ManageEngine
ServiceDesk Plus

ITSM and aviation

Lessons from the sky



Five aviation lessons your IT team can borrow from one of the world'smost reliable industries, to deliver excellent IT service experiences.

Table of contents

No.	DESTINATION	Page
01	Introduction	01
02	Overview	02
03	Lesson 1 — Standardize before you fly	05
04	Lesson 2 — Build your ECAM	09
05	Lesson 3 — Don’t wait for the warning light	13
06	Lesson 4 — Practice the crash before it happens	19
07	Lesson 5 — Read the black box before the next flight	24
08	Conclusion	29
09	About ServiceDesk Plus	30
10	About ManageEngine	31
11	About the author	32

Introduction

Every unplanned outage has a cost: lost productivity, frustrated users, strained customer relationships, and an IT team burning hours on problems that repeatable processes could have prevented. For most service desks, the gap between reactive firefighting and reliable operations is a process gap.

This e-book identifies five places where that gap tends to be widest and shows how to close each one, drawing on the operating principles of commercial aviation. Each lesson is grounded in a story about Zylker Technologies, a fictional 1,200-person financial services firm whose five-hour authentication outage prompted a complete overhaul of its ITSM practices.

Five lessons, five practices, and one platform

Lesson	ITIL practice	What changes
Standardize before you fly	Change enablement	Fewer change related incidents
Build your ECAM	Incident management	Faster and more structured response
Don’t wait for the warning light	Problem management	Early detection to ensure fewer repeats
Practice the crash	Knowledge management	Teams that execute under pressure
Read the black box	Continual improvement	Patterns caught before the next outage

Disclaimer: Each lesson connects to a specific ManageEngine ServiceDesk Plus capability that is available today.

Overview

Commercial aviation is the safest form of mass transportation ever built. The global accident rate in 2023 was 0.80 per million departures, a figure that would have seemed impossible to the engineers who watched early aircraft fall from the sky. That record was not built on luck. It was engineered, lesson by lesson, through rigorous investigation, disciplined standardization, and an industry-wide commitment to learning from every failure, however small.

Your IT service desk runs on similar dynamics. Both fields involve complex, interdependent systems where teams make high-stakes calls with incomplete information, and where users expect everything to work regardless of the complexity behind the scenes. The tools and the physics are quite different, but the management challenges that come with running critical, time-sensitive operations are more similar than most IT teams realize.

The five lessons in this e-book come from aviation's core operating principles: standardization, structured alerting, anticipatory monitoring, simulation-based preparation, and data-driven improvement.

Each maps directly to a capability your service desk should have, and each is explored through the story of one IT team at Zylker Technologies, a fictional company that learned these lessons under pressure.

The ITSM platform referenced throughout this e-book is ServiceDesk Plus, developed by ManageEngine. ServiceDesk Plus is an AI-driven unified service management platform covering incident management, change management, problem management, knowledge management, SLA management, and analytics. It is used by more than 100,000 organizations globally. Where this e-book references specific platform capabilities, including change templates, change workflows, SLA escalation rules, the Problems module, the Solutions module, and the Reports module, those paths are drawn and verified against the product.

Meet Zylker Technologies

Zylker Technologies is a 1,200-person financial services firm with a 25-person IT team. The operation looks fine on paper: thousands of tickets closed each month, a reasonable first-call resolution rate, and a service desk that rarely misses its service level agreements (SLAs), at least until something significant goes wrong.

On a Tuesday morning in March, a routine change request works its way through Zylker's change advisory board (CAB). The change is an upgrade to the company's authentication service, the kind of update that has been done before, and the CAB approves it in 20 minutes. At 10:47am, the change goes live. By 11:05am, 1,200 employees across three offices cannot log in to any internal application. Customer service has gone dark. Finance cannot process invoices. The IT team is fielding 300 tickets an hour.

The next five hours expose every weakness in Zylker's IT operations. This e-book follows that incident from the moment the change was deployed to the point where the team rebuilt their systems and processes. ServiceDesk Plus is the platform they used to make those changes stick.

Aviation and ITSM: the parallels worth taking seriously

Aviation and enterprise IT have more in common than most IT teams realize, and three realities make the comparison worth examining closely.

First, both fields rely on complex, interdependent systems where a single failure can cascade unpredictably across many others. An aircraft hydraulic fault does not just affect the landing gear; it reshapes the entire procedure set for the flight crew. An authentication outage does not just lock users out of one application; it cuts off every service that depends on it.

Second, both require teams to make accurate calls under pressure, with incomplete information, while managing the consequences of decisions already made. Crew resource management, which governs how cockpit teams communicate, divide workload, and reach decisions during an emergency, has a direct parallel in how your incident response team operates.

Third, neither industry gets a second chance in its most critical moments. An aircraft cannot pull over, and a payment processing system cannot ask users to try again tomorrow. The cost of failure is immediate in both cases.

The key difference is that aviation had its catastrophic failures early, investigated them rigorously, and built industry-wide learning mechanisms, including mandatory incident reporting, standardized procedures, and independent accident investigation, that turned each failure into measurable, repeatable improvement.

LESSON



01

Standardize
before you fly

ITIL practice in focus: Change enablement

Minimize the risk and impact of changes by ensuring every change is assessed, authorized, and reviewed before implementation, using standardized procedures that apply consistently regardless of who initiates the change.

In ServiceDesk Plus:

Change enablement is supported through change templates and change workflows, which together enforce mandatory fields, stage-based approvals, CAB evaluation, and backout planning.

