



AI for IT operations and service desk

Using AI to resolve tickets faster, improve knowledge and reduce toil while keeping change control and security intact.

WORKS WITH THE ASSISTANTS TEAMS ALREADY USE



EDITION 46 / EMBER & CORAL

EXECUTIVE WORKING EDITION - 08 OCT 2026

46

AT A GLANCE

Using AI to resolve tickets faster, improve knowledge and reduce toil while keeping change control and security intact.

IT operations teams spend much of their time on repeatable work: sorting tickets, searching for the right fix, writing up incidents and preparing changes. AI can classify and route tickets, suggest answers from a knowledge base, draft and improve knowledge articles, summarize incidents and help reviewers read change requests. It can also do harm quickly if given broad access, because the same systems that resolve a ticket can reset a credential, change a configuration or expose data. This playbook covers where AI fits across ticket triage, knowledge, runbooks, incident summaries and change review, how to keep change control and security intact, how to measure results honestly and how to roll out in stages.

WHO IT IS FOR

IT service desk managers, infrastructure and operations leads, site reliability and platform teams, security and risk owners, and CIOs deciding how AI should work inside IT operations.

TIME TO USE

70 minutes to read; two workshops to apply

What you will have at the end

- A ranked list of service desk and operations use cases by value, risk and readiness
- A permission model that separates reading, drafting and acting
- A knowledge quality plan that makes suggestions trustworthy
- A metric set linking deflection and speed to resolution quality and security
- A 90-day plan from one queue to a decision to scale

Contents

01	What AI changes in IT operations	03
02	Where AI fits in IT operations, and where it does not	04
03	How it works, in plain language	06
04	Operating model, roles and decision rights	08
05	Risks, controls and security cautions	10
06	Measuring quality and value	12
07	Implementation guide	13
08	Pitfalls and a hypothetical scenario	15
09	Questions to ask before you buy or build	17
10	A first quarter that produces evidence	18
11	Terms used in this publication	19

EXECUTIVE BRIEF

What AI changes in IT operations

01



AI shortens the path from a vague ticket or alert to a good answer, and it must do so without weakening the controls that keep systems safe.

A service desk lives on volume and variety. Password and access requests, device problems, software questions, onboarding tasks and outages all arrive through email, chat, portal and phone, usually in the user's own words. Agents classify, search for a fix, ask for missing details, escalate and write notes. The knowledge they rely on is scattered across wikis, old tickets and individual memory. Operations teams have a similar pattern: alerts and incidents arrive, responders gather context from monitoring tools and chat, someone writes the summary, and a postmortem follows days later.

Language models fit this work well. They read free text, match a problem to similar past cases, draft a reply, summarize a long thread and turn a pile of notes into a clean article or timeline. Used well, they cut triage time, lift first-contact resolution, make knowledge easier to maintain and give responders context faster. They also reduce toil, the repetitive manual work that crowds out improvement.

The risk is different from most business uses. IT staff hold powerful access, and an AI connected to those tools can act with that power. A model that reads a ticket containing hidden instructions and then resets an account or opens a firewall rule is a security incident waiting to happen. A summary of an incident that states a cause with false confidence can mislead a postmortem. So the guiding principle in this playbook is least authority: start with reading and drafting, add narrow actions only with approval and logging, and keep change control as the gate for anything that alters production.

- Start with triage, knowledge and summaries, where a human sees the output before anything changes
- Treat ticket text, emails and logs as untrusted input that can contain hostile instructions
- Give the system the narrowest permissions that do the job, with separate identities and full logging

- Keep change management, approvals and segregation of duties in force for AI-proposed changes
- Fix knowledge quality first, because suggestions can never be better than the articles behind them



PLAIN TEST

Would you give a new contractor on their first day the access you are about to give the AI? If not, narrow the access or add an approval step.



WORKING QUESTION

Which three ticket categories consume the most agent time, and what fraction of each is solved by a documented fix?



WHERE IT FITS

Where AI fits in IT operations, and where it does not

02

AI fits best where the work is reading, matching and writing, and where a person approves any action that changes systems or access.

Sort use cases by two things: how much documented knowledge exists, and what the worst outcome of a wrong action is. Answering a how-do-I question from a good article is low risk and high volume. Drafting an incident summary is low risk if a responder reviews it. Resetting a password, granting access or restarting a production service is a different matter: the action may be reasonable, but the identity checks, approvals and logs matter more than the model.



