

In the beginning, there were flight controllers. These were a strange breed. In the early days of the [US Manned Space Program](#), most american households, regardless of class or race, knew the names of the astronauts. John Glen, Alan Shepard, Neil Armstrong. The manned space program was a unifying force of national pride.

But no-one knew the names of the anonymous men and later, women, who got the astronauts to orbit, to the moon, and most importantly, got them back to earth. The Apollo 13 mission changed all that, not because it was successful, but because it was a successful **failure**; no-one died.



It was immortalized in a [movie by the same name](#). Flight director Gene Kranz (played by Ed Harris) became known as the *Steely Eyed Missile Man* who delivered the compelling phrase that goes down in history. It is etched in every incident commander's mind: "Failure is not an option."

But as unlucky as Apollo 13 was, the second moon mission, right before Apollo 13, almost didn't make it into orbit. This is the story of the first 10 minutes of that mission, Apollo 12. Now you might be wondering, what does all this vintage space lore have to do with site reliability engineering in the 21st century? Well, I invite you to read on and find out.

The story of Apollo 12

Apollo 12 was the second lunar mission. This mission, which launched 4 months after Apollo 11, would have a more scientific focus. The crew consisted of Commander Pete Conrad, Command Module Pilot Richard Gordon, and Lunar Module Pilot Alan Bean. The Flight Director who would command Mission Control was Gerry Griffin. This was his first time in that role, promoted up from flight controller.

The weather at Cape Kennedy on the morning of November 14, 1969 was completely overcast with frequent rain. There was a NASA rule not to launch into any cumulonimbus clouds. While there were some cumulonimbus in the area, there was some debate as to whether the rule actually applied, so it was eventually waived. The countdown went off perfectly and Apollo 12 lifted off at 11:22:00, right at the start of the three-hour launch window.

Unbeknown to the astronauts or Mission Control at the time, **lightning struck the rocket**, not once but twice. First at 37 seconds after liftoff and again at 52 seconds. The first strike caused the protective circuits on the fuel cells in the Service Module to trip, taking all three fuel cells offline along with most of the Command Module instrumentation. The second strike, 15 seconds later knocked out the "8-ball", the attitude indicator and scrambled the telemetry stream back to Mission Control. **The crew and Mission Control were flying blind.**

The Saturn V continued to fly normally, however, because Dr. Werner von Braun, the rocket's chief architect, didn't trust humans with the task of getting one of his rockets into orbit. He insisted on a guidance system for the Saturn V separate from any control in the CSM (Command Service Module) where the astronauts resided during takeoff and re-entry. This decision probably saved the mission and perhaps the astronauts' lives.



No one at the time knew what happened, but they knew it was bad. If they couldn't find the problem and fix it in a hurry, they would have to perform a launch abort which meant engaging the LES (Launch Escape

System) at the top of the command module. This would pull the Command Module (CM) away from the rest of the Saturn V before mission control would blow up the rest of the rocket. This was not just demoralizing, but highly dangerous to the crew and the surrounding Florida coastline.