In the cockpit: Checklists that command and preflight SOPs

In 1935, a Boeing B-17 prototype crashed on its maiden evaluation flight at Wright Field, Ohio. The cause was not mechanical failure. The pilot, a highly experienced aviator, forgot to release the gust lock before takeoff. He had completed similar flights many times, yet he missed a step that should have been automatic, a standard operating procedure (SOP).

The military's response was to create a mandatory preflight checklist: a formalized process that removed individual memory as the last line of defense. That decision became the foundation of modern aviation safety culture.

Today, every commercial flight operates under SOPs governing hundreds of tasks before a single engine starts. In Airbus aircraft, the Electronic Centralized Aircraft Monitor (ECAM) sequences mandatory checks and prevents the crew from advancing until each one is confirmed. These procedures exist because building systems that perform consistently, regardless of who is in the seat, is more reliable than depending on any individual's experience or attention.



A gust lock immobilizes an aircraft's controls until it's manually removed before takeoff. Mandatory change-template fields work the same way: progress remains blocked until the change type, associated CIs, operational impact, and rollout and backout plans are fully documented, ensuring every critical detail is in place before work begins.

At Zylker: The change that slipped through

The authentication upgrade that triggered Zylker's outage was submitted as a standard change, a category that implied it was low-risk and preapproved. But Zylker's standard change process had no mandatory template, no field requiring test environment sign-off, and no workflow that enforced a backout plan. The CAB approved the request in 20 minutes because everyone assumed someone else had already done the full assessment, when in fact no one had.

The technician who submitted the request used the same form he always used and did not know what was missing, because the form itself did not capture that information automatically.

What Zylker did

After the incident, Zylker rebuilt its change enablement process around standardization, treating templates and workflows as the organizational equivalent of preflight checklists. Here are the actions Zylker took:

- Built category-specific change templates in ServiceDesk Plus including separate templates for network, authentication, and database changes. Each template enforces mandatory fields covering impacted services, test environment sign-off, backout plan, change window, and required approvers at each tier.
- Configured change workflows to route each change type based on impact. A Normal change cannot advance past the CAB Evaluation stage without sign-off from IT leadership and the affected business unit. A standard change can be auto-approved, provided all template fields are complete.
- Added a mandatory pre-implementation validation field to every Tier 1 and Tier 2 template asking whether the change has been tested in a non-production environment. The field is required in the template canvas, and the change cannot move past the Planning stage without a confirmed response.

Change templates reduce process variance across the team. Change workflows enforce accountability at every stage. When every engineer works from the same template with the same mandatory fields, the CAB can focus on risk assessment rather than chasing missing information.

Aviation echo

Change template = preflight checklist.

Change workflow = SOP.

Backout plan = go-around procedure.

Mandatory pre-implementation sign-off = ECAM procedure that cannot be bypassed, because the stakes don't allow for improvisation.

Standardization is about ensuring that the process catches what individual judgment can miss, consistently and reliably, regardless of who handles the request.

How should your organization get started?

Start with your highest-risk change types and work outward.

- Identify the two or three change categories most likely to cause a service outage if something goes wrong.
- Review your current change templates for those categories. Add mandatory fields for impacted services, test environment sign-off, and backout plan if they are missing.
- Define what makes a high impact change an emergency change versus low impact a normal change, and document the criteria.
- Configure a change workflow for your highest-impact change type that requires an explicit sign-off before the planning stage closes.

LESSON



02

Build your ECAM:

A system that tells you what's wrong

ITIL practice in focus: Incident management

Restore normal service operation as quickly as possible, minimizing adverse impact on the business by ensuring a structured, consistent response to every incident regardless of severity.

In ServiceDesk Plus:

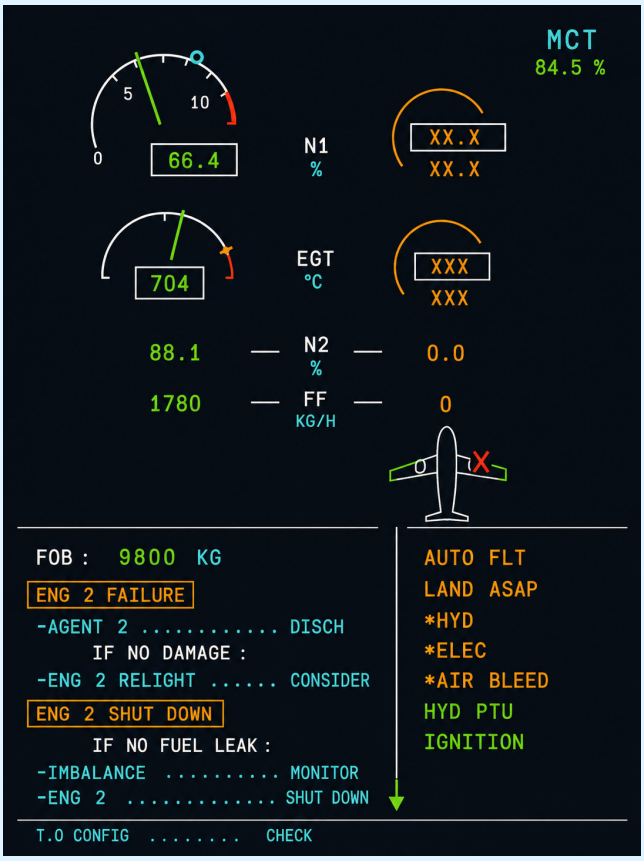
Incident management is supported through incident templates, priority and urgency fields, SLA escalation rules, and notification rules.