USE CASE	FIT	REASON AND CONDITION
Ticket classification and routing	Strong	High volume; measurable; test against past tickets and keep a correction path
Suggested answers for agents from knowledge	Strong with grounding	Agent sees the article and source; reduces search time
Self-service answers to users	Strong for well-documented topics	Needs good articles and a clear hand-off to a person
Knowledge article drafting and cleanup	Strong	Turns resolved tickets into drafts; owner reviews before publishing
Runbook search and step guidance	Strong with review	Responder follows existing steps; AI does not improvise in production
Incident summaries and timelines	Strong with review	Saves writing time; facts checked against logs and chat
Change request review support	Moderate	Highlights missing fields and risk patterns; approvers decide
Access, password and privilege changes	Cautious	Only through existing identity workflows with verification and approval
Autonomous remediation in production	Narrow cases only	Pre-approved, reversible, low blast radius actions with logging
Security alert triage	Moderate	Assists analysts with context; verdicts and containment stay with security staff

Where it does not fit

- Making production changes outside the change process
- Granting or approving privileged access on the basis of a text request
- Being the sole judge of whether an alert is an attack
- Answering from undocumented tribal knowledge that does not exist anywhere
- Working with systems that lack logging, because you cannot reconstruct what happened

COMMON MISTAKE

Turning on a chatbot over a neglected wiki. It will confidently repeat outdated or contradictory instructions, and users will stop trusting both the bot and the desk.

WORKING QUESTION

For each use you are considering, what is the most damaging thing it could do if it were wrong or manipulated, and what stops it?

HOW IT WORKS

How it works, in plain language

03

The assistant reads the request, finds related knowledge and past cases, proposes a classification, answer or summary, and shows its sources.

For ticket triage, the system reads the subject, description and any attachments, then proposes a category, priority, affected service and assignment group, along with a short reason. Good implementations compare the ticket to past resolved tickets and to your category definitions rather than relying on general knowledge. The result is a suggestion with a confidence level. High-confidence routine cases can be routed automatically if you set that rule. Low-confidence cases go to a person.

For answers, retrieval is the core. The system searches your knowledge base, runbooks and resolved tickets for relevant passages, then asks the model to write an answer based on those passages with links. If no relevant passage is found, the right behavior is to say so and route to an agent. The quality of the answer depends heavily on article quality: duplicated, outdated or contradictory articles lead to confused suggestions.

01 Receive

Capture the ticket, alert or chat with user, device, service and history context.

02 Understand

Classify the request, extract key details and identify what is missing.

03 Retrieve

Find matching articles, runbook steps and similar past tickets, with their dates and owners.

04 Propose

Draft a reply, a set of steps or a summary, citing sources.

05 Approve and act

An agent or responder confirms. Any action on systems follows existing approval and logging rules.

06 Learn

Capture outcomes and corrections to improve knowledge and tests.



Three levels of capability

LEVEL	WHAT THE SYSTEM CAN DO	CONTROL NEEDED
Read and suggest	Classify, search, draft, summarize	Data access limits, source citation, logging
Prepare actions	Fill forms, draft change requests, queue steps for approval	Human approval before submission or execution
Act within limits	Run pre-approved, reversible actions such as sending a standard reply or unlocking a verified account	Narrow permissions, verification, rate limits, logs, kill switch



WORKING QUESTION

At which of the three levels does your first use case sit, and is the control column actually in place for that level?



OPERATING MODEL

Operating model, roles and decision rights

04

Put AI inside your existing ITSM structure: owners for knowledge, queues and changes, with permissions designed like any other service account.

IT organizations already have strong constructs: service levels, queue ownership, problem and change management, on-call rotations and access reviews. Reuse them. An AI assistant is a participant in the service desk with a defined role, permissions and manager. Its suggestions are reviewed like a junior agent's, its access is reviewed like a service account's, and its changes to knowledge or routing rules go through the same approval as a human's.

