ORAL HISTORY 3 TRANSCRIPT

JOHN W. AARON
INTERVIEWED BY KEVIN M. RUSNAK
HOUSTON, TEXAS – 26 JANUARY 2000

RUSNAK: Today is January 26, 2000. This interview with John Aaron is being conducted at the Johnson Space Center for the JSC Oral History Project. The interviewer is Kevin Rusnak, assisted by Carol Butler and Rob Coyle.

Thank you for meeting with us again. We were just discussing Apollo 13, so why don't you tell us what was going on when that happened.

AARON: Yes, you asked me offline, before we started here, about whether or not the *Apollo 13* movie depicted more or less what was my role was on Apollo 13, and I basically said, yes, I thought they did a very good job of capturing the drama and the context and the overall sequence that happened. It's hard to tell a four-day intense story in two hours, so I think they did an excellent job.

The one thing that I thought was interesting, probably departed from the actual way we did it, the way we developed the entry sequence of how to power up the command module and service module in preparation for entry was greatly oversimplified in the movie, because it depicted that we had a very high-fidelity simulator that if you went in and tried an entry sequence, the power profile would just be a high-fidelity byproduct of that. And in those days, simulators were computer speed and memory limited, so mostly what simulators concentrated on was the flying characteristics of the vehicles, because that was the key thing to train a crew.

We got very little fidelity with respect to what kind of power profiles it took to fly what the crew was flying in the simulator.

The way we actually worked that out was totally by hand. I mean, it was graph paper, slide rules, and arithmetic. We didn't even have very many calculators in those days. The way the process worked was, I think you remember that we were pulled together by Gene [Eugene F.] Kranz in one of the back rooms of Mission Control Center right after we came off console, right after we got the command and service module powered down, and then the lunar module guys had to worry about how to take care of the command module for a while. We got the command module team back offline and Gene Kranz was making kind of a pep rally about—I'm going to paraphrase here—"The lunar module, they're under control with the command and service module, it's all turned down, so assuming that they can get us back into the vicinity of the Earth, now we're going to do this, this, and this and this, and power up the command and service module and do an entry. We've got to go lay all that out."

He pretty much laid out kind of an assumption that we'd do a normal power-up in a normal sequence, and I think I was the one who finally raised my hand, so to speak, and said, "Well, Gene, we can't do that. We can't do that." And he said, "Well, why can't we go that?" I said, "We don't have enough power to do that." He very astutely, Gene being Gene, said, "Tell me more, John." So I told him some more, and he said, "I'll tell you what." He says, "We're going to put you in charge of power. Anybody that needs to use power in the spacecraft has to get their sequence cleared with John. We're going to put you in charge of the power." So that made me the power broker, so to speak.

He kind of then turned the floor over to me, and I started interacting with the team and some ideas, because I mentioned I had some ideas. Of course, roars came up from the group about, "Oh, my gosh, John, we can't do that. We've got to have this. We've got to have this and this. You can't do it."

I said, "Well, tell you what. Let me spend a couple hours sketching out some ideas, and if you come back, I'll sketch out some ideas, because I know how much power we've got left to expend, and I've got some ideas about how we could do a sequence." Of course, I didn't really have any detailed ideas.

So myself and Jim Kelly sat down after they left, because they went off to get a cup of coffee or do something for a couple of hours, and we started backwards, you know. We said, "We've got to land, and we'll just start working the problem backwards. From landing to entry interface it takes this long. Let's see. Let's run with this kind of power. And from entry interface to blackout, we'll run at this kind of power level. This is the activation sequence," and so forth. So we just kind of sketched out what we had. In other words, rather than dwell on the mood in the room being "This is impossible," which was kind of the first blush look at the thing, Jim and I sat down and said, "What would it take to make this possible? What would we have to do to make it possible?"

We knew basically enough about not only our own expertise in disciplines, because we were EECOMs, which is electrical and environmental, communications, life support, and instrumentation, that was our direct responsibility in terms of disciplines, we knew those systems in great detail, but we also had learned a lot about the other subsystems, about navigation and guidance, about flight control, propulsion, and those other kind of subsystems that other people in the Mission Control Center had responsibility for.

So we made estimates about what equipment it took to come on when to fly this thing, laid that out in a profile and kind of sketched it out on a blackboard, just in a building-block kind of fashion, and listed under each thing when we were going to turn on what.

The group showed back up, the other disciplines, the guidance officers and the GNC [guidance, navigation, and control] guys and the propulsion guys and all that, and so I said, "Okay, here's how we're going to do this. We've only got this much power, so we're going to start here and we're going to only turn this on. Then we're going to do this. Then we're going to turn this on. Then we're going to do this. Then we're going to turn this on," and so forth. And they went, "Oh, no, you can't do that. We've got to have this, this, and this and this." And I said, "Well, guys, this is all we've got. If you're going to turn that on, then you need to find me something we can turn off." So it started the interactive process.

Once we got the first level of agreements between everybody, with me kind of being the broker, then we said, "Okay, if that's what's going to come on, let's start building the switch list and the circuit breakers and the time lines to do that." We kept working the math all the time with our slide rules and arithmetic.

