SYSTEM INTEGRATION ISSUES IN APOLLO 11

Hugh Blair-Smith, Down to the Metal, Dennis, MA; formerly of C. S. Draper Lab, Cambridge, MA

Abstract

"Houston, Tranquility Base here. The Eagle has landed." Two obscure errors almost prevented these words from being spoken. The errors were not made by the crew of Apollo 11 or by the controllers in Houston, nor were they made during the mission. Rather, they were made by engineers and managers, years before the flight. How they happened, and how they went substantially undetected and effectively ignored, is a pair of lessons in system integration that avionics engineers must never forget.

The Apollo Program is justly famed as a giant leap for the techniques of management of complex system design and implementation. Nonetheless, these tools were used by human beings and so, necessarily, imperfectly. One of the most challenging tasks in any complex system is controlling and testing the interfaces between major components that are developed by different organizations.

Among the management tools deployed by NASA were ICDs (Interface Control Documents). The author has not been able to determine whether this phrase was first coined for the Apollo program or the Mercury and Gemini programs that preceded it, but it was certainly a major tool in Apollo. One of the errors under discussion here was caused by a blatant failure to update an ICD in response to an engineering change, which can be classed as a management error of omission. The other is much subtler, involving a question of how previously unsuspected vulnerabilities (to crew procedures, in this case) should be communicated when they fall outside the scope of an ICD, yet turn out to have relevance to the way the interface is used. This becomes a problem because an ICD is a top-level document limited to specifying the design parameters of one subsystem insofar as they are of concern to one other subsystem. It's not surprising that the symptoms caused by the latter problem have been totally misunderstood by almost everyone from President Nixon on down, and only partially understood even by Buzz Aldrin, who along with Neil Armstrong had to deal with them at the time.

This misunderstanding is so widespread that almost everyone with any acquaintance with the Program Alarms during the Apollo 11 landing believes that the LM's Primary Guidance Navigation System (PGNS) "failed" in some way and had to be rescued by human intervention. That is the exact opposite of the truth, which is that performance margins built into this very robust system quarantined the effects of the errors so that the landing could proceed with the designed level of human involvement, specifically dodging the "field of boulders" that the PGNS could know nothing about.

This paper is largely a retelling of the higher-level parts of a paper, *Tales from the Lunar Module Guidance Computer* by the author's colleague Don Eyles [1], but with the orientation changed from a historical narrative to a cautionary tale with recommendations for modern avionics development management. Results of more recent research by the author and two colleagues are also incorporated.

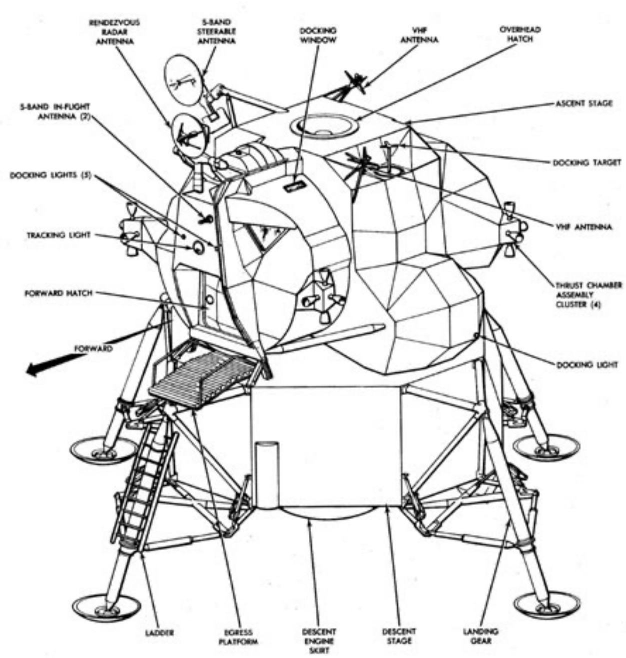

Figure 1. The Lunar Module [1]

LM DPS ICD SNAFU

The Lunar Module (LM) had two main engines, in addition to the Reaction Control System (RCS) hypergolic jets that were used mostly for attitude control. The Descent Propulsion System (DPS) included a gimbaled rocket bell appearing among the LM's legs; it slowed the LM to a zero velocity in the Moon's coordinate frame at the landing site, and so had to be throttlable. The Ascent Propulsion System (APS), an astonishingly tiny engine whose fixed bell occupied a hump in the LM's floor area plus a well in the top of the descent stage, boosted the Ascent Stage into lunar orbit and simply ran full blast until cutoff, with RCS jets vectoring the thrust by controlling the Ascent Stage's attitude. Simplicity had a lot to do with the steady reliability of the APS and of the software controlling it. For instance, there was no point in having logic to decide whether its thrust was building up correctly because there was nothing useful that anyone could do if it wasn't!

