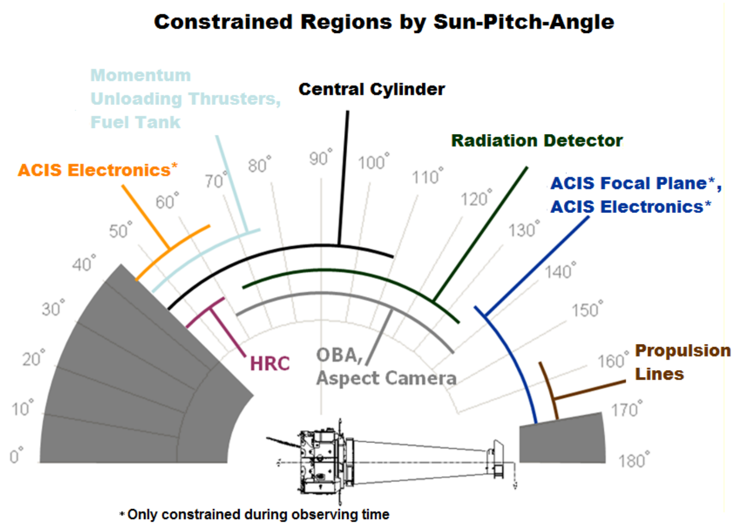

Figure 1. Chandra thermal sensitivity by pitch angle. *The diagram depicts the sun-pitch-angles available for observations with Chandra and the thermal constraints in effect in each region. Some constraints are more restrictive than others, but there is now no region where Chandra is unconstrained for observing time and few regions where Chandra can dwell indefinitely while not observing.*

Linking the subsystems and components together is the communications, control, and data management (CCDM) system. The CCDM system contains redundant on-board computers (OBCs), which run the flight program, or Flight Software (FSW). The FSW, among its many other functions, processes ground commands, which can be sent directly or be loaded into a Mission Stored Command Sequence (SCS).

The Mission SCSs are used to execute the observing program. A week's observing schedule is converted into time-tagged commands, broken into approximately 72 hour pieces, and then built into SCS loads. These SCS command loads are uplinked several days in advance and executed from memory, thus sending the commands needed to configure the spacecraft for each observation at the pre-planned time. As each SCS slot runs to completion (i.e., executes its last command) it is replaced with one several days in the future. Three slots are used in a rotation to continuously execute commands from memory and ensure sufficient commanding on-board to withstand a communications outage.

The FSW also runs monitors which are designed to keep Chandra safe. If the conditions of a monitor are violated the FSW will call Protected SCSs (SCSs permanently stored in Chandra's protected memory) to reconfigure the spacecraft. The SIs are sensitive X-ray detectors and are susceptible to damage from high fluxes of high-energy particles. Therefore one of the on-board monitors is designed to detect elevated radiation rates and command the instruments to a safe state when the rates reach pre-determined levels. With the SIs safed, the observing program must stop. Since the observing program is executed from the Mission SCSs, the sequence that safes the SIs also stops the Mission SCSs to ensure no further, potentially damaging, commanding is sent to the SIs. By stopping execution of the Mission SCSs all pre-planned on-board commanding is stopped. Attitude maneuvers, momentum unload commanding, and eclipse set-up activities are all stopped along with the SI commanding.

Early in the mission, this scheme of stopping all ground based commanding once high radiation was detected worked fairly well. Most attitude commanding was executed to deliver the spacecraft to the proper orientation for observations, observations that would no longer take place. Before the current thermally-driven attitude restrictions were required, the spacecraft could usually safely dwell at the attitude where high radiation was detected for long enough for the radiation storm to pass. When dwelling at that attitude could be hazardous, other safing monitors would prevent any constraint violations. The angular momentum build-up would not be as planned, but autonomous momentum handling could fire the thrusters if they were needed. While it is not ideal to transit an eclipse without the set-up commanding, the flight software would detect an eclipse and configure the EPS system appropriately. Furthermore, the eclipse seasons were short, so this was only a problem for several weeks in a year.