In the cockpit: ECAM and the prioritized alert hierarchy

On November 4, 2010, Qantas Flight QF32 departed Singapore's Changi Airport with five pilots on board. Shortly after takeoff, an uncontained engine failure caused catastrophic damage, triggering simultaneous faults across hydraulics, fuel, flight controls, and landing gear. Within seconds, the aircraft's ECAM was displaying more than 60 warning and caution messages.

All 469 people on board survived, and the aircraft landed safely, largely because of how the ECAM was designed. Rather than presenting all alerts simultaneously, the system ranked warnings by severity, presented them in sequence, and provided the specific procedure for each one. The crew worked through them systematically over 90 minutes and returned to Singapore without further incident.

Boeing's equivalent system is the Engine Indicating and Crew Alerting System (EICAS). Both are built on the same principle: in a high-pressure situation, the crew should not have to determine their own priority order. The system makes that determination for them.



The ECAM display on QF32 ranked more than 60 simultaneous fault warnings by severity and converted them into actionable procedures. ServiceDesk Plus works the same way: incident priorities, SLA escalations, and task flows ensure teams address the most business-critical issues first with every next step clearly defined, even under pressure.

At Zylker: 50 minutes to convene a war room

When Zylker's authentication outage cascaded at 11:05am, three monitoring tools fired simultaneously. One sent an email, one sent an SMS to the infrastructure team, and one logged a critical event in the ITSM platform, though that last alert was not configured to trigger an escalation. The first responder opened a ticket and assigned it to the application team. The application team escalated the issue to the infrastructure team, prompting them to call a vendor. Fifty minutes passed before anyone declared a major incident.

By that point, over 500 tickets had accumulated. There was no tiered incident classification, no major incident playbook, and no defined escalation path. The team was improvising the way the QF32's crew would have if the ECAM had delivered 60 undifferentiated alerts with no sequence and no associated procedures.

What Zylker did

Recognizing that the absence of structure had cost them nearly an hour of critical response time, Zylker overhauled its incident management process, embedding a tiered classification system, formal escalation paths, and a designated incident commander role directly into ServiceDesk Plus. Here is how Zylker prioritized these actions:

- Defined three incident priority tiers with documented criteria to ensure consistent classification: SEV-1 for business-critical incidents where multiple services are down and revenue is affected, SEV-2 for degraded service where partial functionality remains and a workaround is available, and SEV-3 for isolated minor issues with no material business impact.
- Created a SEV-1 incident playbook covering who to notify within the first five minutes, when to escalate to IT leadership, how to communicate with affected business units, when to engage external vendors, and how to maintain a running timeline in ServiceDesk Plus.
- Configured SLA escalation rules in ServiceDesk Plus to fire before response and resolution deadlines are breached rather than after, giving technicians and the on-call manager advance notice while time to act remains. SEV-1 incidents trigger SMS alerts directly to the on-call manager, not just email.

- Assigned an incident commander to every SEV-1 incident, designating one person with clear authority to coordinate the response, manage communications, and make escalation decisions.

Aviation echo

Incident priority tier = ECAM warning/caution/advisory hierarchy.

Playbook = ECAM procedure sequence.

Incident commander = pilot-in-command.

Automated SEV-1 escalation = ATC declaring an emergency and clearing the airspace.

The authentication incident that took Zylker five hours to resolve would, after these changes, have had a war room convened in under 10 minutes. The team was the same. What changed was that the system was now providing the structure for how to respond.

How should your organization get started?

Document your criteria first. Each step depends on them.

- Write out the specific criteria that make an incident a SEV-1 versus a SEV-2. Make the distinction clear enough that any technician can classify an issue consistently.
- Draft a one-page SEV-1 incident playbook on who to call, in what order, within what timeframe. It does not need to be exhaustive to be useful.
- Designate an incident commander for SEV-1 incidents and name at least one backup in case the primary is unavailable.
- Review your current SLA escalation rules. Confirm that at least one level fires before the deadline, not only after a breach occurs.

LESSON



03

Don’t wait for the **warning light**

ITIL practice in focus: Problem management

Reduce the likelihood and impact of incidents by identifying the root causes, managing workarounds and known errors, and shifting your team from reactive firefighting to proactive service health.

In ServiceDesk Plus:

Problem management is supported through the Problems module, creating problem records, associating related incidents, and tracking root cause analysis to complete resolution. Cluster analysis across incident data surfaces patterns that no single ticket makes it visible on its own, enabling the team to move from resolving failures to eliminating the situations that produce them.

In the cockpit: GPWS and the case for proactive signals

Before the Ground Proximity Warning System (GPWS) was introduced in the 1970s, Controlled Flight Into Terrain (CFIT), in which aircraft fly into the ground or water while still fully functional, was one of aviation's leading causes of fatalities. In these accidents, the aircraft were operating normally, but the flight crew had no reliable indication that the rate of descent was placing them on a collision course with terrain until recovery was no longer possible.

GPWS addressed this by monitoring not just altitude but rate of descent, gear and flap configuration, and terrain proximity, alerting the crew when a combination of those factors created a dangerous trajectory. The system does not wait for an impact condition to develop. It fires when the trend is already going in the wrong direction.

Modern airlines extend this principle through avionics telemetry: continuous monitoring of hundreds of engine and system parameters against expected baselines. Leading carriers sustain dispatch reliability above 99% precisely because predictive maintenance flags components trending toward failure during ground maintenance cycles, before they ever cause an in-service problem.



The EGPWS tracks descent rate and terrain proximity continuously, alerting the crew before recovery becomes impossible, likewise, threshold alerts in ManageEngine OpManager Nexus flag infrastructure issues as a ticket, before they ever reach your users.