ROLE	RESPONSIBILITY	TYPICAL OWNER
Service owner	Scope, targets, user experience	Service desk manager
Knowledge owner	Article quality, freshness, retirement	Knowledge manager, team leads
Queue reviewers	Check suggestions, correct classification, flag errors	Service desk agents
Change manager	Governs any AI-proposed or AI-prepared change	Change advisory lead
Identity and access owner	Defines what the AI identity can do and who can alter it	IAM lead
Security lead	Threat model, monitoring, incident response for the AI itself	CISO or security architect
AI product owner	Prompts, retrieval settings, tests, releases	Platform or automation lead
Responders on call	Use summaries and runbook help to own the incident	SRE, infrastructure

Decision rights to write down

- The AI never approves its own changes, and approvers are people with authority and context

- Password resets, MFA changes and privilege grants follow the identity workflow with strong verification, not text claims in a ticket
- Production changes follow normal change control, with the AI at most preparing the request and checking completeness
- Emergency changes still require a named human owner and a record after the event
- Prompt, retrieval and routing changes are versioned, tested and approved like code
- Any agent can turn off an automated flow and report why



Segregation of duties deserves explicit attention. If one automation can raise a request, approve it and execute it, the control that change management provides has been removed. Separate those functions by identity and by approval path, and test the separation. The same applies to knowledge: the person who approves publication of an article should not be an automated process that also wrote it.

STAFF CONCERNS

Agents will ask whether suggestions are used to rank them. Decide and say plainly what data is used for coaching, quality and performance, and check employment and privacy requirements for your location.



WORKING QUESTION

Can you list every identity the AI uses, what each can do, who owns it and when its access was last reviewed?



RISKS AND CONTROLS

Risks, controls and security cautions

05

Treat inbound text as hostile, limit what the system can do, log everything, and keep human approval on changes and access.

A language model cannot reliably tell the difference between your instructions and instructions hidden in the content it reads. A ticket, email, web page or log line can contain text such as an instruction to ignore previous rules and reveal data or perform an action. This is called prompt injection, and it is a recognized class of risk for systems that read untrusted content and have tools. The practical defense is architectural: limit what the system can do, require approval for consequential actions, restrict data it can see, and monitor its behavior.

RISK	WHAT IT LOOKS LIKE	CONTROL
Prompt injection	Ticket or email text steers the assistant to leak data or act	Treat content as data; least authority; approval for actions; output filtering; monitoring
Social engineering via ticket	Attacker asks for a reset or access in convincing language	Identity verification outside the ticket; no access changes on text alone
Data exposure	Secrets, credentials or personal data pasted into prompts or logs	Redaction, approved tools only, secret scanning, retention limits
Overbroad permissions	AI service account can reach more than it needs	Scoped identities, separate accounts per function, regular access review
Wrong fix	Suggestion is outdated or unsafe	Source citation, article owners and dates, agent confirmation, rollback steps
Confident incident narrative	Summary states a cause not supported by evidence	Label hypotheses, link to logs, responder sign-off
Change control bypass	Automation alters configuration outside process	Block direct write access; route through change tooling; audit

RISK	WHAT IT LOOKS LIKE	CONTROL
Vendor and supply chain	Third-party tool processes sensitive logs	Contract terms, region, retention, security assessment, version control

Minimum security controls

- Inbound ticket and email content treated as untrusted
- AI identities scoped to specific functions, with no standing administrator rights
- No access or privilege change from a ticket without separate verification and approval
- Secrets and personal data redacted before processing where possible
- Full logs of prompts, retrieved sources, outputs and actions, with retention set
- Kill switch tested, with a named owner
- Vendor security review and data terms completed
- Red team style tests, including injection attempts, before launch and after major changes

FRAMEWORKS

Security and AI risk frameworks published by standards bodies and regulators can structure these controls. Requirements differ by sector and region and change over time; check current requirements and align with your existing security program.

WORKING QUESTION

If an attacker put hostile text into a ticket tomorrow, what is the worst thing your AI could do with its current permissions?

MEASURING VALUE

Measuring quality and value

06

Report speed, resolution quality and security signals together, from a baseline captured before the pilot.

IT service management already provides data: ticket volumes, time to first response, time to resolution, reopen rates, user satisfaction and backlog. Capture a baseline for at least a month on the queues in scope, noting patterns such as month-end, onboarding waves and release weeks. Define terms carefully. Deflection, for example, means a user got an answer without opening a ticket and did not come back with the same problem, not merely that a chat session ended.