However, with Chandra now more than 7 years past its design life, degraded thermal protection and orbital changes have made it increasingly undesirable to dwell without planned spacecraft commanding. For thermal reasons, it is now difficult to stay much longer than planned at most attitudes. As Fig. 1 shows, there is now no region of the pitch-angle space that has no applicable thermal constraint. Several thermal constraints do not have on-board protective monitors, so a long dwell can cause constraint violations that would not be detected by the on-board safing subsystem. Momentum unloading is now constrained due to the potential for nucleate boiling in the thruster feed tubes during long duration unloads executed at high temperatures. Additionally, a mission low perigee altitude (see Fig. 2) has increased gravity gradient effects substantially, making momentum build-up more difficult to manage. Orbit changes have also made eclipse seasons longer and eclipse entry very sharp, so set-up commanding is more important than ever. For these reasons, the flight team often needed to respond to radiation safing by building fast turn-around schedules to manage thermal and momentum constraints and provide proper eclipse handling. With each rapid response, the case for change was building.

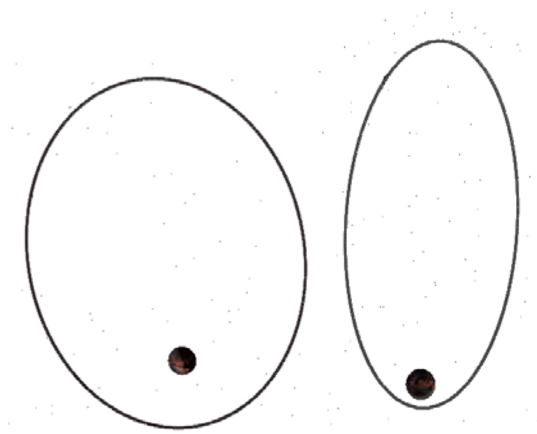

Figure 2. Depiction of Chandra's changing orbit. *The diagram shown on the left-hand side shows Chandra's orbit in June of 2007, the right-hand side in June of 2012. As is clearly shown, the orbit is far more eccentric and the perigee altitude far lower in 2012. These orbital characteristics impact many aspects of operations including, most notably, angular momentum management and eclipse profiles.*

II. THE CONCEPT

If planned attitude maneuvers, momentum handling and eclipse commanding could continue after high radiation is detected without posing a risk to the science instruments, then the need for fast turn-around response schedules would be eliminated. Allowing these commands to continue would improve spacecraft safety and allow planners and engineers to immediately focus on return to science. Further, if the attitude profile continued on, then the planned thermal profile would stay intact. Keeping the thermal profile intact would prevent the increasingly common need to either rearrange the observing schedule or plan a dwell at a cool attitude before restarting to reach necessary initial temperatures for the observing plan. So, if a safe implementation could be found, changing Chandra's response to high radiation could improve spacecraft safety, free up resources to focus on the end goal, and reduce scheduling inefficiencies due to thermal constraints.

All of the commanding needed to achieve a weekly observing schedule is built into the Mission SCSs. Commands are time ordered, with commands to science instruments, spacecraft hardware and flight software all intermingled. To allow attitude, momentum handling, and eclipse commanding, which we will refer to as vehicle commands, to continue while commanding to the science instruments, which we will call observing commands, stops, the two types of commands need some type of differentiation. While the flight software could be modified to do the differentiation, it would be a fundamental change to an integral part of the flight code, and therefore not a desirable solution. A process that could split the commands into sets on the ground would be preferable.

A scheme that split the vehicle and observing commands into separate SCSs was presented to the community. With the commanding in separate SCSs, the radiation safing command sequence, SCS 107, could be modified to stop only the observing SCSs, thus creating a "Science Only Safing Action" (SOSA). The concept of splitting vehicle and observing commands into separate SCSs had one immediate hurdle to overcome. Mission SCSs are loaded into numbered slots. Each slot has a specified size, i.e., it holds some maximum number of commands. The original design only had three slots large enough to feasibly be used for the mission schedule. These slots were large enough for either vehicle or observing loads, but a split load scheme would require at least four, and preferably six, large slots. Creating additional large slots required a change to the flight software. The required change was analyzed and assessed to be feasible and well isolated (i.e., did not cause an unwieldy number of downstream changes).

With this technical hurdle cleared, the tasks required to implement a split load scheme to provide a science only safing action were as follows:

1) Use ground-based processes to divide the commanding for a weekly observing schedule into "vehicle" and "observing" commands.
2) Redistribute the space for the Mission SCSs to create six large slots rather than three very large slots. Ensure the slots are sized appropriately to account for the nominal proportions of vehicle to observing commands.
3) Modify the ground-system command load generation software to build vehicle commands into one set of SCS slots and observing commands into another set.
4) Change the on-board SI safing sequence to stop only the observing SCS slots.
5) Review, and modify as required, all on-board safing sequences to account for the change to the SI safing sequence.
6) Modify ground procedures and software to account for the split load sets.