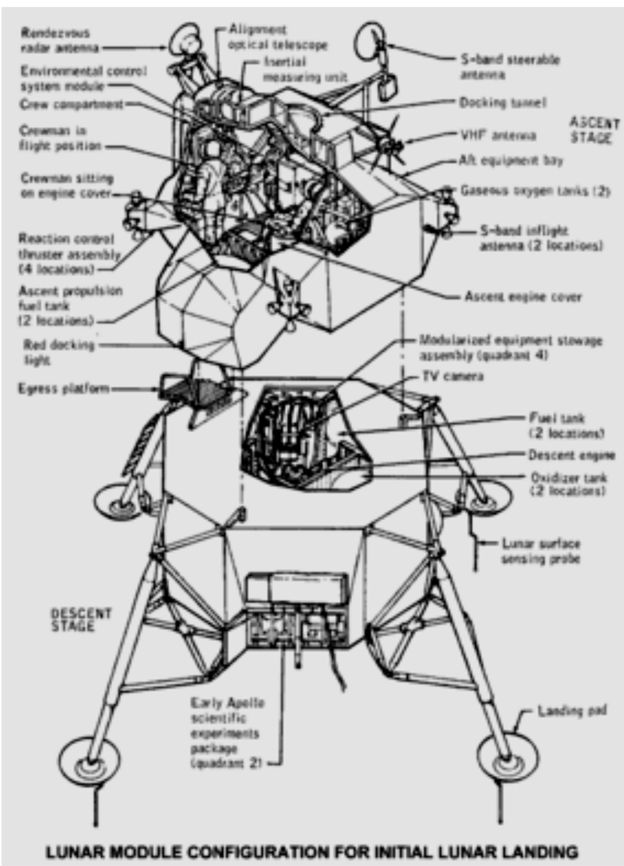

Figure 2. LM Ascent And Descent Stages [2]

So this ICD story is about the DPS and its need for accurate and stable throttling. The challenges for getting this right were daunting. As throttling a rocket engine was almost unprecedented at the time, it should not be surprising to learn that there was a "forbidden range" between 65% thrust and 100% thrust, where turbulence in the combustion would rapidly erode the DPS nozzle at the apex of the bell. Then there was the issue of getting a reliable measure of the actual thrust from the accelerometers in the Inertial Measurement Unit (IMU). The main factor was the considerable distance between the IMU and the center of gravity of that very herky-jerky vehicle, which caused some linear accelerations to look partly like rotations, a phenomenon we called "IMU bob." Then too, in a draconian effort to save weight, the LM's ascent stage was built a lot less rigidly than MIT guidance engineers would have liked, with the result that the IMU was seated on a flexible structure that allowed it to oscillate slightly like a bobble-head doll. We were able to code compensation for the IMU bob, but couldn't do much about the flexing.

Throttle command pulses were sent every 2 sec, each based on the difference between the acceleration requested by the guidance function and the change in velocity measured by the IMU over those 2 sec. To make each throttle command an optimal correction, i.e. to maintain smoothness and stability of the varying thrust, we had to know among other things how promptly the engine responded to control pulses. That knowledge could not be measured, but had to be obtained from the ICD defining the engine's parameters, which specified this "lag" as 0.3 sec.

A Problem Of Interpretation In Development

During software development, the DPS response was modeled by a simulator program that used the 0.3 sec lag time. Don Eyles, responsible for this part, tried a phased approach to finding the right lag compensation, first running the software with no compensation and finding a ringing effect as the thrust overshot its target value and the next command cycle undershot by a smaller amount, and so on. Then he compensated for a lag of 0.1 sec and found the ringing considerably reduced. When he set the compensation at 0.2 sec, the ringing vanished. Had he been less curious, and simply set the compensation at the 0.3 sec value stated in the ICD, he might or might not have seen the other side: thrust oscillations building up into instability. Out of a sense of using no more "medicine" than the situation required, Eyles left the compensation at 0.2 sec. In retrospect, it may be that the lesson being learned was that the compensation value used by the thrust command software should not in any case have been set equal to the lag time, but should instead have been determined as an approximate time constant, about 63% (1 – 1/e) of the lag time.