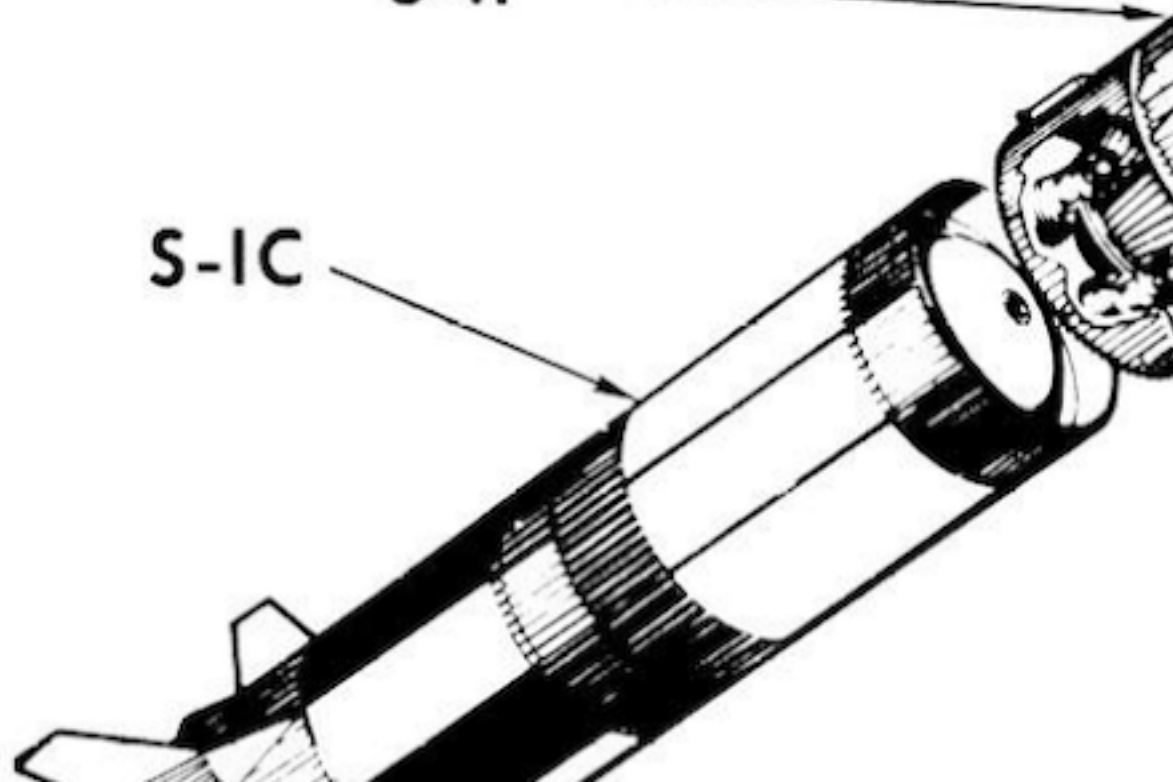
LEA

INSTRUMENT UNIT

S-IVB

S-II

S-IC



This is where a 24-year-old Flight Controller John Aaron comes in. Aaron was not just a flight controller, he was an EECOM (electrical, environmental and consumables manager). EECOMs were flight controllers who exhibited a high degree of professional behavior, were knowledgeable almost to a fault, and had a “take ownership” mentality. They went through a steep learning curve to earn that role. Everyone was expecting that Aaron would recommend a launch abort, but he didn’t. He sent the directive to Flight Director Gerry Griffin, “[Try SCE to AUX](#)” (auxiliary).

No one knew what this meant initially, but Lunar Module Pilot Al Bean did. He remembered it was a switch over his left shoulder. He flipped it and all the telemetry came back on. Apollo 12 continued to orbit without further incident.

“Set SCE to AUX” seemed to come out of nowhere, but it wasn’t a guess. A gamble perhaps, but a gamble based on years of training, ability, and most importantly, pattern recognition. Before we get into the comparison of flight controllers to SREs, let’s learn a bit about flight controllers through the life of one particular flight controller on shift during the Apollo 12 launch, John Aaron.