It then took on the form of finally getting to a first blush view of the switch settings and the crew sequence, and that's the point at which everybody wondered, "Well, that's a very compressed time line. I wonder if the crew can actually do that, actually do all those steps." So that's when Ken [Thomas K.] Mattingly and others started taking our sequence and going to simulators and trying. And then that started another interactive give-and-take process of Ken Mattingly and others coming back from the simulator and saying, "This worked, John. This worked, John. John, this didn't work. You've got to give me ten more minutes here." I said, "Well, if I give you ten minutes at that power level, then I've got to find some way to offset that." So we'd sit there and readjust the sequence and then run the numbers to see if it still met the power profile.

Now, in the movie, they viewed that we had this know-all meter that say when it's this way, you've got some left, and when it's this way, you're drawing too much. That was very much of an oversimplification of what it was. But I thought the *Apollo 13* movie was probably the best space reenactment kind of movie that's been done. It was well done. It caught the context of it.

The one creative thing that I came up with, it was kind of revolutionary in the following sense. Normally, of course, you've got to remember that the command and service module was never designed to be turned off in orbit, so we never built a sequence starting from scratch in orbit, because the vehicles were always turned on days before launch. When you go out and turn a vehicle on to start activating it, you turn it on in the following way. The first thing you turn on, generally, once you punch in a few circuit breakers in the power system, you turn on your visibility systems, which is all your communication transmitters, all your telemetry, all your instrumentation and so forth. So that's the first thing that comes on, because you turn on all the visibility first and then you start activating your subsystems. That means your visibility power is consumed across the whole time frame.

The idea I came up with was, well, the crew is going to be doing this sequence off a checklist. We'll have communications with the LM [lunar module]. One way to make this power profile fit is that we'll ask the crew to turn the vehicle on in the blind. What does that mean? Well, that means they're in doing the large percentage of initial time line by throwing the switches, but not the switches associated with turning on the visibility process. So we left the communications instrumentation systems off.

Then when he got the vehicle almost totally powered up, then I allowed the visibility systems to be turned on as a validation step. That made the team very nervous, first of all, because we not only were powering the vehicle up from scratch, in orbit for the first time, we were going to do it in the blind. But that was one way to make the sequence fit.

When the visibility system came on, I remember that we were pretty close to what I expected the amps and volts were going to read, but they were higher than we anticipated by a couple of amps, if I remember. Of course, then I had to do the quick mental calculation about, "Well, I missed this by 2 amps. I wonder how long that's been on. How much has that caused me to go into deficit?" And it turned out we learned something about the spacecraft we didn't even know, that by punching in a circuit and circuit breaker, we thought it only powered this much, and it actually, due to the way the circuit was actually wired, powered more than that.

So that was the way the power profile and the procedures were actually developed. The procedures just kept taking on more and more and more fidelity, and I think you remember the part in the movie where they said the crew onboard was getting more and more anxious because they kept saying, "I've been looking out the window. The Earth is getting bigger all the time, right? Where's my checklist. We sure would like to see that checklist." And, of course, the ground was saying, "It's coming, it's coming, it's coming." And when the final checklist came in, it was very close to when they needed.

I'll never forget, I ran into the room and I had this checklist, and I said, "Okay, here's the final version," and everybody else in the room says, "Where's the copies? We need some copies." [Laughter] So we had to go run copies so that the CAPCOM, communicator, could read it up and also the other people could follow along. That's how close it went down on the wire and how much empowerment my little team had, was to be able to walk into the room with the vehicle coming at the Earth at 25,000 miles an hour and say, "This is it. Read it up." That's a lot of trust and faith that we all had in each other, because I was not the only person in the room that was doing that. I just happened to be doing it for the power for a while. That was the kind of trust and respect and confidence we had in each other. It took an hour and a half to read that sequence out.

RUSNAK: It was certainly a situation that warranted some unusual steps, but obviously it was successful in getting the crew back, and, as you say, that was the deadline.

AARON: That was the deadline. Of course, if the movie had depicted all that, it would have been a four-day movie.

RUSNAK: Right. It's more interesting to watch the little meter move back and forth than to watch guys doing their slide rules and doing the figures on paper.

AARON: And endless negotiations. The other aspect of Apollo 13 that's interesting was that there were two things I walked away from Apollo 13 that I didn't have the context of during the actual mission because I was working night and day trying to build this sequence. The flight crew, the astronauts, and the team that was flying the lunar module had their own series of problems, and you ought to interview those guys, because there was some very good work and heroism going on there.

I did not have the following sense, that here I was building a very intricate sequence that had to be done perfect the first time and had to be done without visibility. There couldn't be people on the ground in the early part of the sequence watching the crew to make sure they did it right. I did not have a sense until after our mission, the fact that the astronauts were up there in a bitter cold, saturated, damp environment, not getting any sleep, the CO_2 levels were higher than you'd like for them to be, and that they were rationing themselves on water. So here I was assuming perfect astronauts to execute this perfect sequence, and the condition of the astronauts, I felt later, those guys really turned to, because they were not in the best of preconditioning to do that.

The second thing that was interesting about that, you know, we blew the oxygen tank up, blew the whole side of the command module. That vehicle landed. We redesigned that whole vehicle for those subsystems and were back in the air in fourteen months. And we did it right. It was a better spaceship when we got through. But fourteen months and a day's reaction to that kind of a problem is not very long.

RUSNAK: So what did you think of the solution, the fixes that they came up with? Some people suggested that they perhaps over-engineering a fix to it when really it was just a procedural thing that caused the accident in the first place.