ICD SNAFU Emerges In Post-Flight Analysis

The telemetry data from Apollo 12 showed a considerable "throttle castellation" phenomenon, so called from its square-wave pattern reminiscent of a battlement. Evidently all that careful tuning of the compensation time had gone for nought! Investigation turned up that the DPS's actual lag time had been significantly improved to 0.075 sec, and *nobody updated the ICD or communicated this change to us in any way*. The castellations were caused by major overcompensation. Curious to see what sort of abyss we'd danced on the edge of, we ran simulations to see what would have happened if we'd insisted on setting the compensation to match the documented lag of 0.3 sec. The answer was, gross instability of the DPS throttling which would have absolutely triggered an abort, and yes, that would have applied also to Apollo 11 and thrown the program into a freeze until the solution was found and tested. Brrrr!

What To Do About It

In our present age of universal connectivity, it's not too hard to see how this sort of blunder can be eliminated. Every ICD, and every other design document of any status, must be made to reside in the Internet "cloud;" every test by OEMs, integrators, and other interested parties must go to the cloud for absolutely current information. Furthermore, every such user must be "red-flag" informed of any changes since the last time they used it.

The issues in the other problem are somewhat trickier.

Program Alarms 1202 and 1201

During the powered descent to the lunar surface in Apollo 11, the LGC [1] issued several program alarms that indicated that the computer was unable to finish all its assigned tasks in time, and was dropping non-essential display tasks. This led to the formation of a myth, believed by nearly everybody, including President Nixon who should have been better briefed, that the LM's Primary Guidance Navigation & Control System had somehow "failed" in a way that required human intervention or "takeover." The truth is exactly the opposite: the PGNS software had been deliberately designed to recover from such unexpected disturbances and persevere with the high-priority tasks that flew the vehicle. All the humans had to do was to notice that flight was proceeding correctly, and forgive the disappearance of some displays. Admittedly, the excitement did distract the crew, Buzz Aldrin more than Neil Armstrong, because Aldrin's role was to watch the instruments while Armstrong watched the Moon. Whether an undistracted Armstrong might have been able to land short of the crater he elected to glide over (thereby landing with more fuel reserve) is open to question [3,4].

So what was this unexpected disturbance? It was a runaway interface between the RR (Rendezvous Radar), not in use during that flight phase, and the LGC, in which it wasted 12-15% of the computer's time with pointless incrementing of two counter registers in the magnetic-core memory. Absent the low-priority display tasks, that was exactly the slack time that had been designed in. The time-wasting phenomenon had been seen in testing, but so unrepeatably that investigation was hampered, and there hadn't been enough going on in the test to provoke the alarms.

By great good fortune, the homestretch of flight control drills preflight had been prompted by a LM-simulator failure event to inject the alarms, without any idea of what might cause the disturbance, and Guidance Officer Steve Bales was sufficiently baffled by them to call an abort. The review of this drill concluded that the abort call was an error because flight was proceeding normally with only

[1] LM Guidance Computer, so named for its "habitat" though identical, except for ROM-resident software, with the Command Module's Apollo Guidance Computer (AGC).

one alarm indication. NASA's Jack Garman made a list of all these previously untested alarms with their appropriate responses. Thus between the two of them, when the real alarms came up, they knew to check the flight parameters and say "Go!"

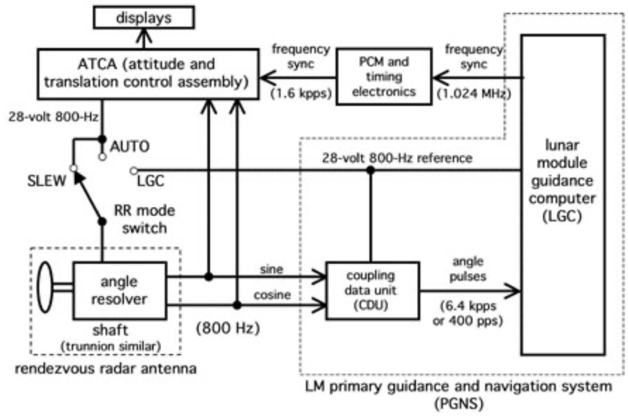

Figure 3. Rendezvous Radar Interfaces [1]

Rendezvous Radar As A Shared Resource

The LM's Rendezvous Radar (RR) was a dish antenna that could be panned by slewing it on a vertical shaft, and tilted by slewing a horizontal trunnion axis. Its normal function was to lock onto the Command Module (CM) during the ascent phase, and feed frequent readings of the range vector between the two spacecraft into the LGC for use by the Guidance function in the PGNS. Three LGC "registers" (actually ordinary core memory locations with special functionality based on their addresses) were used, one for the scalar distance, one for the shaft angle, and one for the trunnion angle.

The less normal function was to provide the same information to the Abort Guidance System (AGS) during an abort, out of the descent phase, that required the PGNS to be shut down. In a spacecraft where weight was at a premium, there could not be a separate backup radar for dedicated use by the AGS.