III. DECIDING TO PROCEED

The concept of a science only safing action had clear benefits and the split load implementation appeared feasible. However, implementation also represented a large expenditure of program resources, required program wide collaboration, required change to a portion of the FSW not previously patched, and modified processes and procedures ingrained through over a decade of successful operations. So implementation itself presented risk. Therefore, deciding to implement SOSA required a thorough review of risks versus benefits.

As of June 2010, Chandra had stopped the mission loads and safed the SIs due to high radiation 64 times, accounting for approximately 80% of all mission safing actions. The number of thermal constraints managed through the attitude profile had gone from zero at launch to five in June of 2010. As shown in Fig. 3, with each new thermal constraint the probability of needing a "maneuver only load" to put the vehicle at an acceptable pitch angle for the thermal constraints increased. Radiation had been quiet over the past few years, so the full impact of the newer thermal constraints on radiation response had not yet been experienced, but solar max was

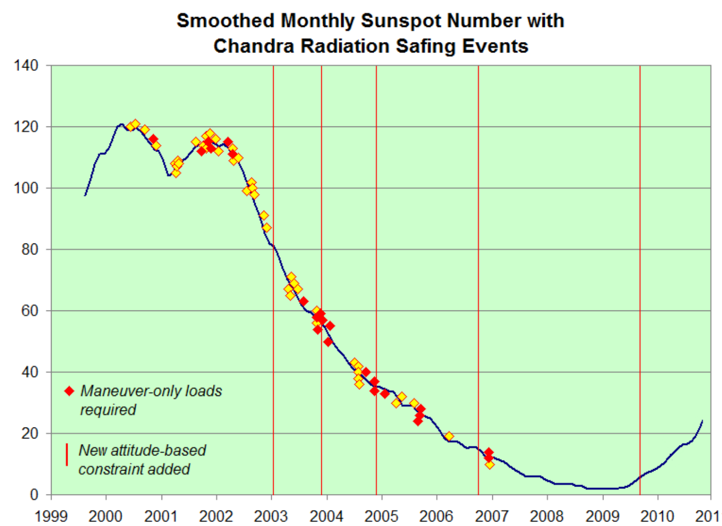

Figure 3. Monthly number of sun spots with Chandra radiation safing triggers. *The number of monthly sunspots indicates the activity level of solar radiation storms. With higher levels, radiation safing events on Chandra are more common, as is indicated by the diamonds indicating times when Chandra was safed for radiation. Note that with each new thermal constraint the fraction of radiation safing events that required a "maneuver only load" increases.*

approaching within the next few years. Of the few safing actions that did occur in the previous several years, most required maneuver only loads.

Stopping all spacecraft commanding for high radiation was estimated to cost approximately 150 hours of labor for each of the three years centered on solar max. This estimate was arrived at by (1) using the last solar max to gauge the predicted number of radiation safing events; (2) looking at observing time spent in each of the thermally constraining regions to predict number of maneuver only loads; (3) estimating averages for the work required to generate a maneuver only load and rework the observing schedule for new thermal and momentum states; and (4) estimating the work required for eclipse season contingency planning. Much of this time would be during off-hours. So, strictly from a resource stand-point, if implementing SOSA was estimated to take much over 450 hours it would cost more than it gained (assuming the intangible inefficiencies introduced by unplanned, off-hours work are ignored). The work required to fully implement the split load scheme to provide a science only safing action was estimated to be approximately 2900 hours of labor. So, strictly from a resource expenditure standpoint, the project was a non-starter. However, resource expenditure is not the whole story.