METRIC	DEFINITION	WHY IT MATTERS
Triage accuracy	Share of suggested categories and priorities matching reviewer verdicts	Routing quality
Time to first meaningful response	Ticket created to first action that helps	User experience
First-contact resolution	Tickets resolved on first contact without reopen	Quality, not just speed
Reopen rate	Resolved tickets reopened within set period	Catches wrong fixes
Verified deflection	Self-service answers with no repeat contact within a set period	Honest self-service value
Suggestion acceptance and edit rate	Share of suggestions used as-is, edited or rejected, with reasons	Knowledge and prompt quality
Knowledge freshness	Share of articles reviewed within policy	Foundation of answer quality
Incident summary effort	Responder minutes spent on write-ups and postmortem drafts	Toil reduction
Security events involving AI	Injection attempts, blocked actions, access anomalies	Control effectiveness

Illustrative baseline

These figures are illustrative only. Suppose a desk handles 5,000 tickets a month and agents spend an average of 4 minutes on classification and searching before they can act, about 333 hours. If assisted triage and suggested articles cut that to 2.5 minutes without raising reopen rates or lowering satisfaction, the team would recover about 125 hours a month. If reopen rates climbed because the suggestions pointed to outdated fixes, the net gain would shrink. Decide in advance how recovered time will be used, such as proactive work on repeat problems.

- Sample and audit suggestions weekly, including ones agents accepted without edits
- Break results out by category, since password questions behave very differently from application faults
- Track knowledge gaps found by the assistant: questions with no good article are a work queue for the knowledge owner
- Count the cost of licenses, integration, review time and knowledge maintenance
- Report ranges and sample sizes rather than single figures

WORKING QUESTION

Which metric would tell you first that suggestions were leading agents to wrong fixes, and who looks at it each week?

IMPLEMENTATION

Implementation guide

07

Fix the knowledge, start with suggestions, add automation in narrow steps, and keep change control in place throughout.

01 Pick one queue and one job

Choose a high-volume queue, such as end-user software and access questions, and a first job such as triage suggestions plus article retrieval for agents.

R

02 Clean the knowledge

Review the top 50 to 100 articles used by that queue. Remove duplicates, mark owners and dates, retire stale items and fill the most common gaps from resolved tickets.

03 Build the test set

Collect 300 to 500 resolved tickets with correct categories and fixes. Include ambiguous, multi-issue and badly written tickets. Score triage and retrieval separately.

04 Design permissions

Define the AI identity, what it can read, what it can draft and what it cannot do. Document it and have security review it.

05 Pilot with agents

Release suggestions to a small group. Require them to confirm and to mark bad suggestions with a reason. Review weekly.

06 Extend carefully

Add self-service for well-documented topics, then incident summaries, then change-request checks. Consider narrow automation only for reversible, low-risk actions with approval.

R

K

Knowledge as an operating practice

Knowledge quality is the quiet determinant of success. Make article ownership explicit, set a review cadence, and tie knowledge updates to ticket closure: when an agent solves a new problem, the system drafts an article and the agent or owner reviews it. Retire articles on a schedule. Add a feedback button to every suggestion so poor guidance is found within days, not quarters.

Readiness checklist

- Named service owner, knowledge owner, change manager and security lead
- Top articles cleaned, owned and dated
- Test set scored for triage and retrieval
- AI identity and permissions documented and reviewed
- Injection and misuse tests completed
- Logging and retention configured
- Fallback to manual operation tested
- Staff briefed on purpose, data use and how to flag problems

R

RUNBOOKS

Use AI to help responders find and follow existing runbooks and to draft new ones from resolved incidents. Do not let it invent steps during an outage. Responders execute steps they have verified.

**WORKING QUESTION**

Which 50 articles would you fix first, and who owns each one by name?

PITFALLS

Pitfalls and a hypothetical scenario

08



Problems come from stale knowledge, excess permissions and unchecked automation rather than from model choice.



- Launching self-service before articles are accurate and owned
- Giving the assistant a broad administrator account for convenience
- Letting reviewers accept suggestions without opening the cited article
- Counting chat sessions that end as deflections
- Allowing automation to create, approve and implement changes
- Pasting logs with credentials or personal data into unapproved tools
- Treating the first successful pilot queue as proof for security alerts or production remediation



HYPOTHETICAL SCENARIO (ILLUSTRATIVE, NOT A REAL CASE)

A service desk pilots an assistant that can unlock accounts when a ticket says the user is locked out. A tester submits a ticket claiming to be a manager and asks for an unlock on behalf of an employee, with a line instructing the assistant to skip verification. Because unlocks run through the identity workflow, which requires a verified callback, the request is held. The team records the test, adds injection cases to the suite and removes direct unlock permission from the assistant.

The example shows the value of testing hostile inputs before launch and of putting verification in the identity workflow rather than in the assistant. Controls that depend on the model behaving well are fragile. Controls that live in systems that enforce them regardless of what the model says are durable.