To provide angle data uniformly and efficiently to two very different computers, both axes were fitted with resolvers producing analog voltages proportional to the sine and cosine of each angle, rather than the relatively straightforward use of digital shaft angle encoders. For PGNS purposes, these voltages were fed into two Electronic Coupling Data Units (CDUs), which interpreted any change in the voltages as an incremental change to the LGC's record of the angles—effectively unwinding the sine-cosine transformation performed by the resolvers.

The resolvers were part of the shared resource, but they got their 800 Hz excitation supplies from only one (at a time) of the two major systems: either the PGNS power supply when feeding the LGC interface, or the general LM power supply when feeding the AGS interface. The two 800 Hz supplies were of quite different voltages, but were specified by the RR ICD to be "synchronized." The appearance of that word in quotation marks is crucial, because different organizations interpreted it in different ways! NASA's System Test Lab was built on the supposition that a single 800 Hz power supply for the two modes was sufficient because "synchronized" means "indistinguishable," right? Wrong, alas. We at MIT had to accept Grumman's desire to keep their power supply as independent of ours as possible, by accepting the interpretation as frequency synchronized [only], not phase synchronized. The fact that the PGNS had to be powered up by a manual action while the LM power was already running meant that a random phase angle difference would be established between the two 800 Hz supplies—though that angle, once thus established, would thereafter remain constant.

Like any other subsystem, the RR's power for most purposes was routed through a pair of circuit breakers which were popped in and out as the crew procedures directed. That, and only that, is what was properly meant by the RR being "on" or "off." Now those quotation marks are important, because signals from the resolvers were not shut off by popping the breakers. The paramount control panel item for our story was the RR mode switch, which had three states: LGC, [manual] SLEW, and AUTO TRACK.[2] The RR angle resolvers were excited by the PGNS 800 Hz supply only in the LGC state, otherwise by the LM supply. That's because the SLEW and AUTO TRACK states were intended primarily for use during an abort, where the AGS might not know the CM's position well enough to slew the dish automatically. Among people with a PGNS orientation, it was not unheard off for the phrase "RR on" to mean the LGC state, and "RR off" to mean the

[2] These state names underwent some confusing revisions during LM development.

SLEW or AUTO TRACK states, regardless of the state of the circuit breakers.

Proximate Cause Of The 12-15% Time Waste

The CDU logic that back-transformed the sine and cosine analog voltages into shaft and trunnion angles naturally depended on those voltages having a proper relationship. That meant they had to obey (at least approximately) the relation $\sin^2 x + \cos^2 x = 1$. Whenever the LGC's current value for either angle was found to be lower than the one implied by the analog voltages, the CDU ordered the LGC to take a PINC (Plus INCrement) memory cycle (11.72 μsec) to increase the LGC angle register value by one, and correspondingly a MINC (Minus INCrement) when the current value is too high. The sampling driving these increments was done at 6400 times per second for each angle, for a combined rate of 12,800 times per second. Usually, a fairly small fraction of those samples would result in a PINC or MINC, but if both angles were being updated at the maximum rate, the time taken per second was 12,800 x 11.72 μsec = 150 msec—exactly 15% of the computer's time [5].

What, then, could cause such a storm of increments? When the RR was in the LGC state, nothing could, at least not for anywhere near a full second. In either of the other states, usually nothing could either, but it turned out there was a rare case when it could. That's where the random phase angle difference between the two 800 Hz supplies comes in. If that phase angle difference was within about 1° of ±90°, the resolver voltages came to the CDUs looking like an assertion that both the sine and the cosine of the angle were close to zero. This impossibility set the two increment trains in motion at (mostly) their maximum rate, searching for an angle that would correspond to the voltages. There was a way to reset the CDU logic to stop this runaway, but none of us had any idea at the time that the problem could arise. Well, almost none of us …

Why Wasn't This Found In Testing?

As stated above, all the simulated hardware in the test labs used a single 800 Hz supply for all the systems in the LM, effectively forcing a zero phase angle difference. Only in the mating of actual flight articles at Kennedy Space Center (KSC) was it possible for the problem to be seen.