Stopping all spacecraft commanding after a radiation safing event also introduces spacecraft risk. Figure 4 summarizes the risk position before and after implementing SOSA. While implementation does introduce a new risk item, it is low level and acceptable. Implementation mitigates the more impacting risk items, providing an improved risk posture after implementation. Item 3, time to recover from radiation shutdown impacts science efficiency, is the most quantifiable change. SOSA would allow faster resumption of science due to reductions in response effort and schedule rework required to safely return to science. Using the past solar max as an example for the expected number of radiation safing events, and assuming 50% of safing event recoveries are ready for uplink one communications support earlier, then approximately 40 hours of science time per year could be saved during the solar max. Considering that in 2010, the year of this analysis, the team running Chandra was approximately 200 full time equivalent heads, one hour of science time could be mapped to approximately 48 hours of labor. This method does not include the development costs for Chandra when assigning value to science time, adding that in would only increase the estimate. Using this methodology, the conservative estimate of 40 hours per year of science time saved equates to 1920 hours of labor. So now the labor savings estimate is 6210 equivalent hours (150 labor + 1920 converted from observing time, for each of three years) over the solar max period, against the development estimate of 2900 hours. Factor in: (1) an improved risk posture; (2) downward pressure on resources and funding, increasing the importance of future labor savings; and (3) tight operating budgets for NASA making efficiency and science return per dollar spent critical to keeping a mission alive, and the trade-off equation flips. The non-starter becomes a top priority.

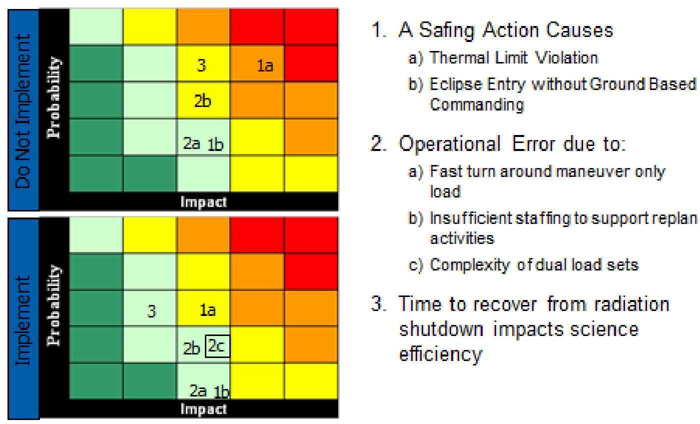

Figure 4. Chandra risk posture with respect to managing radiation safing events before and after implementation of a science only safing action. *An assessment of spacecraft and mission risk before and after implementation was an important factor in deciding to proceed with implementing a science only safing action on Chandra. As the figure shows, implementation was assessed to mitigate more risk than it introduced, thereby improving the overall risk posture.*

In order to build support and fully flesh out the project plan, the design concept and benefits of the project were rolled out in layers. Key decision makers and influencers were briefed early to build support and ensure the design did not contain any "show-stoppers". Once the time was right for implementation (the flight team had been working another multi-year enhancement project, which could not feasibly be run in parallel within existing resources), the science, mission planning, and engineering teams, who would most benefit from the change, were brought in. Questions were answered, and the design tweaked. Next the ground-system development team, who would bear a lot of the work, but not realize as much benefit, was brought in. The ground system observation scheduling routines are complex, difficult to modify, and even more difficult to verify. So the design focused on modifying only the command load generation portion of the software and a requirement not to modify the observation scheduling portion of the mission planning software was agreed upon. Priorities were laid out and a scheme that would allow the software development team to work SOSA and another top priority in parallel was agreed to. Showing that, when broken down into elements, the work load was reasonable removed the last reservation of much of the technical team. So, by the time the project went for management review, it had enthusiastic support from the teams that would be doing much of the work. This buy-in from the team was crucial to executing the project on a reasonable timescale and to getting to the best possible final design.

IV. PROJECT MANAGEMENT

Implementing a science only safing action on Chandra posed several challenges. First, though the concept was simple, it touched all aspects of operations, so several teams would have to collaborate and effectively work together to keep the project on track. Second, the team is sized for maintenance and small upgrades. The day-to-day work of keeping Chandra running safely and efficiently takes up a large, and difficult to predict, fraction of the team's time. Operations must come first, so tasks for implementing SOSA were executed on an "as possible" basis for many of the team members. This makes projecting and sticking to a project timeline very challenging. Finally, to achieve the full benefits, SOSA would need to be in place before the next solar max, so the project timeline was important. The Chandra team employed several proven methods to keep the project on-track, while recognizing that schedule slips were to be expected and needed to be taken into account in planning.