AARON: Well, you're right, I think the design was adequate if we had just fixed the problem that we had, the heater wiring problem. The fact that we had a third oxygen tank in a different location is built-in suspenders. It's a better vehicle. Did we have to do it? Probably not. But that's kind of our tradition, is when we fix a problem, we'll over-fix it and make it more robust.

That's kind of the Apollo 13 view. Unless you have some follow-up question, we can probably branch to another topic.

RUSNAK: Had you ever considered anything like this happening to the spacecraft?

AARON: We had never simulated it in a formal sim [simulation] so that the whole team got a chance to interact. Back at the office, a few of us, Jim Kelly being one that I mentioned, had played with the idea of what might happen, how could you do this, is there a way we can get power from the lunar module fed backwards to the command module. There was a lot of "what iffing" going on. There were some "what if" studies that went on. They were not matured, but kind of sketched out. So, no, we had never formally trained to do this.

In fact, I'll give you another anecdote. When the command module had the problem and the decision was finally made after I came in, that it was lost, we had to power it down, that's a traumatic thing. We're going to power down the mother ship. That's a traumatic psychological step. So the team had a hard time coming to terms with that. But when the decision was, "We're going to go live in the LM, start powering the command module down," and so forth, if I remember right, we got that all accomplished in like—I'm going to trust my memory here—in like an hour and a half.

So the next thing is for following missions, we just built this into our repertoire of training, so we really engineered how to do this thing right. If this command module has this failure, we're going to power it down and transition the crew to the lunar module and power it up and shut the hatch.

Do you know that in subsequent simulations on subsequent missions, we never were able to even come close? It would always take us two to three hours to do it. We were never even close to as quick as we did it the first time in real time.

RUSNAK: I guess that adrenaline helps out a little bit.

AARON: And it just shows you that the steps you absolutely have to do is often a shorter list than the steps you'd like to do. It's kind of the bureaucratic tax on engineering. If you've got time, you'll take time. It's that phenomenon. I thought that was an interesting byproduct, because we did formally train on this kind of a failure, I believe for every mission after that, and we never did it in an hour and a half in any of the simulations. Of course, we never got to try to get it to fly, fortunately, but never did it again as fast as we did in real time.

The only other thing that I would add to that is my personal experience of how I got activated on Apollo 13. I was at home at the time, shaving, when that call came, because I was getting ready to come on shift. Now, I need to set up this—I think I've already mentioned that one of my specialties was the telemetry system, instrumentation system, in addition to specializing in these other subsystems, but my home office assignment was to be absolute expert in instrumentation.

The thing that instrumentation, since it's a big multiplexing scheme, where a whole lot of sensors come in to get integrated into a single pipe, then as you neck that down, there's failures that take out patterns of sensors. Like this failure may take out this sensor, this sensor, this sensor. So if you go analyze all the single-point failures in the system, then that helps you sort out the signature of massive failures, because always the problem was, if you're sitting at the console and all the lights come on, the question is, is that a real failure or is that some failure in your monitoring system. And maybe all the lights don't come on. Some subset, some massive subset of lights come on and some massive subset of the parameters jump out to funny readings.

So one of the things I did back when I was in the office was go through the design with my team, and we would pick out the single failures and then script the signature that that would present you with. You know that if this sensor looks funny or is reading, the first question that we implicitly train ourselves is, well, is that a true representation of what's in the spacecraft or is this some instrumentation failure? And the best way to check that would be, well, is that part of some pattern? If it's part of some other five-sensor group that comes through a single junction, then go check these other five. Are they reading good? If they're reading good, it tells you something. It validates more about what's going on. If they're all reading bad, of course, five lights would be on, and that will give you a clue that maybe you don't have five problems, maybe you only have one problem.

Now, I set that up in a long-winded fashion because the people who are on console monitoring systems and they see massive—a large number of lights come on at once, the most probable cause is not that the spacecraft has got a massive failure, but you've had an instrumentation failure. Well, I got to the point that not only did we have all these patterns written down, I had them memorized. Arnie [Arnold D.] Aldrich, who was my supervisor—I don't know whether you've interviewed Arnie Aldrich.

RUSNAK: He's at least on the list.

AARON: You ought to make sure you talk to him on the list, because he'll probably tell you the other half of the story. Of course, he knew one of my specialties was instrumentation, so he called me up and I was shaving. I took the call, and he said, "John, I don't know what's going on out here." He says, "We've got something that's happened that's given us all kind of telemetry indications." And he says, "The team can't really sort out for sure whether it's real failures or instrumentation failures."

So he was on a long-corded phone, and I said, "Well, Arnie, kind of tell me what the subsystems were. Arnie, go to the EECOM console and read me out some parameters. Read me out this parameter. Read me out this parameter. Read me out this parameter." I said, "Go over to the GNC console and look over their shoulder and tell me what this parameter and this parameter and this parameter is reading."

And went back and forth and back and forth, and so I said, "Arnie, I'll be right in, but in the meantime, that is not an instrumentation failure. You tell the guys that that's not an instrumentation failure. There's something really going on there." And, of course, I jumped and ran in.

When I got there, I found that I had an advantage because I'd kind of entered into the room fresh, so I hadn't tunneled into someplace trying to over-concentrate on some faction. I got to see that the EECOM and the GNC and other people in the room were kind of tunneled into their subsystems and I could go across by just walking in the room and listening to what they were having to say over here, listening to what they were having to say over here, and listening to what they were having to say over here.