What seems very remarkable is that the time-wasting events were seen twice, once in the testing of LM-1, which flew unmanned, and once in testing of LM-3, which flew in Apollo 9; and in those cases the LGC was relatively lightly loaded, so there were no program alarms. It was MIT's Test Operations Director at KSC, George Silver, who spotted it because the time waste actually fouled up the stream of telemetry data. The cause of the time waste in LM-1 was not a phase angle difference, but a side effect of the RR being absent (without proper resistor terminations) from that unmanned flight. Being a veteran of avionics systems based on resolvers, Silver was inspired to look, in the LM-3 incident, for the phase angle difference and compare it against phase angle differences experienced in other test runs. He found that there had been 70-odd test runs of LM-3 (including those that came after his discovery). This empirical probability of one part in 70-something led to our estimating the width of the bad-phase band at approximately 2° or a little over, which multiplied by two bad-phase bands, yields a rectangular-distribution probability of 4 or 5 parts in 360. Silver wrote an Interim Description Report (IDR) describing his findings, and made a very economical design of a hardware change to perform phase synchronization. Grumman, under the gun to add not a single ounce nor insert a single hour into the schedule, could not see a way to incorporate the change.

Were Crew Procedures Partially At Fault?

The state and use of the RR during the descent phase underwent many reconsiderations during system testing. The original idea was simply that it would be unused then, and would come into use only on ascent, or an abort during descent. Then there was the idea that it would track the CM as long as possible during the early portion of descent, while the LM was "face up" but of course it could not "see" the CM during the "face down" portion. For the part we're concerned about, the final disposition was that it should be "ready" for instant use by the AGS in an abort. At a minimum, readiness required that the circuit breakers be on. Maximum readiness called for the mode switch to be set to SLEW so that manual acquisition of the CM could begin with no delay. Buzz Aldrin, the Apollo astronaut who had acquired the greatest scholarly knowledge of spacecraft rendezvous from his studies at the MIT

Instrumentation Lab, was particularly interested in this readiness. Discussing this point with the author in June 2009, he said, "How could MIT not believe that Dr. Rendezvous [a nickname we'd pinned on him in his MIT days] wouldn't want the RR to be on during descent, just in case?" This suggests that he thought the concerns about whether the crew procedures were "at fault" related to the state of the breakers, but we know now that the state of the breakers was irrelevant.

These issues were considered with no thought or knowledge of the mischief that could be caused by a critical phase angle difference between the 800 Hz supplies. As with any other error of omission, the list of people who could have connected the dots but failed to do so is very long. While it's clear enough that the problem would have been avoided if the RR switch had been set to LGC, it's not fair to say that the procedures were "wrong" to specify SLEW. Among other things, a different change to the crew procedures might have been a more robust fix.

Based on a full understanding of the role and dynamics of the phase angle difference, and of how it was seen in the LM-1 and LM-3 testing, there could have been an additional step in the LM power-up procedure. While not exactly a leisurely situation, that part of the flight profile did have room for preliminary checks of the LM systems as they came up. One such check could have been to set the RR switch to SLEW while powering up the PGNS and have Houston scan the telemetry from the freshly powered-up LGC (as George Silver had done with LM-1 and LM-3 pre-flight testing) to determine whether the 12-15% time waste was happening. If it was, CAPCOM would have requested a recycle of PGNS power, almost guaranteeing that the new phase angle difference would be harmless. Then the RR state wouldn't have mattered. The check and recycle could of course be repeated if necessary.

In the event, the change that was eventually made to the procedures was software logic to reset the RR CDUs when the RR switch was set to other than LGC.

Was The ICD At Fault?

It's tempting to say that the RR ICD should have alerted its readers to the probability and hazards of a critical phase angle difference, or at least should have explicitly stated that the synchronization of 800 Hz supplies was for frequency only, and not for phase. The problem with this approach is that an ICD is a top-level document, settled as soon as the design of its subsystem is frozen, and not to be updated except to reflect externally significant changes in that design. The critical phase angle difference issue was not due to a design change; rather, it came about as a late-in-the-game understanding of what the design (occasionally) implied. Certainly this ICD could have been written to underscore the absence of phase synchronization, and if it had done so, the system test labs could have modeled the effects as far as anybody knew them, in which case the time waste could have been seen there first. But nobody knew about the critical phase angle difference and its effects when the ICD was released, so it's not fair to blame the ICD.

Was The Whistle-Blowing Inadequate?

George Silver's IDR went up the proper channels to Houston, and from there to Grumman, and back to Houston and so forth, like so many memos. Silver did express his discomfort with the lack of response, but what more should he, or could he, have done? A Morton Thiokol engineer tried to raise a red flag about the effect of cold weather on the Space Shuttle *Challenger*'s O-rings a few years later with equal frustration. Large projects are littered with the metaphorical bones of the rare individuals who fully understood problems that the "powers that be" didn't want to hear about.

It's tempting to point to any link in those IDR channels and say, "That person should have appreciated the importance of the phase angle differences," but it would amount to a random choice of scapegoat. In view of the fact that Silver's IDR was one ripple in a stream of reports, memos and notes, mostly on matters much less critical, combined with the fact that Silver himself could not then know what all the consequences might be, I'd like to suggest either of two conclusions.