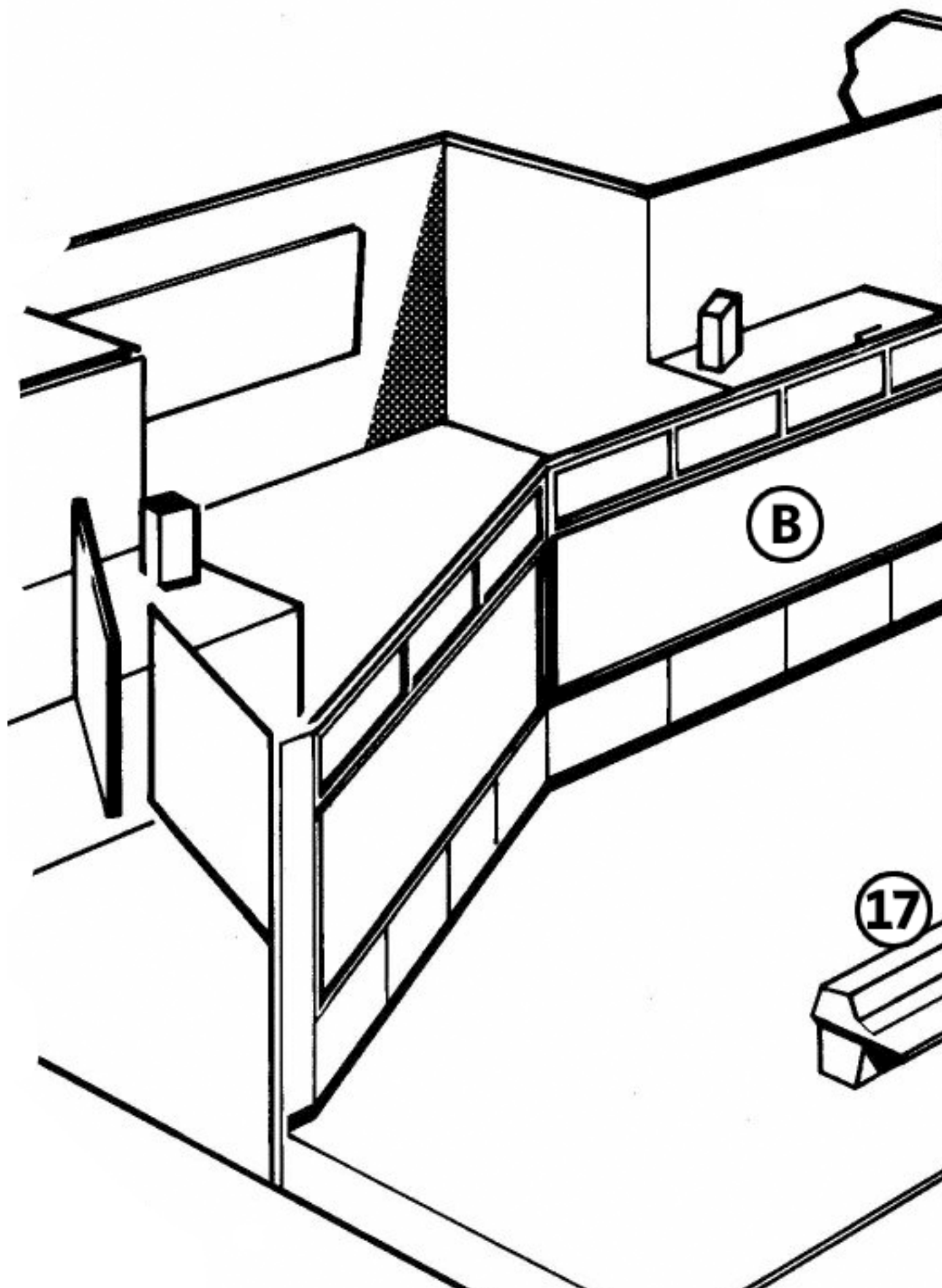
John Aaron was a graduate of Southwestern Oklahoma State University. “My parents expected that we all would go to college and the family was committed to finding a way to make it happen,” Aaron said. “Had that not been the environment I was raised in, I likely would not have gone.”

He graduated with a double degree in Physics and Mathematics and when NASA came recruiting, he signed up for a job in the Flight Operations Directorate at the Manned Mission Center in Houston. He thought that he'd earn a little money, help out his country, and eventually make it back to Oklahoma to continue farming. But, as Aaron said, "Exploring outer space became my new passion and I never made it back to ranching."