It kind of gave me the context of what was going on, and this was the point where I then went to the EECOM console and started the process of talking to them about, "You really do have this kind of a failure," and started discussions about it's inevitable that the command module is going to have to go down and you would be better off going ahead and turning it off and conserve the batteries, because the fuel cells were dying. They already had the entry emergency batteries on. Those are supposed to be safe for entry. And so they were burning up their entry reserves trying to troubleshoot the problem. I'm oversimplifying this, but that was the discussion we started to have, was "We've got to turn it off so you'll save the entry power." Well, that's not only a traumatic thing to everyone, including myself, it was a philosophical obstacle as well. But we did manage to get it turned off in time to save enough entry batteries, and we then designed a sequence that would give us an entry with that amount of power.

One of the things we tried to do early in the game was to go broker some power with the lunar module guys, and, of course, they had their hands full, because we had worked out this idea sometime back that if you went through some big intensive circuit breaker switch arrangement, you could feed power backwards. See, there was a power channel that went from the mother ship, the command module and service module, to the lunar module, and it was keeping some housekeeping things alive during coast. The smart guys figured out how you could wire that thing up and have it feed power backwards.

So we went over and started talking to the lunar modules, "Are you going to have any extra power left? We sure would like to recharge our batteries." Of course, they threw us out on our ear. I mean, it was not a friendly response we got. It was like, "We've got enough problems of our own. You guys get lost." [Laughter]

Now, in the end—so I told Jim and others, "Okay, let's not pursue that. Maybe in the end they'll give us some power," and it turned out that way. They managed, within their margins and such, that when we got close to Earth, they did give us a topoff of one of our batteries so we could have that in reserve.

It was a tremendous team, and more than it being a tremendous set of individuals, it was a set of individuals who had worked so intensely with each other that the individuals had that kind of confidence in each other such that there wasn't this requirement to go cross-check, double-check everybody as to whether or not they were giving you good stuff.

RUSNAK: Along that line, aside from the technical, what do you think you learned from Apollo 13? What did the controllers learn in general from that experience?

AARON: Well, the first thing you learned was that the controllers and all that intensive training we did, because we were there to handle the way-out cases, I mean, the first line of defense was always just the procedures. If this happens, throw this switch, do this. We were there for not only help in that process, but we were really there for the way-out, in case the unexpected happened. And we trained a lot on unexpected things, so even though we didn't train on this specific thing, the flight control team were trained on how to handle problems, and it paid off. I think the one thing it brought home was it paid off. The thing that we were trained to do when push comes to shove, we actually can do it, and it was something we had never done before. That's probably the bottom—in terms of being a systems disciplines person, you know, it was the proof of what we were trained to do. I mean, it was the final exam, so to speak.

RUSNAK: Well, you passed quite clearly, I think.

AARON: Yes, and there were other instances where that came into play, not maybe that dramatic in that wide scale. Apollo 12 was an example of that, when we were struck by lightning.

The other spacecraft that we salvaged—I'll use that word—was Skylab. Since the salvage occurred during the unmanned phase of that program, it didn't get the coverage this did, but we launched Skylab and it came unshucked during launch, and that caused us to have to take our four teams that were going to do this long-duration mission, flight controllers, two teams stayed on the console and two teams went off to plan the salvage. I was one of the EECOMs that stayed on the online team to keep the vehicle flying, and we literally flew that thing by the seat of our pants for about four or five weeks there, in a mode that it had never been designed to fly before. So that was a multi-day intensive, creative fly-by-the-seat-of-your-pants problem.

Again, that fell on the EECOM’s lap because one of the things unshucked and came off during launch as far as the meteoroid shield. Well, it turned out that the real purpose of that shield wasn't meteoroid; it was really the thermal shield. Then one of the solar wings came off, a big solar array wing came off, and as soon as we got in orbit we found that if you flew in the mode it was supposed to fly in, which was perpendicular to the sun, the temperature went out of sight. Since the solar panels did not articulate, then they were just fixed and you flew always in a constant reference with respect to sun, that generated the power.

We figured out right quick that if we were going to stay in this perpendicular attitude to the sun, that we were going to melt the spacecraft, almost melt the spacecraft. So we said, "Okay, well, the way you do that is to cut down the heat flux on spacecraft. We've got to change the angle to the sun." Well, you can't change the angle totally, because you can't generate power. So then it became a balancing act between how much power you needed to generate versus how much sun you can stand without completely overheating the spacecraft. So we were having to fly it that way. It was unmanned.

Now, the other complication that got involved in is the fact that this was a low-cost spacecraft, so the only way it knew where it was is if its sun sensor could see the sun. You move one degree, it didn't have an attitude reference anymore except for gyros. Of course, gyros drift. So the EECOMs were balancing this pitch angle with respect to the sun based on solar panel voltages and temperature sensors so we could get the pitch angle by reading voltages, just by looking at our graph paper that said at a certain incident angle we ought to have a certain power level.

Of course, the other dimension was roll, how do you sort roll out. Well, we had some temperature sensors that were next to a shroud so that if you roll off the sun very far, the temperature sensors would change. We got kind of a crude feel for what the roll angle might be.