At Zylker: The warning that was already there

After the authentication outage, Zylker's team ran a post-incident review and pulled the system logs. The finding was not surprising in retrospect, but it was uncomfortable: a database performance issue had been recorded 48 hours before the incident, and query latency had spiked twice in the preceding 24 hours. Neither event had triggered an alert, and neither had been reviewed by anyone on the team.

The early warning signs were already in the logs. What Zylker lacked was a system to surface them before they compounded into an outage. Its monitoring setup was reactive by design, with thresholds configured to fire at the point of failure rather than at the point where intervention would still have made a difference.

What Zylker did

Zylker rebuilt its monitoring strategy around the principle that the goal is not to know when something has failed—it's to know when something is trending toward failure. Zylker then initiated these solutions:

- Connected ManageEngine OpManager Nexus to the servers, network devices, and application layers underpinning critical services—authentication, email, and core business applications. Defined performance baselines for each service and configured threshold alerts against CPU utilization, memory pressure, query response time, and interface error rates. When any metric deviates from its baseline, OpManager Nexus automatically opens an incident ticket in ServiceDesk Plus, prepopulated with the affected device, current metric value, and alert severity. The ticket exists before any user has reported a problem.

- Used ServiceDesk Plus's analytics module to run a weekly cluster analysis on recently closed and still-open incidents, grouping them by affected CI, reported service, and symptom category. A cluster of five authentication failures raised by users across different departments over three days looked routine at the ticket level—each was resolved individually and closed. Viewed as a cluster, they pointed to a directory service that OpManager had been flagging with elevated query latency for 36 hours. The cluster surfaced the underlying problem before a sixth ticket came in.
- Established a standing rule from that finding: any cluster of three or more incidents sharing a common CI, service, or symptom signature in a rolling 14-day window is escalated to a problem record immediately, regardless of whether each individual ticket was resolved. Closing the ticket does not close the problem. Only confirmed root cause identification and remediation closes the problem record.
- Built a weekly incident review into the service desk manager's Monday routine: 30 minutes reviewing the previous week's data by category, technician, and service. The cluster analysis runs before this meeting so patterns are visible before the conversation starts, rather than discovered during it.

OpManager Nexus catches infrastructure degradation at the metric level before it becomes a ticket. Cluster analysis catches patterns that only become visible across tickets before they become a recurring problem. Together, they give Zylker two distinct early-warning layers: one at the infrastructure layer, one at the incident layer.

Aviation echo

OpManager Nexus threshold alert → ServiceDesk Plus ticket = avionics telemetry: Continuous parameter monitoring fires before the crew notices anything wrong.

Cluster analysis across incident tickets = Flight Operational Quality Assurance (FOQA): No single flight triggered an alert, but analyzing the data across many flights revealed a hydraulic seal trend that grounded the aircraft for inspection before an in-service failure.

Problem record triggered by incident cluster = National Transportation Safety Board (NTSB) safety recommendation: The investigation doesn't close when one accident report is filed—it closes when the contributing factor is eliminated across the fleet.

Weekly incident review = dispatch reliability tracking: Structured review of trends before they become failures.

The shift from reactive to proactive ITSM is not only a mindset change. It is a structural one. Zylker's team built a system in which vigilance was the default.

How should your organization get started?

Start by looking at the last 30 days of incidents.

- Pull a report of incidents from the past 30 days grouped by affected CI and service category. Look for any CI or service appearing in three or more separate tickets—even if each was resolved. That cluster is your first problem record candidate.
- Define performance baselines for your two or three most critical services: What does normal CPU, memory, response time, and error rate look like? Without a baseline, no monitoring tool can tell you when something is trending wrong.
- If you have OpManager Nexus or a comparable infrastructure observability tool, verify that threshold alerts are configured to open ServiceDesk Plus tickets automatically—not just send email. Email gets lost; a ticket in your queue gets assigned and tracked.
- Schedule a recurring weekly incident review. Run the cluster grouping report before the meeting so patterns are on the table when the conversation starts.

LESSON



04

Practice the crash

before it happens

ITIL practice in focus: Knowledge management

Ensure the right information reaches the right people at the right time, especially under pressure.

In ServiceDesk Plus:

Knowledge base articles are stored in the Solutions module. The Solutions module keeps operational playbooks, escalation paths, and incident procedures internal through technician-only visibility. Teams can practice these same articles during tabletop exercises, building familiarity with the actual platform and workflows.

After each drill, identified gaps are addressed, and updates are recorded in the knowledge article's version history, including what changed, and when. This ensures teams respond using documentation they have already practiced with and know is current.