First, a cross-functional working group was established. The Chandra operations community already had several such working groups, which were very successful. A Star Selection and Acquisition Working Group (SS &AWG) was established early in the mission to look at issues related to the star camera, aspect determination, and pointing control. It pulled together members of the flight, science, flight dynamics and ground-software development teams. The SS&AWG still meets today; the group has kept issues of aspect determination and pointing control well coordinated and has made several proactive improvements to keep ahead of future concerns. When the number of conflicting thermal constraints began to over-restrict mission scheduling, impacting the ability to schedule science targets and science time efficiency over-all, the Mission Planning and Constraints Working Group (MPCWG) was established. MPCWG brings together flight team planners, science team planners, subsystem engineers, the science instrument teams, ground software development personnel, and operations management to address emerging constraints and monitor existing ones. Their mission is to maintain the balance between spacecraft safety and mission scheduling capability. The group has been highly successful at achieving that balance and managing a large and complex set of constraints. Many of the people from these two groups, with some additions, formed the SOSA working group. The group met weekly to work through interactions, discuss requirements, report status, and coordinate interconnected activities. Group collaboration identified several improvements on the original concept which were accepted and implemented. One such improvement eliminated the need to modify the science data processing software, removing the longest lead item, and an area of some management concern, from the project schedule. Participation in the weekly meetings by the management team and members of the NASA review panel allowed such improvements to be incorporated smoothly and eased the overhead for status briefings and formal reviews. The working group established a TWiki web, which served to facilitate collaboration, document progress along the way, and provide the material for reviews. The working group and its TWiki web became the driving force behind SOSA.

Second, the team did not try to invent new processes for this project. Each element has its own time-proven change process, and each process was followed as it would be for any other change. The FSW went through patch request and approval, design review, code review, and test results review as usual. The ground software changes went through requirements development, release content approval, pre-release testing, benchmark testing, release testing and deployment review and approval as usual. Changes to load review software went through the load review panel as usual. Changes to tools and procedures all went through their normal development, test, and approval process as well. By allowing each element to proceed with familiar and proven processes, each element was completed efficiently and with the proper level of oversight, test and control.

Third, a layer of system level reviews and tests was added above and beyond the usual processes for each element. These reviews were intended to verify that all interactions had been properly managed and that the system had been tested as a whole. The system wide reviews were as follows:

1) Preliminary Requirements Review
2) Project Management Review
3) Full Requirements Review
4) Test Readiness Review

5) End-to-end Test Results Review
6) Implementation Review

Attendance at the working group by NASA program and engineering representatives helped these reviews go smoothly and removed much of the customary formality. The TWiki that had been used for collaboration and documentation was also used to present material for the reviews, removing the overhead of generating formal presentation packages. Problems were generally worked as they came up so there were no surprises at the reviews. With all of this pre-coordination, the system level reviews may seem unnecessary, but they forced the team to stop and take a system wide look, which was critical to the success of the project.

V. IMPLEMENTATION

A. Flight Software

The flight software changes were made smoothly and with very few changes to the initial design. However, a few improvements were found along the way. The allocation of memory space was tweaked to take best advantage of blocks within memory and to maintain the size of downstream SCS slots. Two extra SCS slots were added to the list of command sequences to be stopped for high radiation. These extra slots would allow the SI teams to run short, special purpose procedures or sequences knowing that if the radiation environment became potentially damaging for the SI, then the sequence would be stopped. The flight software changes required were in different portions of memory, and the SCS definition change was too large to be uplinked in a single file. So the modifications were divided into seven patch files. Flight software ground testing, using the spacecraft simulator, included unit-level tests, tests as each patch file was applied, spacecraft system-wide tests, and regression tests for some of the core spacecraft functions (command and data handling, power management, pointing control, and safing functions).