So we thought we were pretty clever until the following thing happened, because we figured the gyros would drift within spec. There was another problem laying in there, in that the gyros were just drifting randomly. In fact, they were changing signs. So I would request—the guidance officer had a blank commands to the spacecraft to tell it to go to a certain attitude and stay there, and then lo and behold, it wouldn't stay there. It would come back over to another remote telemetry site and it wouldn't be where we put it, and we couldn't figure out what is going on. I say, well, I don't know why it’s off over here, but, guidance, can you get us a command load that will pitch it six more degrees up and I need to go three degrees to the left." And they'd gin it up and send it.

Of course, the guidance officer at one time, we always talked over headsets, and about the third or fourth time—I forget who the guidance officer was. I was negotiating with the flight director and the guidance officer to get these commands, and the guidance officer, because he was in the trench. You know, in the hierarchy of schemes, I was just a poor systems guy. He was in the trench, right? That's where the real power was. Off the air, he walked up to the console and turned around and looked at me, said, "EECOM, do you know what the hell you're doing?" [Laughter]

What happened was that—and of course we wound up taking up a six-pack of gyros in one of the later flights and replacing the gyro package—what happened with the gyro package is that the case deformed during ascent and the heater stuck on, and the gyros were just drifting at a much higher rate than we ever anticipated. In fact, they would even change signs and they'd drift a while in this direction and they'd drift a while in this direction, and that was what was going on, and we didn't have a sun sensor because we were cocked up in attitude. That was just one of the things that was going on when we were flying by the seat of our pants.

Then we wound up, I think you probably know the rest of the story, getting up a sun shield over that, getting a solar wing out, the other solar wing out, and actually had a very successful mission. I wish we still had Skylab up there today. We could be building onto it.

RUSNAK: You're not the only person who's said that, actually.

AARON: But that was a real salvage operation as well. Again, that feeds back to, well, what were fight controllers for and what were they trying to do? And that's another example of earning our money.

Now, where would you like to go next?

RUSNAK: How much time do you have?

AARON: The next thing I'd like to touch on, because I think it can probably use the background, I came from an environment that was very precious, in that I came from an environment that was an incredible team. Then I moved over to the—that kind of ended my ten years of experience with the operations team, and I was very blessed to have been at the right place in the right time to do that.

I then moved over to the development side of the organization because this new thing called Shuttle was in the process of getting built, and that turned out to be a good move for me. A good move for me, a good move for the program, because in that operational setting I had picked up a lot of expertise on a lot of different subsystems, a lot of cross-discipline experience, because the next job I had was to go over and be part of the Orbiter development program, and my particular assignment was to go work in flight software.

The flight software was a separate and distinct project from the prime contractor called Rockwell [International], so we were GFE'ing the flight software—government-furnished equipment—to Rockwell. You can imagine the kind of teaming and team playing you have to have between the group who is building the flight software that's going to fly a vehicle. There's a lot of interaction, particularly when you're designing both of them at the same time.

I was specifically sent over there since I had developed a lot of ground monitoring techniques on subsystems. The intent was originally that the Shuttle would be autonomously monitored on board, so I was sent over there to transpose that technology of called how to monitor vehicles, which we did on the ground prior to that, to plan anything onboard and have the flight software and the flight system on board do the monitoring and diagnostics. That was the reason I was sent over there, because I didn't have any particular experience in software development.

Of course, my job turned out to be much broader than that as we went forward, because it turned into a job that was really about how software—to build the software, you've got to understand what the subsystem guys need. Well, I can go talk their language. In fact, I could talk to them in their language as opposed to talking to them in "softwarese." So that allowed me to be very successful collecting up all the requirements and designing a flight software system, and I picked up the software development expertise and capability along the way, but it was a lot about interacting with subsystems and teams and management. I picked up also a lot of good project management skills there, because it was a hundreds-of-millions-of-dollars project, and I wound up being the head of that as part of the spacecraft software division. So I did that from like '73 to '84.

That was a very challenging, very challenging software development, because remember back in the seventies Shuttle, for the first time, was a vehicle that was going to be totally digital fly-by-wire. There had been test programs that did that, but this was going to be an operational vehicle that was going to be fly-by-wire, which means it was going to be totally under software control. All the flying and critical sequencing was going to be done by software. So that brought software into the centroid of vehicle control, whereas previously on Skylab and Apollo and so forth, software and the guidance system were kind of subsystem. The sequencing on the vehicle and the utilities and all the subsystems was done with hard-wired switches and hard-wired logic, and then software was just the GNC system.

So Shuttle was going to take the software from the subsystem role to the system management role, the vehicle management role. That digital fly-by-wire, making it the centroid, and the fact that we were going to manage the redundancy of the vehicle avionics was software.

Contrast to that, if you go back and look at the Apollo command and service module, we had a digital flight control system—that was the primary—and we had a completely analog backup system. It was degraded kind of backup, so you had unlike redundancy. You had a very good primary system, but the backup system was not built on that technology in software at all; it was a degraded analog system.

The plan was that we were going to take five computers and strings of equipment and managing the redundancy and the failures which were going to be done in the software, so that the challenge became how do you build software that runs in five computers and can tell when computers or strings of avionics fail and vote them out. That had never really been done before at that level, and we had to do it error-free because the one thing you could not stand in an ascent entry is for the software to have an error such that it quit. I mean, you can't stop in ascent or entry. And even though you have five computers, they're all executing the same software, so you've got to have error-free software. So you not only have to design this thing that basically with these computers and their software rendezvous at 500 points a second and agree they're all at the same point and agree to move on to the next point, which they check to see where each other are and where you've been, and if you're not there, how long should you wait before you go without them? And if you go on without them, you've got to take them out of the set.