A good candidate for a flight controller is a generalist with a specialization in a particular science. On the surface, a flight controller works in front of a console viewing, processing, and monitoring telemetry data coming from the various systems of a space vehicle in real-time.

This information, in the old days, would come rushing by on a CRT (Cathode Ray Tube). The flight controllers also wore headsets where they were monitoring 12 audio loops simultaneously. This clearly was a condition of information overload. But as Marshall McLuhan wrote, "Information overload equals pattern recognition."

A seasoned flight controller adapted to taking all of this visual and audio input and discerning patterns. They were adept at interpreting these patterns and transforming them into advice and recommendations for the flight director, who then makes the decisions. Decisions are data-driven, but *data is "pruned" by experience and judgment.*



There was a station that was manned by the “best of the best” and that was EECOM. The role required you to be able to process your “information overload” and to see “the forest AND the trees.”

Aaron’s training and the ability to see these patterns allowed him to make the “Set SCE to AUX” call that he did on that November morning. How did he know to try this and not just signal an abort? It came back to the process of relentlessly simulating and drilling for what could go wrong.

About a year before, he had seen that very same pattern while helping out a flight controller team in a simulation. The team had managed to disable main power to the CSM Signal Conditioning Equipment, which was not part of the initial simulation. In any event, the strange telemetry readout was, as he said “etched in his brain.” He was curious about what could be done to recover from this. It wasn’t his drill, but his natural curiosity got the better of him. He determined that setting the Signal Conditioning Equipment switch to enable backup power restored power to the telemetry systems of the simulation.

The switch was pretty obscure, just another switch among the hundreds of switches in the Command Module. No one in Mission Control really knew where it was, and Flight Controllers pride themselves on knowing everything. Not even the Astronaut Mission Commander Pete Conrad knew where it was, but fortunately Alan Bean, the Lunar Module Pilot remembered it. He flipped it and the telemetry came back. They were no longer flying blind. Once in orbit, all the systems were checked out and it was determined that there was no serious damage to the spacecraft.

John Aaron and his team took exceptional care to prepare for this mission. This deep calling and pride in being a flight controller is exemplified by the plaque on the door of Mission Control.

The Flight Controller’s Creed

1. To instill within ourselves these qualities essential to professional excellence:
 - DISCIPLINE — Being able to follow as well as to lead, knowing that we must master ourselves before we can

master our task.

- COMPETENCE — There being no substitute for total preparation and complete dedication, for space will not tolerate the careless or indifferent.
 - CONFIDENCE — Believing in ourselves as well as others, knowing that we must master fear and hesitation before we can succeed.
 - RESPONSIBILITY — Realizing that it cannot be shifted to others, for it belongs to each of us; we must answer for what we do—or fail to do.
 - TOUGHNESS — Taking a stand when we must; to try again, even if it means following a more difficult path.
 - TEAMWORK — Respecting and utilizing the abilities of others, realizing that we work toward a common goal, for success depends on the efforts of all.
 - VIGILANCE* — Always attentive to the dangers of spaceflight; never accepting success as a substitute for rigor in everything we do.
2. To always be aware that suddenly and unexpectedly we may find ourselves in a role where our performance has ultimate consequences.
 3. To recognize that the greatest error is not to have tried and failed, but that in the trying we do not give it our best effort.

* Vigilance was added in 2003 after the Challenger disaster. Upon investigation, it was found that one of the contributing factors to the accident was the “reliance of past success as a substitute for sound engineering practices.”

As you can see, there is a lot of pride and self-worth that goes with

becoming and being a Flight Controller. Many felt that this creed extended into how they conducted their personal lives as well.

And how this relates to SRE?

Now the context is set. What does this have to do with site reliability engineering? Well, Google saw how SRE is firmly rooted in NASA Mission Control. This is referenced by the patch on the jacket that was given out at one time to all Google SREs.