WARNING SIGN	LIKELY CAUSE	FIRST RESPONSE
Reopen rate rising	Outdated articles or weak retrieval	Audit reopened tickets, fix sources, tighten citations
Agents ignore suggestions	Low relevance or poor interface	Interview agents, review misses, adjust
Unusual AI account activity	Misuse or injection	Pause automation, investigate logs, review permissions
Incident summaries disputed	Facts unsupported by logs	Require links to evidence and responder sign-off

WORKING QUESTION

Which control in your design still works if the model is fooled completely?

BUYING QUESTIONS

Questions to ask before you buy or build

09



Test on your tickets and knowledge, probe the security model, and be clear about who owns change control.

Questions for any supplier

- Will you run a supervised trial on our own tickets and knowledge and share triage and retrieval accuracy by category?
- How does each suggestion show its source, owner and date?
- What permissions does the product need in our ITSM, identity and monitoring tools, and can they be scoped to read-only first?
- How does the product defend against hostile instructions in tickets, emails and logs, and can we see test results?
- What actions can it take without approval, and can we turn each one off?
- What is logged, for how long, and can we export it to our SIEM and audit tools?
- Where is our data processed, who can see it and is it used to train models for others?
- How are model and version changes announced, and can we pin and re-test?
- What is the full cost, including integration, knowledge work, review time and usage-based fees?



OPTION	FITS WHEN	WATCH FOR
AI features in your ITSM platform	You want quick adoption and one data location	Customization limits, pricing changes, roadmap dependence
Specialist service desk or AIOps tool	Large volumes or complex estates	Integration depth, data terms, overlap with existing tools
General AI platform you can configure	Varied use cases and internal skills	Own testing, security and maintenance
Custom build	Distinctive needs and engineering capacity	Ongoing upkeep and security responsibility



Whatever the route, keep the durable assets in your hands: knowledge base, test sets, permission model, logs and change process. They let you compare suppliers fairly and switch if needed. Halden & Company helps IT teams choose the first queue, clean the knowledge, design permissions and build the test and control framework. Free assessments are available at halden.company/assessments, and the team can be reached at halden.company/contact.

WORKING QUESTION

If you had to switch AI vendors in a quarter, which of your assets would make that easy and which would make it painful?

30 / 60 / 90

A first quarter that produces evidence

10

Fix knowledge and permissions first, prove suggestions on one queue, then widen with change control intact.

First 30 days: ground it

- Choose one queue and one first job, and name service, knowledge, change and security owners
- Capture a one-month baseline for response, resolution, reopen and satisfaction
- Clean and assign owners to the top 50 to 100 articles
- Define the AI identity and its permissions, starting read-only
- Build a 300 to 500 ticket test set with correct categories and fixes

Days 31 to 60: test it

- Score triage and retrieval separately and fix the largest misses
- Run injection and misuse tests with security
- Release suggestions to a small group of agents with feedback on every item
- Set up logging, retention and a tested kill switch
- Review weekly: accepted, edited and rejected suggestions and reopen rates



Days 61 to 90: decide

- Compare results with baseline, including reopen rates and security events
- Decide to scale, change or stop, and record the reasons
- Add incident summaries or self-service for well-documented topics
- Document stop conditions and access review dates
- Plan the next queue using the same test, permission and review pattern



GLOSSARY

Terms used in this publication

11

TERM	MEANING
Triage	Classifying and prioritizing a ticket and routing it to the right team.
Knowledge base	The collection of articles and runbooks that agents and users rely on to resolve issues.
Runbook	A documented set of steps for handling a known operational task or incident.
Deflection	A user resolving a problem through self-service without needing a ticket, and not returning with the same issue.
Change control	The process that reviews, approves and records changes to production systems.
Segregation of duties	Keeping requesting, approving and executing a change in different hands to reduce error and abuse.



TERM	MEANING
Least authority	Giving a system only the permissions it needs, so mistakes or misuse cannot do wide harm.
Prompt injection	Hostile instructions hidden in content a model reads, intended to change its behavior.
Toil	Repetitive, manual operational work that does not improve the system and grows with scale.
Postmortem	A blameless review after an incident that records what happened, why and what will change.
Retrieval	Finding relevant articles or past tickets so the model answers from approved sources.
Reopen rate	The share of resolved tickets that the user reopens because the fix did not work.

A note on use

This playbook is general guidance, not legal or security certification advice. Security, privacy and employment requirements vary by region and sector and change over time; check current requirements.

TAKE THE NEXT STEP

Take the free Halden assessment at halden.company/assessments, or talk to us at halden.company/contact about turning this into a plan for your organization.