That turned out to be a very important challenge, and we looked at various architectural ways to do it. There's basically two ways you could do it. You could have multiple computers solving the guidance and flight control equations independently, but what you'll find there is if you're off due to sensor errors, individual sensor errors, then there are biases and drift will set up in those systems so then you have to worry about, well, how do you cross-trip them and equalize them. Well, as soon as you open a cross-trip and equalize, then you have to ask the question, how far can I let this drift? Pollution could be an issue. You could pollute each other.

The architecture that was picked was based on the theory that says if all computers read all the sensors and that given bit by bit identical inputs by computers executing the same software in the same sequence, that they will get bit by bit identical outputs, so that was the premise. That was the fundamental design premise. It took a lot to make that true and always hold up that step.

It was a very intense activity, because we were literally trying to design this and build it in parallel with the vehicle, so we literally had to deal with how to deal with change, because every month the vehicle guys would say—particularly the flight control, because flight control and guidance people have their own set of problems. They were going to run this vehicle from mach whatever, you know, 100, all the way down to landing. And they were going to go through a lot of unknown regions for which there was no full-scale vehicle test data. So the flight control and guidance people, they were all having their own set of problems like so. They started numbering their flight control versions, you know, and so we'd be going off to build flight software and the word would come, "Well, forget Version 2, John. We're going to send you Version 3." They had a very complex problem and we had a very complex problem because we were trying to implement their algorithms to fly their vehicle and also all those other things that I mentioned.

The one thing that we learned how to do is make change management an art, because our first reaction was, "Don't tell us that. Let us finish what we're doing." It was like, "Let us finish what we're doing before you incorporate the next set of changes." Well, we soon figured out that was not going to work. We had to understand a way and a methodology to implement changes as we go. Takes a very disciplined process to do that.

We got to the point where we peaked at about 1,000 changes a year, so the question was how to do high-integrity software development, basically stay on schedule and support the program, and also absorb 1,000 changes a year coming from the program. The interesting thing about that is that we not only developed a good methodology, which is some of the foundations of what critical software developers around the country use today, we also wound up, again, in my viewpoint, with one of the best teams, probably the best avionics team of government, IBM, and Rockwell and all the [MIT] Draper labs and so forth across the country, probably one of the best teams I ever worked with.

Now, that's in contrast with how we got started, and I don't think I told you the starting point of this story that made it very complicated. When Rockwell first won the Orbiter contract to build the Orbiter, within that scope of the Orbiter was the vehicle and the software. NASA management then went through the decision process and made a decision to extract flight software from the Rockwell contract. Well, you can imagine, from a business standpoint, that's not the way to influence people and make friends, because Rockwell, rightly so, was concerned about, "Well, how am I ever going to get this government agency to build the right software to fly this vehicle I need to design and sell to the government?" It started off as very hostile relationship.

RUSNAK: Do you know why NASA did that?

AARON: I can speculate why NASA did it that way. First of all, you need to remember in that time frame we had guys like Chris [Christopher C.] Kraft, Bill [Howard W.] Tindall [Jr.], George [M.] Low, and others, particularly the first two individuals I mentioned, who grew up in the mission control data systems and analysis directorate side of the JSC organization, so they knew not only what it took to develop flight software, but they knew what it was about. So they were very tuned into the fact that the flight software and how you do it is a very strategic aspect of a vehicle.

That was not a common understanding at that time, as I mentioned before. Usually software on previous vehicles was kind of a byproduct of this group over here that wore sandals and tennis shoes and built the GNC subsystem, right? Here was the senior management of the center, who had an appreciation for what flight software is.

So now I'm going to speculate. My speculation is that Shuttle was a general-purpose vehicle that was meant to go a lot of places, and it was meant to be turned around so we could go from one mission to another mission. I think they were fortuitous enough to understand that that flexibility and that reconfiguration was largely going to fall into the role of being implemented in software. It was the software that was going to give you that variability. And if you're not careful and you build software the cheapest way, then it won't necessarily have an architecture that allows you to reconfigure it with high integrity.

So I believe they were farsighted enough to understand that the variability and flexibility of what it would take to go from mission to mission, and maybe missions that they didn't even understand at that time, would probably be vested in the software. So the government wanted a much more hands-on role in how the flight software design came out. That's my speculation.

Of course, that speculation is somewhat derived from the fact that I share that view, of course. Not of course, but I share that view, and it's turned out to be the case. As we went through even the development cycle of the flight software, more and more variability, capability that came along actually was allocated to software to be built in. And then also the government used that when idiosyncrasies in the hardware showed up. Rather than having to go back and go through the prime contractor to get the hardware changed, in many cases if we could adjust and adapt to those idiosyncrasies in the software, I would get the call.

So I think when you look at it overall, the flight software, the scale of it and the complexity of it and also the limitations of computers that we had in those days, you would have to say was a very successful people, very successful project, and we were able to stay on track with the program and therefore not cause it delays. It's not unusual. It's kind of the mold, you know, that hardware gets ready and the program manager always assumes he has to sit and wait on software. Well, in the Shuttle program, we didn't, in the big scheme of things, wait on the software.