Do all SREs have to be at the level of proficiency of flight controllers? Probably not. But there are a lot of takeaways that if noted, could make us better at our jobs. Let's take a look at the Flight Controller's creed and how it applies to the modern SRE. The first tenant is a list of characteristics.

1. To instill within ourselves these qualities essential to professional excellence:

- **DISCIPLINE** — Being able to follow as well as to lead, knowing that we must master ourselves before we can master our task.
- **COMPETENCE** — There being no substitute for total preparation and complete dedication, for space will not tolerate the careless or indifferent.
- **CONFIDENCE** — Believing in ourselves as well as others, knowing that we must master fear and hesitation before we can succeed.
- **RESPONSIBILITY** — Realizing that it cannot be shifted to others, for it belongs to each of us; we must answer for what we do—or fail to do.
- **TOUGHNESS** — Taking a stand when we must; to try again, even if it means following a more difficult path.
- **TEAMWORK** — Respecting and utilizing the abilities of others, realizing that we work toward a common goal, for success depends on the efforts of all.

Let's break down each characteristic and how it relates to SRE.

DISCIPLINE — Being able to follow as well as to lead, knowing that we must master ourselves before we can master our task.

It's no coincidence that the first characteristic is discipline. And it's understandable that many people have a negative connotation when it comes to this word. One of the negative connotations is *punishment*.

However, it is more productive to think of discipline's other connotations in regards to SRE. Site Reliability Engineering is a *discipline* (meaning a field of study, or a system of rules) where we take the results of *measurements*, apply agreed upon and codified *processes* and enact the results either in *real-time* or via *follow-up actions*. This cycle is repeated over and over again. **Discipline is about certainty.** Rigor in processes matters. If you know your teammates will follow through and document, you can resolve incidents with more confidence.

As we automate and expand measurement and processes, we will be enabled to complete these cycles faster and with more accuracy. If we can act on byproducts of the processes (either in real-time or follow-up actions) in as an efficient manner as possible, then the processes become self-refining, self-correcting, and a stream-lined workflow emerges naturally. This part takes **discipline**, as in self-control and orderly conduct. What does this look like for SREs and their organizations? Well, some examples might be:

- Having a ticketing system AND open tickets for work being done or about to be done, however trivial. Updating the ticket's progress (as close to real-time as possible to prevent information loss) is equally important.
- Sticking to established SLOs and error budgets and refining them over time. But always be mindful that the best, most tuned SLOs and error budgets alone just become "the noise before defeat" if you don't take action on the underlying problems. These are probably lurking in your backlog, depleting your error budget slowly. Work to update your monitoring systems and instrumentation and take action to meet your goals.
- Performing, reading, and reviewing incident retrospectives and following up on action items. This is very important as it's the only way you can actually improve the reliability of your operation. Just putting the follow-up actions into the backlog doesn't move the reliability dial appreciably forward and can lead to recurring incidents.

COMPETENCE — There being no substitute for total preparation and complete dedication, for space will not tolerate the careless or indifferent.

Some of our “unplanned work” incidents are things outside of our control like weather was with Apollo 12. During these incidents, SREs switch from the development cycle to the operations, incident management cycle. This is similar to flight controllers switching from their monitoring cycle into an incident command/control cycle should they detect anomalies that require urgent action. This context switching requires competence.

As John Aaron brought this level of competence to the game, how can we as SREs do the same? It's about *leadership*. Good leaders are prepared and dedicated. They treat their teams with care and respect. They're the kind of people you *want* to get into a spacecraft with.

Here are some examples of what competent leadership looks like in action.

- Not *cocky*. These teammates are here to help, not in it for the glory.
- People come to them because they feel like they'll get guidance.
- Can be counted on to help or give tools to help teammates help themselves.
- Are comfortable with saying “I don't know.”

These leaders don't need to prove anything to anybody; they speak with their hands. They're perceived as approachable.

CONFIDENCE — Believing in ourselves as well as others, knowing that we must master fear and hesitation before we can succeed.