There was initial concern that putting the modified code into the flight computer would require stopping the flight computer or disabling its safing capabilities. Neither of these situations is desirable, and both would require extra time and contingency planning and add risk to the uplink. Therefore, an uplink plan was developed and tested early. The uplink executed the following steps in order:

1) Put Chandra in a safe configuration, with the SIs stowed
2) Clear the running Mission SCSs
3) Disable the SCS slots that require size or location modifications
4) Disable the small subset of safing SCSs that require modification (with the SIs safed and the attitude selected for uplink, each SCS could be disabled without impacting the safety or redundancy of Chandra)
5) Uplink all patches
6) Dump all patched regions of memory to ensure modifications were applied correctly
7) Re-enable disabled safing SCSs
8) Uplink replacement split load set

The uplink plan required neither stopping the flight computer nor disabling large portions of the safing system. It was successfully tested using the ground-based hardware-in-the-loop simulator used for flight software testing and procedure check-out and was accepted for flight use. The mission loads that covered the uplink timeframe continued on past the uplink time-slot so that if the uplink was scrubbed, or the planned communications pass failed, Chandra would not sit idle. The replacement loads simply picked up in the middle of the planned week. This introduced some bookkeeping challenges with respect to the weekly observing plan, but a similar technique had been used successfully before.

B. Ground Software

The first task to having the ground software split the weekly loads into vehicle and observing sets was to define which commands belong to which set in a simple, flexible, yet robust way. The system that builds the weekly loads uses Absolute Time Sequences (ATSs) to build blocks of commanding for common tasks. For instance, there is an ATS for maneuvers, one for star acquisition, and several for commanding the HRC. The team classified the individual ATSs into one of two categories, observing or vehicle. ATSs such as maneuver, star acquisition, and momentum unload were placed in the vehicle category. ATSs such as HRC high voltage, and Science Instrument Module translation were placed in the observing category. Fortunately, no ATS fell into both categories, but handling for that eventuality was considered (make two copies with different names). Rules were established for calling an ATS from another ATS, and for standalone commands. Each ATS file was edited to add a category flag and the ground software modified to process the flag and place all commands from that ATS into the appropriate command set. Sorting by ATS divided commanding at the functional level, rather than the command level, therefore reducing the processing required for the sort and the overhead in maintaining the category lists, while maintaining the flexibility to use a single command in different categories depending on function.

The initial categorization was revisited several months into the project to take advantage of an identified savings. The unique identity number used for observations (OBSID) was originally planned to go into the vehicle loads. Over the years, people, and therefore the tools they use, had become accustomed to each attitude having a unique OBSID. The OBSID served as a useful tracking and marking tool for attitude based analysis. If it were to stop changing with attitude the assumptions made in these operational tools would no longer hold, so OBSID was categorized with maneuvers into the vehicle load set. However, commanding the OBSID for observations with no associated science data would cause accounting problems for the data processing system and the observation planning system. These problems could be fixed, but doing so brought additional groups into the SOSA timeline and added work. Therefore, OBSID commands were moved to the observing loads and operational tools and expectations with respect to OBSID adapted to handle the new scheme. Adapting the operational tools was not without work itself, but the interactions were fewer, the user base much smaller, and the lead time far shorter, so the change presented an overall savings in effort.

Next, a new verification method had to be developed to ensure the command loads were split correctly. Chandra maintains a set of independent mission load verification tools, collectively called Backstop. A new Backstop check was developed to confirm each command is placed in the correct load set. The check was developed bottom up. Instead of using the ATS down approach used for building the loads, a dictionary of commands previously used in the mission loads was collected by script. The commands in the dictionary were then categorized into vehicle, observing, or both. If a command appears out of its category a warning is raised. If an uncategorized (i.e., never before used in the weekly loads) command is detected a warning is raised. This provides an independent check intended to catch both software errors and the potential human error of an incorrectly categorized SCS or command set. It still allows a command to be used outside its usual category, but that use must be presented before a load review panel and verified as acceptable. It gives the added bonus of warning the user if a command has never been issued in the mission loads before. Over twelve years after launch new commands in the weekly loads should be very rare, and should receive extra scrutiny.

With the categorization method established, the ground-software development team could start working the required changes. There were extensive discussions about naming conventions and what types of reports would be required; all were addressed at the working group level. Maintaining command timing and separation as they were, despite the commands being issued from separate SCS slots, was simple on-board but proved to be a real challenge in terms of reporting and verification. Several iterations of the ground software were required to find and resolve the corner cases. After the initial requirements were accepted and the coding work started, a concern was raised that a simple operational error could allow the observing loads to run without the vehicle loads. Configuring for observations at the incorrect attitude could potentially damage the SIs, so some robust method of preventing such an occurrence was required. The team came up with a solution where the vehicle commands and the observing commands would be built into the same command load, but still sent into different SCS slots. Uplink of a single file would load the vehicle SCS and then the observing SCS. This way, one could not be uplinked without the other. Operational procedures would be modified to require use of scripts that activated first the vehicle load and then the observing load so that the observing load would never be active without the vehicle load running. Implementing this change meant changing the agreed upon requirements, reworking part of the new code, revamping the Backstop code that reverse engineers command loads, and developing new operational scripts, but it was an important change and therefore the added work was accepted as necessary.