One, which I'd much prefer to avoid if at all possible, is to say that bad things can happen that are nobody's fault in the sense that nobody can give top priority and full understanding to everything they see; we all perform triage on the stream of incoming events and documents in our jobs, and such triage is as capable of error as any other human activity.

The other conclusion is to say that the project management system needs to be improved by introducing a new category of document, or perhaps installing better teeth in an existing one. The obvious candidate here is the IDR that George Silver wrote and distributed through channels, so I'll say that no new document needs to be invented. What's needed is a set of cyberspace "hooks" in such a document that require action from a carefully designated set of technical groups. Notice that such hooks are a significantly stronger management structure than mere links, which make it possible (but not mandatory) to involve the appropriate organizations. As permanent features of the management structure, designed in from the beginning, the effectiveness of hooks does not depend on the influence of the originator's organization, or the originator's forcefulness as a personality, or any other such irrelevancies.

A Cloud-Resident Document With Hooks

Having composed a narrative of what happened, the originator's next task in preparing a properly "hooked" IDR-type document is to research the incident in a way calculated to ensure that all the relevant hooks are set in cyberspace. What this would have meant for the incidents seen in LM-1 and LM-3 is something like this:

1. Identify the fact that the impact is on LGC performance, and hook the IDR to the project management board concerned with performance and scrubs thereof.
2. Identify where in the telemetry data evidence of the phenomenon can be found, and hook the IDR to specifications for telemetry analysis.
3. Identify the role of the critical phase angle difference and find the RR ICD language that hadn't addressed that case, and hook the IDR to the ICD—not to suggest that the ICD was incomplete, but to red-flag every user of that ICD to see the description of the new thing that had been discovered about that interface. The proposal to add the hardware fix to achieve phase synchronization is hooked to both the IDR and the ICD here.
4. Ideally, identify the loophole left open in the system test lab by forcing phase synchronization there, but let's assume the IDR originator didn't know about that loophole.
5. Notice how the PGNS RR CDUs are involved, and hook the IDR to the part of the software specification concerning those CDUs.
6. Identify and document the way in which the control panel setting of the RR mode switch allows or prevents the critical phase angle from wasting the LGC's time, and hook the IDR to crew procedures that involve the RR. (This should be fairly straightforward, based on Silver's identification of this point from old documents, four decades after the fact.)
7. Identify the significance of the fact that the phase angle difference is established at LM PGNS power-up and remains constant thereafter, and hook the IDR to the crew procedure for PGNS power-up—even though the crew isn't focusing on the RR at that time. (This may seem a bit of a stretch, but it needn't be. The way in which the phase angle difference varies or doesn't, and how its value gets established, are as much key facts about it as a person's place and date of birth are key facts about the person. Again, Silver's after-the-fact research turned this up quickly.)
8. Make at least a pro forma proposal for upgrades to all dimensions of testing and hook that to the IDR; in this case it will have scant detail because the possible solutions are so various.

These hooks, once established in cyberspace, will highlight the meaning of the incident for the engineers that can devise fixes or workarounds, and the managers who can assess the problem's importance and coordinate proposed solutions.

Proper Response To The Hooks

The many hooks outlined here for the Apollo case should not be a surprise. In a well-managed project, anything that does slip through the cracks will probably turn out to be something so interdisciplinary that it's hard for any one group to recognize it as theirs to solve. Here's my take on how the various groups could and should have responded to the hooked IDR—had such a thing been available to them at the time.

1. Computer performance board: impressed by the phenomenon of losing 12-15% of the LGC's

time, they maintain pressure on the hooked engineering groups, including meetings, to force a solution.

2. Telemetry analysts: having added the method of detection to their checklists, they hook to the IDR a proposal to bring it to the attention of crews and flight controllers in whatever way those groups think appropriate.
3. RR ICD users (which mostly overlap the other engineering groups): seeing that the problem affects their "baby," they both seek to apply their own specialties and informally press other hooked groups to find a solution. The spacecraft integrator and the RR developer, however, agree that the hardware phase synchronization can't be adopted within their weight constraints and deadline, and they attach that opinion to the change proposal.
4. One group of ICD users, the system test lab engineers, get a real shock out of noticing the discrepancy between the phenomenon and what their model of the 800 Hz supplies do; they propose to modify the lab's hardware to emulate the actual independent phasing, and hook that to the IDR.
5. LGC programmers concerned with RR CDUs: noticing that resetting their CDUs can neutralize the problem, they (naturally) propose that as a good solution by hooking a change request to the IDR, thereby red-flagging their proposal to all the groups involved.
6. Crew procedures planners (including crews): having learned to their surprise how the breaker and switch settings actually affect the problem, seek ways to keep the RR switch in the LGC state during phases of high LGC activity, and may hook a proposal to that effect, or perhaps an opinion that doing so is too great a constraint, to the IDR.
7. However, the same group notices that recycling the LM PGNS power-up (if the time waste can be reliably detected) is a good solution, and may prefer to put forward that solution in concert with the telemetry analysts because of their assurance regarding detection.
8. Hardware/software/procedures testing groups see the IDR and its hooked elements, and hook to the system test lab change proposal their own proposal for a switch in the lab to force one of the critical phase angle differences on demand, to avoid having to repeat tests until it happens randomly.