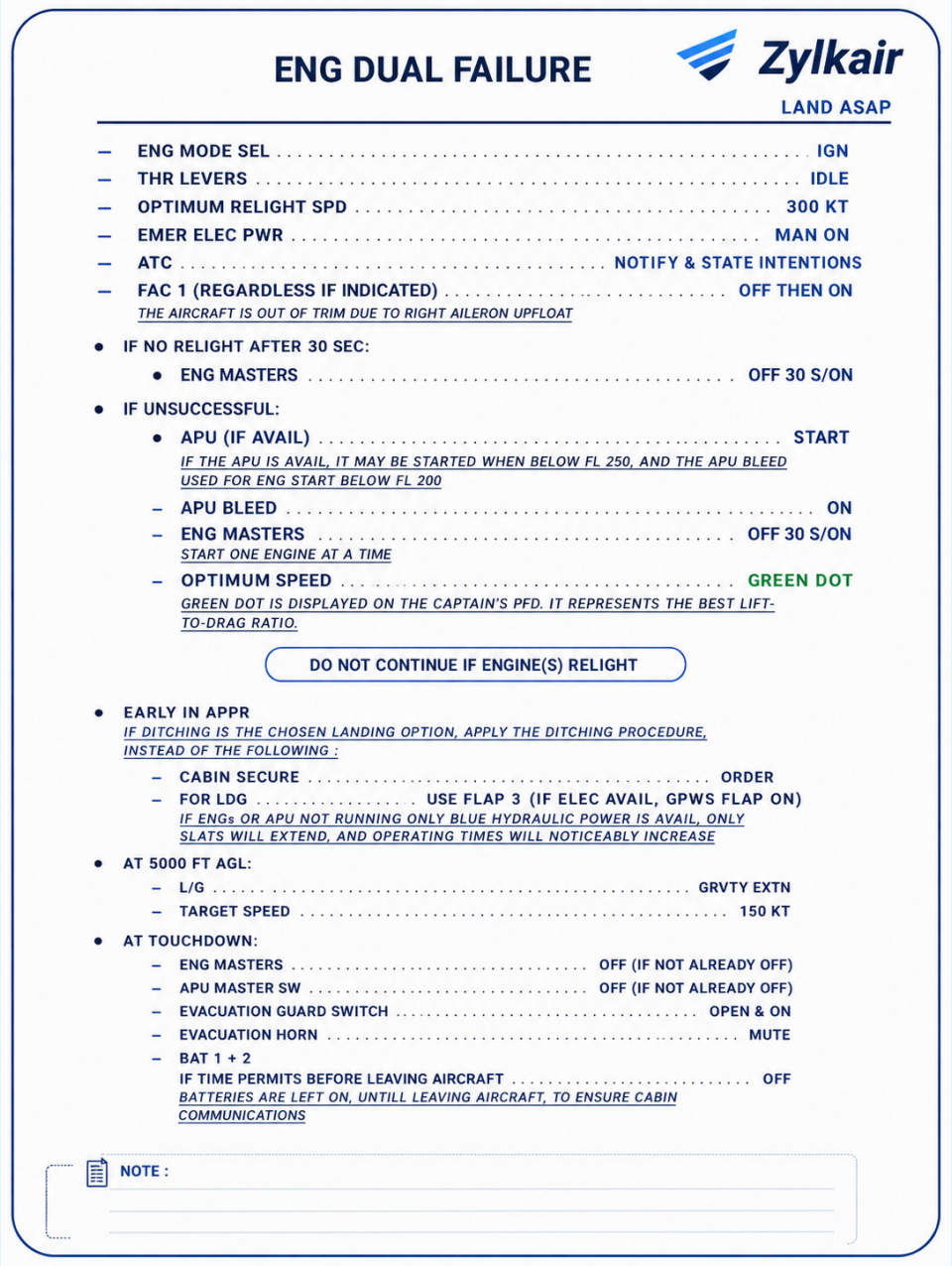
In the cockpit: The flight simulator and crew resource management

Before a commercial pilot is authorized to carry passengers, they spend extensive time in a flight simulator practicing not routine conditions but failure scenarios: engine fires at takeoff, hydraulic failures at altitude, bird strikes, and electrical failures in instrument conditions. These are situations a pilot may never encounter in a full career, but they require an immediate, correct response if and when they occur.

Crew resource management training extends this preparation further by focusing on how cockpit teams communicate under pressure, how workload is distributed during an emergency, how authority gradients affect decision-making, and how to avoid the cognitive traps that contribute to human error in high-stress situations. Crew resource management as a discipline grew directly from accident investigations in which the aircraft was flyable but the crew's coordination and communication had broken down under pressure.

Beneath both simulator training and crew resource management are documented procedures such as SOPs, emergency checklists, and briefing guides that capture how crews should respond. Simulators test whether these rules actually work in real scenarios and highlight errors before procedures are updated.

The 2009 Miracle on the Hudson," in which Captain Chesley Sullenberger successfully ditched US Airways Flight 1549 in the Hudson River after dual engine failure, demonstrated the importance of preparation. Captain Sully had been practicing engine failure scenarios for 40 years, and his first words to the first officer after the bird strike were calm, specific, and task-oriented. These actions helped ensure all 155 people on board survived. This outcome was the result of practiced, systematic training.



ENG DUAL FAILURE

Zylkair

LAND ASAP

- ENG MODE SEL ..... IGN
- THR LEVERS ..... IDLE
- OPTIMUM RELIGHT SPD ..... 300 KT
- EMER ELEC PWR ..... MAN ON
- ATC ..... NOTIFY & STATE INTENTIONS
- FAC 1 (REGARDLESS IF INDICATED) ..... OFF THEN ON

  THE AIRCRAFT IS OUT OF TRIM DUE TO RIGHT AILERON UPFLOAT

- IF NO RELIGHT AFTER 30 SEC:
  - ENG MASTERS ..... OFF 30 S/ON
- IF UNSUCCESSFUL:
  - APU (IF AVAIL) ..... START

    IF THE APU IS AVAIL, IT MAY BE STARTED WHEN BELOW FL 250, AND THE APU BLEED USED FOR ENG START BELOW FL 200
  - APU BLEED ..... ON
  - ENG MASTERS ..... OFF 30 S/ON

    START ONE ENGINE AT A TIME
  - OPTIMUM SPEED ..... GREEN DOT

    GREEN DOT IS DISPLAYED ON THE CAPTAIN'S PFD. IT REPRESENTS THE BEST LIFT-TO-DRAG RATIO.