While the load building system was under development, load verification tools and processes were also being reworked. The science teams had to rework their verification scripts to expect the separation of commanding and the new SCS slots. The engineering team checks had to be modified to verify that both the combined loads and the vehicle loads alone were safe. Directory structures, naming conventions, and scripts had to be modified to handle two sets of products, combined loads and vehicle-only loads. Finally, load verification checklists had to be audited and updated. Sample products (command loads and the associated reports) were required before any of these changes could be made, but the timeline did not support developing all of the checks after the ground software work was complete. To facilitate working in parallel with ground system modifications, mock products were worked up by hand using the requirements for the ground system changes. The mock products were used to design and perform preliminary testing of the necessary changes. Once the first test products were released, the load review software and processes were ready and only small tweaks were necessary.

VI. TEST

Each piece of the SOSA chain underwent unit-level testing and then subsystem level testing. The ground system changes were further verified by recreating existing mission schedules to test the load generation and review thread. The schedules were carefully selected to cover a wide sample of operational scenarios and events. The categorized ATS list was compared against the composite list of ATSs called by the test schedules. Coverage was assessed to be adequate, but it was not complete. In all, 14 operational schedules were recreated with the new software and reviewed with the new verification processes and tools. Six of the schedules were also used for formal verification of the ground system change requirements.

Only after each component of the full end-to-end thread was verified and the test results documented, reviewed, and accepted, did system end-to-end testing begin. Four end-to-end test cases were laid out and conducted. The first verified the uplink process for the FSW patches. The second case tested three variants of a weekly load, showing that each could be successfully put through the load review process and loaded into the OBC on the spacecraft simulator without generating errors or unexpected changes in the loaded commands. The third case called for running a “nominal weekly load” on the spacecraft simulator. However, the spacecraft simulator was designed for flight software verification, not ground commanding and science load verification. So, some creativity was required to successfully design such a test. Changes from the nominal process intended to help the test case run successfully on the simulator included: (1) reduced observation times to keep the run time reasonable; (2) artificially shortened command loads to force roll overs between the SCS slots; (3) elimination of large maneuvers to minimize the chance of calibration errors in the simulator causing failed star acquisition; (4) modification of star catalogs to work with the simulator’s mini-star-catalog (5) scheduling less than one full orbit of observations; and (6) filling the non-observing time with activities that would be unlikely to be scheduled together, to provide as wide a range of commands as possible. Many of the commands issued by the test load have no associated verifiers on the simulator, so a new tool to “catch” commands on the spacecraft data bus was developed. The tool catches all commands, those issued by the loads and those generated by the OBC (e.g., torque commands to the wheels to maintain attitude). The captured commands therefore had to be analyzed in detail to identify those that were issued from the SCSs and then to verify that each SCS-issued command was issued at the correct time and that no command was missed. Test four repeated test case three, but triggered a high radiation safing action part way through to verify that the observing commanding stopped while vehicle commanding continued. Each of the system level tests was also used as an opportunity for training. Flight team controllers were involved in the execution of the tests whenever possible to maximize their exposure to SOSA before it became an operational reality. All end-to-end tests passed with no adverse findings.

VII. UPLINK AND RESULTS

The Science Only Safing Action uplink plan was successfully executed on December 1, 2011. The command load generation software was run in a parallel configuration for the week leading up to uplink so that loads could be prepared in both the legacy and the new configurations. This parallel scheduling ensured that if the uplink did not occur as planned Chandra could continue with nominal operations. Nominal Configuration Management practices and procedures were used to execute a smooth and coordinated transition of all necessary operational products (e.g. commanding scripts, real-time displays, SOPs, verification software, utilities) shortly after successful uplink was confirmed.