Confidence flows from competence. **A confident SRE is not threatened by the confidence of their peers.** They believe in the

strengths of their teammates and realize that standing next to those people gives them even more confidence. Confident SREs also strive to elevate the confidence level of all contributors.

In practice, this might look like giving excellent feedback on a PR. Or it could be helping teammates develop a helpful runbook for a difficult recurring issue. However, this level of leadership always entails two things:

1. Leaders will help teammates resolve issues, but not solve them for the team.
2. Leaders are a coach as well as a player and seek to uplevel the entire team.

RESPONSIBILITY — Realizing that it cannot be shifted to others, for it belongs to each of us; we must answer for what we do - or fail to do.

Take ownership of the decisions and the calls that you make, the actions that you perform, or the actions that you request others to perform. If you break it, you own it. That doesn't always mean that you have the skills to fix it. But you can direct incident response efforts and rally your team, providing details of how the incident occurred and assisting in marshaling forces for a swift restoral of service.

This also means taking responsibility for your organization's reliability and acting as a steward for customer happiness. SREs are customer advocates who ensure that users are satisfied with their experience. This is a hefty responsibility, requiring vast amounts of inter-team communication, insight, empathy, and a vision for customer satisfaction.

TOUGHNESS — Taking a stand when we must; to try again, even if it means following a more difficult path.

Sometimes you know the solution. Sometimes it's an uphill battle to convince others of what you know. It's an art to stand your ground on a position that you know has substantial merit without trampling on others'

opinions. This is a type of toughness that requires introspection and the ability to engage others in the knowledge that you are presenting.

Other times, you're coming off of a 12-hour shift and you have dinner waiting for you at home, and all hell breaks loose. If you're carrying the pager and your team needs you, dinner should wait. And after the incident is over, you should document the underlying reliability issue leading to this incident in follow-up actions, so you have fewer dinners interrupted in the future. This is a type of toughness where you are dedicated to making sure the follow-up actions are not just tossed into the backlog and forgotten.

TEAMWORK — Respecting and utilizing the abilities of others, realizing that we work toward a common goal, for success depends on the efforts of all.

Do you remember the names of flight controllers, or astronauts? While astronauts often go down in history, flight controllers tend to be overlooked. However, one thing is for sure: the *astronauts* remember those flight controllers. How else would the mission be a success? Teamwork is key, even when some work is more visible or shiny than others.

In spaceflight, astronauts as well as mission control have set teams for missions. Astronauts train as a team so they know what to expect from each other not only in a crisis, but in the normal cadence of work that needs to be done during the mission. Mission Control also has squads of flight controllers who train together for the same reason.

There is an advantage in approaching things like on-call from a team perspective. There should always be a primary and a secondary on-call. It disperses the amount of stress when incidents arise. It can also reduce the time to resolution and service restoration as there are two brains with different views attacking the problem.

VIGILANCE — Always attentive to the dangers of spaceflight; never accepting success as a substitute for rigor in everything we do.

This is somewhat melodramatic by today's standards, as spaceflight in the early 70s was very daunting. It may be less daunting now, but just as

dangerous. Those of us who are not operating in the aerospace industry can still be lulled into the normalcy of our everyday operations. This includes the normality of embracing failure on a daily basis.

What we still must be vigilant for is **The Black Swan**. Black Swan events are events that you don't train for because they can't happen. Until they do. They live in the space of what you don't know that you don't know.

Set SCE2AUX was not a Black Swan event. John Aaron had seen the pattern before in a simulation. But the cryo tanks exploding on Apollo 13 was a Black Swan; no one had anticipated an event like that could happen. There were no simulations, and no drills. Black Swans can't be predicted by their very nature. All the monitoring in the world won't find them. Once a Black Swan happens, we look back and say "We should have seen it coming." Hindsight is always 20/20. But these events cease being Black Swans once we deal with them.