DO NOT CONTINUE IF ENGINE(S) RELIGHT

- EARLY IN APPR

  IF DITCHING IS THE CHOSEN LANDING OPTION, APPLY THE DITCHING PROCEDURE, INSTEAD OF THE FOLLOWING :
  - CABIN SECURE ..... ORDER
  - FOR LDG ..... USE FLAP 3 (IF ELEC AVAIL, GPWS FLAP ON)

    IF ENGs OR APU NOT RUNNING ONLY BLUE HYDRAULIC POWER IS AVAIL, ONLY SLATS WILL EXTEND, AND OPERATING TIMES WILL NOTICEABLY INCREASE
- AT 5000 FT AGL:
  - L/G ..... GRVTY EXTN
  - TARGET SPEED ..... 150 KT
- AT TOUCHDOWN:
  - ENG MASTERS ..... OFF (IF NOT ALREADY OFF)
  - APU MASTER SW ..... OFF (IF NOT ALREADY OFF)
  - EVACUATION GUARD SWITCH ..... OPEN & ON
  - EVACUATION HORN ..... MUTE
  - BAT 1 + 2

    IF TIME PERMITS BEFORE LEAVING AIRCRAFT ..... OFF

    BATTERIES ARE LEFT ON, UNTILL LEAVING AIRCRAFT, TO ENSURE CABIN COMMUNICATIONS

NOTE :

This checklist captures exactly what to do during a dual engine failure, and pilots drill it repeatedly before it's ever needed. Your incident playbook in ServiceDesk Plus only holds up under pressure if the team has actually practiced using it.

At Zylker: The playbook that hadn’t been practiced

Three months after the authentication outage, Zylker had done the knowledge management work: a thorough SEV-1 incident response playbook, reviewed by IT leadership, stored in ServiceDesk Plus’s Solutions module, and accessible to every member of the team. When a network failure hit on a Thursday afternoon, the playbook was in place and the team knew it existed.

Even so, the team took 40 minutes to convene a war room. The incident commander named in the playbook was on leave, and no backup had been designated. Two senior engineers started working on the same problem in parallel without coordinating. The business stakeholder communication went out an hour late. The playbook itself was not the problem and the necessary knowledge existed. What hadn’t been tested was whether the team could actually use it when the situation was real.

What Zylker did

Zylker instituted a quarterly tabletop exercise program—structured drills that simulate realistic incident scenarios using the actual playbooks in ServiceDesk Plus. Here are the steps Zylker took:

- **Separated internal and end-user knowledge:**
  Moved the SEV-1 playbook, escalation paths, on-call contacts, vendor SLAs, and stakeholder communication templates into the Solutions module with technician-only visibility. End users see a separate knowledge base, while the incident response team works from an internally controlled set of trusted, current articles.
- **Designed scenarios around real incidents:**
  Built tabletop exercises around Zylker’s incident history, involving the full response team with clearly assigned roles. Debriefs assessed communication, coordination, and decision-making alongside technical response, applying the same crew resource management principles used in cockpit emergencies.

- **Practiced directly in ServiceDesk Plus:**
  Required every participant to navigate the playbook in the same ServiceDesk Plus interface used during live incidents. This exposed execution gaps under pressure so they could be addressed before the next exercise.
- **Updated and version-controlled solution articles:**
  Logged gaps identified during each exercise as ServiceDesk Plus tasks, resolved them, and updated the relevant articles. Version history records what changed, and when, ensuring technicians always use a current, tested playbook with a traceable revision history.

Aviation echo

Tabletop exercise = recurrent simulator training.

Technician-only solution articles = crew operating manuals.

Running drills in ServiceDesk Plus = practicing with actual SOPs.

Version-controlled solution articles = flight manual amendments.

When the next network failure hit six months later, the team resolved it in 22 minutes. The incident commander was clear from the outset, the war room was operational in under 10 minutes, and the business stakeholder update went out in the first half hour. They used the same playbook. The difference was that this time the team had practiced it.

LESSON



05

Read the **black box** before the next flight

ITIL practice in focus: Continual improvement

Align an organization's services and practices with changing business needs through the ongoing identification and improvement of services, components, and practices—using data and evidence, not instinct.

In ServiceDesk Plus:

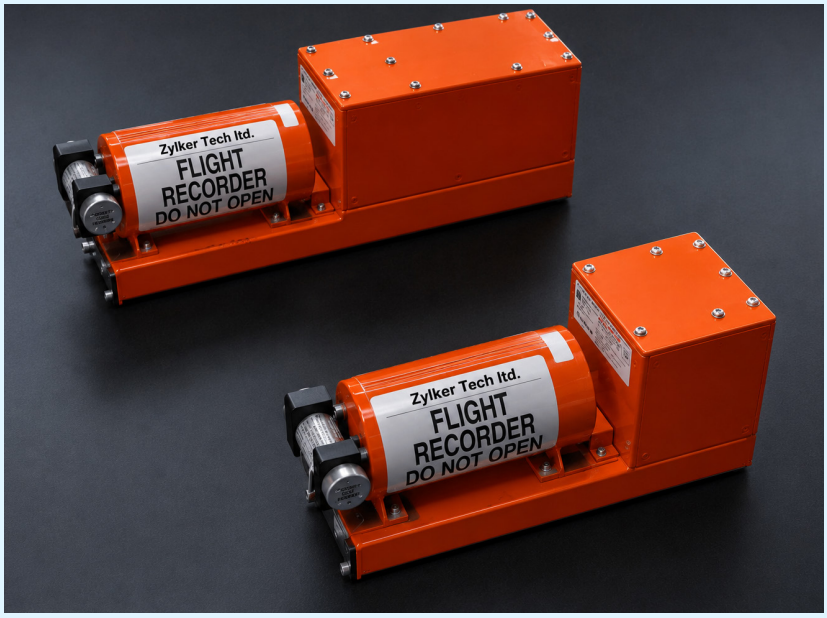
Continual improvement is supported through the Reports module (custom reports, including request metrics, SLA compliance, and technician performance), the survey module (automated satisfaction surveys triggered on ticket closure), and customizable Home dashboards with table and graph widgets spanning requests, problems, and changes.