I spent ten years doing that, so that was not only an opportunity to use my expertise, but also it gave me a way of picking up development skills, management skills, and it's very important these days to be equally versed in the background of hardware and software. It's kind of unusual to cross that boundary, but it should be crossed more. People should try to cross it more. I found out the same principles apply. If you're a disciplined manager, you manage software the same way you manage hardware. There's no mystique about it. So there shouldn't be reservations about people crossing that line, because if you cross the line, then you're much better to deal with the systems integration of large-scale projects at that next level up, because all projects these days include elements of hardware and software, but it takes two of those to make a system.

That's probably enough on that. Let's see. We were going to talk about something else. I don't know whether that was beneficial or not.

RUSNAK: I think it was.

AARON: It was a very rewarding program, because in the end we started off with this hostile relationship with our prime contractor. We were very much interdependent on each other for success and, through time, became really zipped together to where we were a very crack team and got to that point that I described on Apollo 13, where we had trust and respect and confidence in each other almost as individuals in an independent organization. That's when you know you've got it.

Let's see. We were going to talk about some other things. I then went from there to Space Station. One of the things I'd like to mention about Space Station—we could talk for days about Space Station since I worked on it so long. I've been working on Space Station since '84, except for a two-year time-out on Mars exploration. That's sixteen minus two, that's fourteen years. I've spent almost half my career on Space Station.

When we started Space Station, one of the things that was an interesting piece of Space Station is that the agency, kind of independent of what the Space Station was, had already decided what it was going to cost. Eight billion dollars. Eight billion dollars in 1984 dollars, so it was in constant year 1984 dollars, it was 8 billion. The 8 billion turned out to be almost like the pie profile thing. You're right. It was the fundamental constraint. The only difference is that it took longer to add up the numbers. [Laughter] The iteration process was much slower.

I came on the program in '84, and that's when the formal management structure was set up. I was the deputy program manager to Neil [B.] Hutchinson. The concept development group, they'd been very studied on Space Station. In fact, the most concentrated one was in '83, the year before. I think you know, probably, because it's been studied on Space Station as long as there's been a NASA. But in '83 there was a concept development group that worked on

Space Station configurations, and in addition to wrestling with all the things they wrestled with, one of the first things they wrestled with was the 8 billion. The search for configuration, they would make people happy and still cost less than 8 billion.

Within that year, they went through two major iterations of where you put all the requirements together, then do your preliminary concept layout, then you get this NASA cost model I talked to you about last time, that we were all captured by, and then you run out the numbers. You know, you have to do a large amount of work to get a configuration, and then you run it through the numbers and you say, "Oh, my gosh. It's not 8 billion." And then you start talking about, "How can I squeeze it out?"

The 8 billion was challenged even in '93. This thing naturally doesn't want to cost 8 billion. It just naturally wants to cost 10 or 12 or 14. The agency management at the time says, "No, 8 billion." So it became a question about nobody exactly knew on the team, and even the management structure that I was in, exactly what the origin of this 8 billion was.

So finally the concept development group, in order to declare success, came up with this configuration and they chopped enough out of it that they came in under 8 billion. So Neil and I then put together a skunk works here and started working, again, configuration, put the next little detail on it, and working out some of the idiosyncrasies that the CDD development group had done in '83. And lo and behold, we went off to a budget review with the administrator and the comptroller in '85, I guess it was—maybe it was '84—and our costs had come in at 12. Neil named me the scrub mother. That was what I was basically known as. And I got it down to 10.6. So we took our stuff to the administrator. Of course, I think he knew at the time, he and the comptroller, that we were not at 8. So it took us a while to get him to agree to have a meeting with us, and I don't know what the background of that was, whether or not that was overt or whether or not it was a tough schedule.

We finally wound up in his office kind of in an ad hoc meeting, and Neil went through his blurb about the fact that to have a space station, given the requirements that everyone seemed to want as a minimum, it wasn't going to cost 8; it was really a 10.6. And we talked around that for a while and talked around that for a while, and finally [NASA Administrator] James [M.] Beggs looked over at Tommy Campbell, who was the comptroller at the time, and said, "Tommy, what do you think?" And he said, "Well, you know, this thing really needs to be 8. Remember we've got a projection and we've got an agreement with OMB [Office of Management and Budget] and the Congress and the White House based on a wedge that we laid out, that the sum of that wedge is going to be about 8 billion dollars between now and 1992. So we need to stay within that."

So I took away from that discussion that there had been a fundamental agreement to build a Space Station, with Congress and the administration, and that they laid out some budget projections about what the NASA budget was going to be, and then shook hands about what the NASA budget could grow by, and I think it was 2 percent, was the number that they had agreed to. So that left a wedge, a pie-shaped wedge, with a sum of 8 billion dollars between then and 1992.

Because I think I told you previously that the original plan was to launch it in '92 on the 500th anniversary of Columbus.

So to recap, from what I constructed from the obstacle that we were up against, 8 billion dollars, is that that was the funding wedge that had been agreed to between the administration, the Congress, and NASA, that said if you add up the slice of the wedge that we can fit in between '85 or '84 and '92, which is when we were supposed to launch, the 500th anniversary of Columbus' discovery of America, that that was the sum of 8 billion dollars.