The way to survive a Black Swan is to be resilient. Resilience keeps you in the game. It buys you time so that discipline, competence, and teamwork can kick in and allow your team to work its way out of the situation. You can cultivate more resilience by training in simulations as a team. While the Black Swan event of Apollo 13 had not shown up in a drill, like John Aaron had seen in Apollo 12, various elements of the solution had been practiced. What the crews needed was the ability to think outside the box, and the time to stitch the solutions together. Plus the teamwork to realize and cancel work on a dead end all the while remaining focused and blameless.

2. To always be aware that suddenly and unexpectedly we may find ourselves in a role where our performance has ultimate consequences.

In the early days of spaceflight, changing hardware once the vehicle left the launch pad was very dangerous if not virtually impossible, requiring an EVA (Extra-Vehicular Activity, a "spacewalk"). Changing software was indeed very difficult as the majority of the software was etched in read-only memory, with only a few hundred words of writable core memory that could be used to judiciously "patch" existing code. Once the rocket left, there was no turning back: either the mission continued or

there would be an abort.

We have a lot of luxuries today, but we could learn from some practices that flight controllers performed regularly. The primary one I want to highlight here are *drills and simulations*. Flight controllers, when not active on a mission, simulated missions constantly. They simulated not just the “happy path” but the “unhappy path” as well.

The ability to inject faults into simulations was key to being prepared for spaceflight with a live crew or actual payloads. Though it may be macabre, having the crew die over and over again in the simulator increases the chances that they will live if similar situations come up during the actual flight. Remembrance of one of these simulations allowed John Aaron to make his “Set SCE2AUX” call. SREs today have access to these processes through Chaos Engineering platforms and methodologies, and through tried-and-true techniques like canarying deployments.

3. To recognize that the greatest error is not to have tried and failed, but that in the trying we do not give it our best effort.

Teams can perform drills based on intentionally misconfiguring configurations. At a previous company, I led Customer Engineering. I would host weekly Troubleshooting Extravaganza gaming exercises. Here’s how they worked:

1. The day before, members of the team would take time to intentionally figure out a way to “break” the product by a subtle misconfiguration. They would also have to find a way to recover. They would write up the scenario along with the solution and have it ready for the next day.
2. The day of the extravaganza, we would all assemble into a war room with a big projector screen. Individuals who had devised a scenario would put their name and an identification phrase on a piece of paper that would go into a hat or bowl.
3. The contents of the bowl would then be scrambled and a random person outside of the group would pick one entry. The entry would be read aloud to the group and the person owning the entry would

have 15 minutes to set up the failure with everyone out of the room.

4. When they were ready, we would re-enter the room and the person would declare what the symptom was and demonstrate it, as if a customer were calling in. The rest of the team then attempted to identify the problem (there is a difference between symptoms and problems) and come up with a mitigating strategy. They are only allowed to ask superficial questions to the scenario creator, such as “Is this the configuration parameter you intended?” This phase of the game was time boxed to 45 minutes.
5. At the end of the simulation, if the team hasn’t figured it out, the creator will go over the fault injection, and the path to recovery/mitigation.
6. At the end of the month, rewards are given to the creator who came up with the best scenario or the team that solved an issue the fastest. Often, one of these scenarios showed up in an actual customer incident days or weeks later.

It became obvious early on that the newer people on the team were more adept at creating difficult to solve scenarios than the senior members. This seemed to be because of “expert bias” (the experts knew how the happy path looks very well, and some may have designed it). Senior members also expect the product to be used a certain way and never consider “stupid mistakes,” or edge cases that those unfamiliar with the product would expose. The new engineers approached the product with a beginner’s mind, and they didn’t yet know the difference between a “stupid mistake” and a common use case (much like new customers).

Birds of a feather

Here is a mapping table of associations. It acts as a summary of what flight controllers did (and still do) and what some of us as SREs do. Some of the functions have definitely evolved over time.