In the cockpit: The flight data recorder and systemic learning

Every commercial aircraft carries two flight recorders: the flight data recorder and the cockpit voice recorder, together known colloquially as the 'black box.' Designed to survive crashes, these devices capture thousands of flight parameters and hours of cockpit audio that investigators use to reconstruct precisely what happened and why.

The black box's most significant contribution to aviation safety, however, is not what happens after an accident but what happens after every flight. Airlines routinely analyze the flight data recorder data as part of FOQA programs, using automated tools to flag anomalies including hard landings, unstabilized approaches, speed deviations, and temperature overshoot. Most of these events are identified and corrected before they ever develop into incidents.

The NTSB publishes detailed investigation reports for every major aviation accident, identifying systemic contributing factors rather than just proximate causes and issuing industry-wide safety recommendations. Collision avoidance systems, wind shear detection technology, and fuel exhaustion protocols all trace directly to specific NTSB findings.



The black box captures every flight parameter so investigators can find the pattern behind a failure, in the same way, ServiceDesk Plus' reports and dashboards turn your closed tickets into evidence worth examining, instead of records that just get filed.

At Zylker: Connecting dots that weren't being connected

Six months after the authentication outage, Zylker's service desk had transformed: no major incidents, consistently met SLAs, and change templates that had already flagged three high-risk changes before production, something the old process would have missed. By October, the service desk manager ran the first structured monthly report. It raised no alarm, but revealed something more significant.

Mean time to resolve (MTTR) for remote access tickets had risen 40% over the preceding quarter. No single week looked bad enough to act on, but the trend, plotted across three months, told a different story. User satisfaction scores from the new survey module confirmed it: remote access was the lowest-rated service category on the desk, not because tickets were being left open but because users kept coming back. While the ticket were being marked as resolved, in a lot of cases, the user did not receive a permanent fix to their issue.

Cross-referencing ticket data with problem records and change history revealed that the same workaround had been used across seemingly unrelated tickets with different users, symptoms, and dates. It helped teams stay within SLA limits and improve first-call resolution rates, making every metric appear green. But beneath the surface, the data revealed a "watermelon effect", green on the outside, red within, with recurring issues that threatened operational resilience.

The governance process had worked as designed, but it was never built to examine its own patterns. Zylker had built the black box; in October, it was opened for the first time. Clear dashboards and reports would now help decision-makers identify shortcomings early, address root causes, and embed continual improvement before operations broke down.

What Zylker did

Zylker built a structured analytics practice around its ServiceDesk Plus data, treating historical incident, change, and problem records the way an NTSB investigator treats a flight data recorder: as evidence to be examined, not just records to be filed.

- Built a monthly ITSM health report for IT leadership using ServiceDesk Plus's Request Metrics module. It tracked MTTR by incident category, SLA compliance by technician and service, and first-call resolution rates. The October report was Zylker's first view of these metrics as an ongoing trend rather than isolated snapshots, making the growing remote-access issue visible.
- Added automated satisfaction surveys to resolved incidents, triggered when tickets close. The scores gave leadership qualitative insight alongside MTTR data and confirmed that remote-access users were experiencing recurring issues: the tickets were being closed, but the underlying problem remained unresolved.
- Created a formal problem record for the remote-access issue—the first identified through analytics rather than automatic incident-volume escalation. By combining ticket data, satisfaction scores, and resolution notes, Zylker traced the issue to a known error repeatedly managed with work-arounds across a dozen seemingly unrelated tickets. The record remained open until the root cause was confirmed and a permanent fix implemented.
- Established a continuous-improvement practice: patterns in monthly reports—rising MTTR, declining satisfaction, or repeated workarounds—are reviewed as potential problem records, even when individual tickets do not meet escalation thresholds. Closing a ticket does not mean the problem is solved.
- Configured real-time dashboards for the service desk manager and team leads, combining request, problem, and change data in table and graph widgets. Ticket volumes by priority, technician workload, open problems, and SLA status—previously dependent on manual exports—are now visible in one place.

- Established quarterly service reviews with each business unit, using ServiceDesk Plus reports to show IT's performance, identify gaps, and track improvements. The remote-access findings led the Q4 agenda, helping the affected business unit understand both why the issue had remained hidden and how it would be addressed.

Aviation echo

Monthly ITSM health report = NTSB investigation findings.

MTTR trend analysis = FOQA anomaly detection.

Quarterly service review = safety board briefing.

Problem record via analytics = NTSB safety recommendation.

October's remote access issue never escalated into a major incident not due to a lack of an underlying issue, but because proactive analytics caught it before it broke out. The data had remained in ServiceDesk Plus for months; the turning point was simply building the habit of looking for it.