We see that such an interdisciplinary problem involves a large number of groups, several of whom put forth their own specialties as a good solution. One, the system test lab engineers, can't offer a solution but are highly motivated to make their lab model the reality correctly so as to make testing in their facility more effective.

Resolution Of The Responses

Because of the variety of organizations involved, the computer performance board probably has to buck the resolution up to a broader-scale review board, which nags the organizations whose documents have been hooked until they either propose a solution or state that they can't do so. As proposed solutions come in, that board finds before it a considerable menu to choose from. Let's assume they decide to recommend the recycling of LM PGNS power-up, which they present to a meeting in which all parties whose documents were hooked to the IDR are represented. Let's assume that the meeting settles on the power-recycling approach, and also approves the upgrade of the system test lab, along with the additional switch requested by the testing groups. (This is contrary to the history of the actual solution, which was to expand the LGC software to reset the RR CDUs at appropriate times. I'm just taking advantage of 40 years of hindsight to suggest what I think should have been done.) Here's how the various organizations might feel about it:

1. Computer performance review board: delighted because detecting and eliminating bad phase angle differences means no 12-15% time loss. Having lots of other performance issues to worry about, they're glad to unhook from the IDR.
2. Telemetry analysts: happy to be part of the solution anyway, but also pleased that even a low-probability problem that has been polluting their data stream will be made to go away as soon as it occurs.
3. Spacecraft/RR engineers: relieved to have dodged the hardware-fix bullet, unhook that change proposal from the IDR, also delighted

that their subsystem is not going to be seen as a problem child. They unhook from the IDR.

4. System test lab engineers don't love the amount of extra work they're in for, but it's a relief to close a loophole they'd been letting through.
5. LGC programmers: delighted to not have to dedicate any program space and development effort to band-aiding the problem with CDU resets. They unhook their software change proposal from the IDR, and unhook their organization from the IDR.
6. Crew procedures and flight controllers concerned with the RR: happy to not have to worry about the RR switch, they unhook both their proposed switch rule change and the RR part of the control panel from the IDR.
7. Crew procedures and flight controllers concerned with LM power-up: wary of adding anything to their flight workload, but able to live with a simple repetition of something they are doing anyway. The crew is happy they don't have to think about it until and unless CAPCOM makes them, just like so many other recycle calls
8. Testing groups are happy to have the new switch to force bad phase angle differences in the lab.

There is a constituency that hasn't been discussed: the electrical power system engineers, who don't love the idea of more occasions to recycle a big operation like LM PGNS power-up. They weren't hooked in by the IDR originator because their involvement was created only by the recycle solution. Let's suppose they're mollified by the low probability (1 in 75 or so) and the fact that they were required to make such recycling possible and benign anyway. They hook themselves to the IDR to stay apprised of the testing.

Closing The IDR

After all the unhookings authorized by the review board have been made, the IDR remains hooked to the change proposals for the telemetry analyst procedures, the system test lab design specification, the LM power-up flight controller script, and the LM power-up crew procedures. It also remains hooked to the testing groups and the electrical power system engineers. The IDR is also hooked to a new document specifying the changes, and remains open until a review board signs off on the completion of testing of those changes. When that happens, it is closed and thereby unhooked from everything but the change spec, which is also closed. The importance of the unhookings is to declutter the various groups' cyberspace as much as possible. Anyone who later has to reopen the IDR can easily restore some or all of the hooks.

Cloud-Resident Documents: Summary

Different as these two stories and their "morals" are, they have in common a principle, that any authoritative design document should be available only in a way that assures that every use not only involves the current version but also is forced to notice any changes in content and any new documents that have been hooked to it since the previous use. Furthermore, significant activities like subsystem (and higher) reviews and testing should be structured to be users of such documents.

Making such documents "cloud-resident," i.e., available only from the Internet "cloud," is key to enforcing these principles. Requiring test software to refer directly to cloud-resident documents to make pass/fail decisions is also crucial; since different tests will prefer to use different types of documents as references, the hooking together of documents should be designed to prevent inconsistencies.