As one boss at a company that I worked at a few years ago reminded me that, “We are just selling tickets, not landing the space shuttle.” It’s true that in most of our roles, human lives are not at stake as in manned spaceflight. However, company reputation and revenue often are. We can still approach our jobs with an equivalent amount of dedication to the tasks at hand.



Flight Controller	SRE
Monitor telemetry, take action	Respond to alerts, create alerts, take action
Lots of manual work (toil)	Eliminate toil (automate yourself out of a job)
Static roles (only during a mission)	Static roles (only during an incident)
Knows the infrastructure	Improves the infrastructure
Strong STEM background	Multidisciplinary
Disciplined	Should be ;)
Steep learning curve	Always learning from incidents
Contributes to runbooks	Creates runbooks
Precision in data recording and communication	Discipline in recording in tickets and wikis
Relentless Simulations	Chaos Engineering
“Failure is not an Option”	“Failure is inevitable” - avoid burnout
Be Right	Be Blameless

Site Reliability Engineering in the 21st century is still evolving. As the systems we are responsible for become more mission critical to our global communication and life support infrastructure, our jobs will begin to have lives in the balance. Like the early days of spaceflight, except we will have these responsibilities at scale.

Look at the Internet as an example for a novelty that has grown exponentially more important over time. The world wide Internet was resilient to the Black Swan of the Global Pandemic (we could debate whether it was a Black or Grey Swan, but that’s another essay in and of itself). For many businesses, teleconferencing systems like Zoom, Skype, and others had to take up the slack of human to human communication as office buildings closed down. These systems had to adapt to a massive increase in customers and traffic, often building out infrastructure that was planned over the next three years in a matter of weeks.

And all through this, underpinning it all was the Global Internet

Infrastructure which pretty much went unnoticed and unsung because it was so resilient. Its operators had practiced their own version of the Flight Controller's Creed for decades. And like their predecessors, no one knows their names.

I'll conclude with a small homage to our great grand-SRE-parents, the flight controllers of Mission Control. I submit to you the SREs creed, patterned after the Flight Controller's Creed, hopefully keeping with the spirit, yet adapting it for the 21st century challenges at hand. May we live up to your legacy as we step into our own. At scale.

The SRE's Creed

1. To instill within ourselves these qualities essential to professional excellence:
 - DISCIPLINE — Being able to be counted on to do what you've signed up for. Imagine also the traits of consistency and dedication.
 - COMPETENCE — There being no substitute for total preparation and complete dedication, complex systems are inscrutable.
 - CONFIDENCE — Confidence is not the same as cockiness. Believe in others as well as yourself and lead by lifting everyone around you up.
 - RESPONSIBILITY — Realizing that it cannot be shifted to others, for it belongs to each of us; we must answer for what we do - or fail to do.
 - BLAMELESS — Realizing that everyone doesn't view the world or situation as you do and that their views are also valid and useful. Be mindful of that during crises and be respectful of that during following retrospectives.
 - TEAMWORK — Realizing that each member of your team has unique skills that can be brought to the tasks at hand,

and lead in a way that you can optimize that gift.

- VIGILANCE — Always attentive to the possible Black Swan; never accepting success as a substitute for rigor for Black Swans come when you least expect them.
- RESILIENCE — You cannot predict a Black Swan, but if your systems and processes are resilient, you can survive them.

2. To always be aware that suddenly and unexpectedly we may find ourselves in a role where the outcome of our actions can have consequences for everyone around us, the entire company, and maybe the world.

3. To recognize that the greatest error is not to have tried and failed, but that in the trying we do not give it our best effort. Failure is always an option, but giving up isn't.

If you enjoyed this blog post, check out these resources:

- [Webinar: Advanced Incident Management](#)
- [eBook: The Complete Guide to Pragmatic Incident Response](#)
- [5 On-Call Practices to Help you Sleep through the Night](#)