Now, the interesting thing about that is that there's nothing wrong with that approach either technically or management-wise, in theory. Space Station is not like the Orbiter in that it has to navigate the atmosphere, because you can't build half an Orbiter. It won't fly. But Space Station is something since it flies in space. There's nothing that really drives the mole line or drives the fundamental scale of that, and there's no reason it couldn't be modular such that 8 billion dollars could have been the right first step. So it's not that by this little sequence am I casting dispersions on the fact that there's something wrong with that. I mean, the agency was ready to take the next logical step. The next logical step could always be about, well, how big a chunk do I bite off? And you can kind of bite by the yard. So 8 billion dollars was not a ridiculous number with respect to that.

So if you changed your way of doing the vehicle scaling and change your engineering process, no reason you can't build it in a design-to-cost environment. But given that that's what you're going to do, then you have to make cost a fundamental driving parameter, not a byproduct of what you'd like to have.

We never, in Space Station, learned to deal with the following problem, and that is how to manage the requirements as they came in and keep them scaled consistent with what the cost is. And that's not unusual. A lot of programs, as you go back not only in NASA, but in other places, have failures because of lack of requirements management. If you don't get requirements under control, understand what new requirements are going to cost you early enough, you can get so far into that, that you never recover.

If we had been able to manage the requirements—and that's not only a NASA statement, you know. There was also a lot of debate about Space Station. Just to answer the criticisms of what Space Station was in this 8 billion dollar view, the tendency would be, well, this particular discipline scientifically or whatever wants this function, so to get their support for Space Station, the temptation is, "We'll add it in."

Given the right kind of requirements control and given a true design-to-cost engineering cycle, you make cost an engineering parameter just like weight and power. It can be done. We talked about that earlier. Then the debate about the 8-billion-dollar Space Station, with the proper parameter around it management-wise, could be done, but it would have probably been an okay step. But you can't say, "I'm going to satisfy everybody's needs and we'll do what everybody wants technically. I'm going to distribute it across the agency the way the agency would like to do it, the way the various centers like to do it. I'm also going to use it as a way of reinvigorating the agency, keep adding all those objectives onto it." It will cost more than 8 billion dollars.

And that was a constant struggle we had from '85 through, for sure, '93 when I left the program management function, is that we went through these cycles where requirements would grow and about nine months or a year would say, "Oh, my God," because the iteration would cause us to be able to account for that cost. "That's more than we can afford. Redesign it. Take it down." And that was even driven by it costing too much or Congress cutting it, so we spent too many cases of doing this to design it and then doing this to redesign it and then realizing that was too much and so forth.

You've got to have fundamentally stable requirements and fundamentally stable funding profiles to do one of these major programs, either that or have programs cut up into smaller increments so that you can accomplish the objective of getting your arms around the requirements and delivering a piece and then move on to the next piece. Space Station should be able to do that. It should be able to do it modular. In fact, we are doing modular today. But it's been a constant battle of fighting requirements and cost.

Okay?

RUSNAK: Okay. [Tape interruption]

AARON: ...out of the Phase B. The thing that drove that configuration—I don't have a model of the skunk works, but it was a power tower configuration. The thing that drove this configuration was that the modules moved up to the center because microgravity sciences became very much in vogue as an initiative, and so materials processing needs very little microgravity, so you need to be at the center of mass to accommodate that. That then makes it hard to get to for the transportation system.

The power tower previously had the modules on the bottom and the power and truss at the other end that allowed the transportation vehicles to come and go. This configuration over here was what the reduced-cost version of that was. This was broken into Phase 1 and Phase 2 before we started the [Phase] C/D design contracts, because one of the ways we got from—as we cut this thing down to get started, we took the upper and lower keel off and basically ran this configuration. There's a lot of redesigns that went on that's not apparent at that scale. So that's how the Freedom configuration wound up, and you can see the similarity between it and the dual keel.

We went from that configuration in Freedom to the—I don't have it—model of the International Space Station, but if you look at that you would see these elements that have been rearranged and then, of course, the major addition would be the Russian contribution to the program and its modules and propulsion stages and the power systems that are being brought.

In the middle is the Option C, and I need to dig out a picture. This was, as I mentioned earlier, the single-launch core station. We didn't launch it modular. It went up all at once. It was based on a large cylinder that had a lot of volume in it, and it was elevated in the floors. The way it was launched, if we can jump from there to here—this is a cartoon of all the people who worked on Option C—the way it was launched is that this section here was the Space Station. The rest of these were Shuttle components—the main engines and truss structure, external tank, and SRBs [solid rocket boosters]. This not only got you a Space Station in one launch, but it also had a byproduct of getting the agency an unmanned cargo vehicle or unmanned launch vehicle as a byproduct, because you could visualize this could be cargo as well.

This was Option C, and I think we mentioned earlier that the other two competing design concepts in 1993 was Option A and Option B. Option B was basically a refinement of the Freedom configuration, and Option A was a Freedom set of parts, but they did not articulate. It did not fly around with the solar panels articulating to look at the sun. It basically flew in a mode where the vehicle itself pointed toward the sun. Now, that's since been redesigned back to be articulating, which the international configuration is right now.

Let's see. Those are the basic major configuration changes that evolved across Space Station life. Of course, there were many redesigns and reconfigurations below the level of what you could see from the five-mile point.

RUSNAK: Okay.

[End of interview]