Effect On The LM DPS Lag Time Case

When the DPS lag time was reduced from 0.3 sec to 0.075 sec, there must have been a design specification reflecting this change. The proposed rules would ensure that in-house subsystem testing would make pass/fail decisions by direct reference to the cloud-resident specification.

MIT software testing of throttle commands used the ICD as the source of the lag time. The proposed rules would ensure that such data in a cloud-resident ICD would be automatically obtained from the same cloud-resident design specification being used by the vendor's in-house testing.

Effect On The LGC Time-Wasting Case

In this case, the RR ICD couldn't contain enough information (because it wasn't known) to enable testing to detect the critical phase angle differences and their consequences. Since the problem could not cause a change in the RR subsystem's design, there could be no update of the ICD to reflect one. Still, the users of the ICD needed to understand the problem and the (ultimately) chosen solution. Hence the somewhat elaborate-looking scheme to hook the IDR to the ICD in some way that should draw the attention of relevant (or potentially relevant) people, without distracting other people.

The other factor in this case history is how a well-built IDR made its way through all the specified channels without ever quite turning into an action item that somebody had to resolve. The scenario above is an attempt to suggest how the cyberspace hooks could supply the "better teeth" that the IDR needed, by guiding management to assign action items to all the people who can probably help.

None of this is an "automagical" solution to any multi-disciplinary problem. It still depends on the originator of an IDR to know, as George Silver did, who the appropriate players are (most of them anyway) and set hooks accordingly. It also depends on project management people being willing to use the hooks as intended, and not spending their energies on defeating that purpose. All the cloud residencies and hooks can hope to achieve is to prevent anything from being lost, and to prevent anything from being used when obsolete. That's a major advance, but still a tall order to use efficiently.

One could argue that the scheme of hooks I've described would not necessarily have prevented this problem, on the grounds that the IDR was closed based on somebody's judgment that the probability of the bad phase angle difference was too small to worry about. But I think the hooks would have raised more red flags earlier and alerted more teams to think about the problem, by requiring them to issue an overt response.

Parallel Between Clouds And Attics

As a residence for documents, the Internet cloud is similar in some ways to attics: they're handy places to put things, in places that seem logical when you put them there, and they will stay put until you seek them again—if that occasion ever comes. What attics and document-laden databases need is rigorous weeding by somebody whose job is it to get rid of obsolete material, or at least move it to an "off-site" archive. In any scheme like the one above, the standard database problem of referential integrity is complicated by the variety of hooks and their potential to disturb and distract people in whose purviews the hooked items have lost relevance. The design of such a scheme must include a careful specification of how the weeding is to be done, and the operating organization will avoid drowning in documents and hooks only if they keep that weeding funded and managed.

References

[1] Eyles, Don, February 2004, Tales from the Lunar Module Guidance Computer, AAS 04-064, 27th annual Guidance and Control Conference, Breckenridge CO, American Astronautical Society, http://doneyles.com/LM/Tales.html.

[2] Numerous authors and editors (not identified), 2002-2010, Apollo Lunar Module, Wikipedia, http://en.wikipedia.org/wiki/Apollo_Lunar_Module.

[3] Hansen, James R., 2005, First Man: the Life of Neil A. Armstrong, paperback edition, New York NY, Simon & Schuster, chapter 28, esp. pp 464-5.

[4] Mindell, David A., 2008, Digital Apollo: Human and Machine in Spaceflight, Cambridge MA, The MIT Press, chapter 9, esp. pp 222-225.

[5] Cherry, George W., August 1969, Exegesis of the 1201 and 1202 Alarms Which Occurred During the [Mission G] Lunar Landing, memo no. 370-69, Cambridge, MA, MIT Instrumentation Lab, p 6 [available through NASA's JSC History Office].

Acknowledgements

My MIT Instrumentation Laboratory colleagues (1960s-70s) Don Eyles and George Silver have supplied large amounts of historical research based on their private archives of the development of the Apollo Primary Guidance Navigation and Control System. Paul Fjeld of NASA also supplied important recollections. I am also indebted to Buzz Aldrin for his provocative question that inspired our efforts to make a fair assessment of the project management issues, and to propose corresponding enhancements.

I've acknowledged these people's contributions in this way rather than by formal references because the relevant documents (insofar as they even exist

after 40-some years) are not publicly available. What we have is just copies that have happened to survive in my colleagues' file cabinets. In particular, we've been unable to come up with even a private copy of the RR ICD, and have had to guess that the word "synchronized" appears there—though it seems certain that either that word or the substantially equivalent word "locked" must have been used.