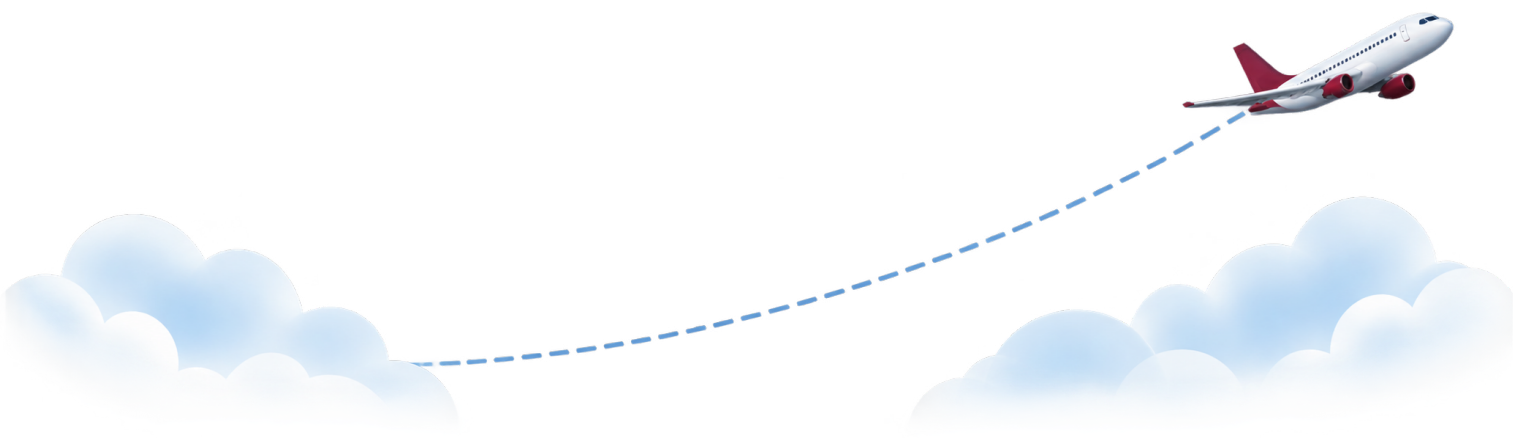
Conclusion

Zylker's story did not end with the authentication outage. That five-hour incident became the catalyst for building a genuinely resilient operation, forcing an IT team that had been improvising in real time to rethink its cascading failures and handle hundreds of frustrated users. Because it didn't change its team or its technology, the resulting transformation was entirely structural.

Aviation did not become the world's safest form of transportation through luck or talent. It got there through structural guardrails: standardization that removed reliance on individual memory, alerting systems that prioritized critical info under pressure, and monitoring programs that caught trends before they became failures. It relied on simulations that made crisis response instinctive, backed by a data culture that treated every incident as evidence worth examining. Far from one-time improvements, these changes were built directly into the operating model so they would last.

Your IT service desk can follow the same path. The five lessons in this e-book aren't just aspirational; they are operational, measurable, and proven on the ground at Zylker.

With the tools in place and the patterns thoroughly vetted under the most demanding conditions imaginable, these lessons are ready for any IT team prepared to apply them.



Aviation principle	ITIL practice	ServiceDesk Plus
Mandatory preflight checklists	Change enablement	Change Templates, change Workflows
ECAM prioritized alerting	Incident management	Incident Templates, SLA escalation, Notification Rules
GPWS proactive warning system	Problem management	Problem records, associate incidents, cluster analysis, and root cause analysis
Recurrent simulator training + revision controlled crew operating manuals	Knowledge management	Solutions module: Technician-only knowledge articles with version history, Tasks, Reply Templates
Black box analysis / FOQA / NTSB	Continual improvement	Custom Reports, Scheduled Reports, Dashboards

About ServiceDesk Plus

ServiceDesk Plus is the AI-driven unified service management platform from ManageEngine, the enterprise IT management division of Zoho Corporation. Built on industry-recommended ITSM best practices, ServiceDesk Plus comes packed with contextual IT and business integrations that help service desk teams better align with their organization's business objectives.

A wide set of AI-powered features, spanning the predictive, generative, conversational, and autonomous spectrum, ensures that service delivery teams apply the right technology to the right use case, and spend their time on work that needs human judgment. With native enterprise service management capabilities and unrestricted extensibility offered through low-code scripting, ServiceDeskPlus helps organizations design, deliver and support their business and IT services. It comes in three editions and is available in 37 different languages.

To learn more about ServiceDesk Plus and its features, please visit manageengine.com/products/service-desk/.

Here are five reasons why ServiceDesk Plus is trusted by some of the leading global enterprises

- High-value AI capabilities for IT and enterprise service management are not paywalled behind add-ons but included within your subscription.
- Powerful workflows orchestrate enterprise and IT services from end to end.
- From servers, networks, and switches to workstations and peripherals, it's a single system of record for your entire digital infrastructure.
- Platform capabilities power up ServiceDesk Plus to digitize and optimize workplace service delivery.
- ServiceDesk Plus integrates natively with every ManageEngine application and other third-party business apps .

Consult with our product experts to learn how ServiceDesk Plus can help you implement these five lessons in your service management operations. Reach out to us at hello@servicedeskplus.com.

About ManageEngine

ManageEngine is a division of Zoho Corporation that provides comprehensive on-premises and cloud-native IT and security operations management solutions for organizations and managed service providers globally. Established and emerging enterprises rely on ManageEngine's real-time IT management tools to ensure the optimal performance of their IT infrastructure, including networks, servers, applications, endpoints, and more. ManageEngine has 18 data centers, 20 offices and more than 200 channel partners worldwide to help organizations tightly align their business to IT. For more information, please visit the company site, follow the company blog, and get connected on LinkedIn, Facebook, Instagram, and X (formerly Twitter).

About the author



Saket Pasumarthy, a product marketing expert at ManageEngine ServiceDesk Plus, is an ITSM enthusiast and is fascinated with understanding the latest advancements in the IT space. Saket writes guides, articles, and blogs that help IT service management teams globally handle service management challenges. He also presents user education sessions in the ServiceDesk Plus Masterclass series. Saket spends his free time playing football and flying planes on a flight simulator.

Explore more from Saket Pasumarthy