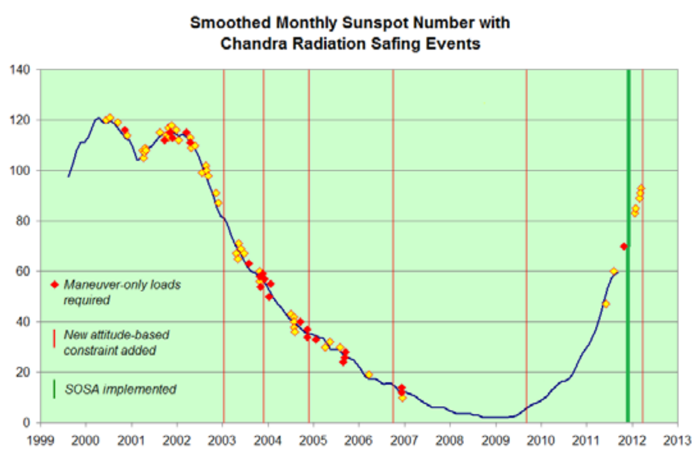

Figure 5. SOSA Implementation Timing and Results. *An update of the plot in Fig. 3 showing data up-to and after the implementation of SOSA. Note that SOSA was implemented before increasing solar activity levels triggered many safing events for Chandra and that no extra loads for vehicle safety have been required since its implementation.*

The first on-board execution of the Science Only Safing Action occurred on January 23, 2012. An elevated radiation environment triggered an autonomous on-board call to the radiation safing SCS. The SCS ran as expected. The science instruments were safed, commanding to the SIs stopped, and commanding from the vehicle loads continued. Had SOSA not been implemented and the vehicle loads stopped, real-time actions would have been needed within 8 hours to manage system angular momentum levels. Subsequent radiation safing events were also successful. Chandra safely transited an extended downtime due to a complex solar storm in March of 2012. Multiple sets of vehicle loads were used to maintain temperature and momentum profiles and to position Chandra for return to science as soon as possible after the radiation environment returned to acceptable levels. Recovery timelines have been very aggressive and recovery from one storm was moved up an additional 8 hours due to the reduced effort to recover with the vehicle loads running.

VIII. CONCLUSION

Operating missions are so often asked to do more with less. With teams sized for maintenance, and stretched at that, implementing changes and enhancements can seem beyond what can be accomplished. However, failing to anticipate downstream challenges and risks will only exacerbate resource shortages and put a mission at risk. So finding ways to implement necessary changes and enhancements shown to provide downstream savings and risk mitigation is an important part of operating a successful long term mission. Chandra has found success in implementing changes such as the Science Only Safing Action by employing several simple techniques.

- Expect timelines to be long and fluid. Trying to enforce the type of cadence necessary for a development program will only frustrate management and undermine the morale of the team. Since day-to-day operations must come first and anomalies happen, upgrade and enhancement tasks will slip. Failing to plan for this can jeopardize the project.
- Make communications between groups as easy and open as possible. The Chandra model is to form a cross-functional working group. This simple technique has proven itself over and over again.
- Involve the people who will do the work early and listen to their inputs. Get their support, make sure they are invested. Again, an enhancement activity is unlikely to be at the very top of their priority list. If they are invested in, or, better yet, excited about, the project the work will get done and done well. If they are not, the project will take far longer and the results will not be as good.
- Keep changes as isolated as possible without compromising on the goals of the project. For SOSA, ground system changes were isolated to command load generation, flight software changes kept outside of core routines, and operational tools were modified to keep data processing untouched. Each of these items was a deliberate design choice aimed at minimizing interactions and test effort.
- Think about uplink (or transition) early in the design. If the great new module cannot be uplinked without significant downtime or spacecraft risk you need to know that very early and consider it in your decision making process.
- Do not reinvent the wheel. Just because a project is larger in scope than most operational changes does not mean that the time proven methods for generating, reviewing, verifying, implementing and controlling the changes become invalid. For system wide changes, a top layer of systems management may be necessary to monitor and verify interactions and end-to-end functionality, but each piece can be most effectively managed using existing and familiar processes.

In all it took approximately two years to implement the changes necessary for a Science Only Safing Action on Chandra. The uplink and transition to operations went smoothly and it has been seen in action on-orbit with no surprises. The benefits are already evident in the operational response necessary and the return to science timelines for recent high radiation events. While the project was in process, day-to-day operations were never compromised and several anomalies were successfully handled. Implementation of SOSA on Chandra clearly shows that a small, but dedicated, team can make proactive, impacting changes well into a mission’s operational life.