

MANAGEMENT REVIEW

Management without theatre.

OPERATIONS

THE HUMAN OVERRIDE: WHEN AUTOMATION IS WRONG

Who takes responsibility? Six chapters on trust, alarms, system interruption and human judgment across everyday operations: the green dashboard, the ironies of automation, the base rate of an alert, what the driver knows, the night without the system, and the override that someone must own.



74

studies met the inclusion criteria in a systematic review of automation bias, out of 13,821 retrieved papers.

GODDARD ET AL., 2012

INSIDE

COVER STORY

The green dashboard, the unfinished job

PAGE 05

THE EVIDENCE

What the reviews actually found

PAGE 17

FIELD TOOL

The Human Override Card

PAGE 27

EDITORIAL · VOLUME 3

The decision after the signal

A screen may offer an answer before the person closest to the work can explain the question.

An automated route, a scan event, a red dashboard indicator and a hotel status are not interchangeable. Each is a record produced for a particular purpose. At its best, the system compresses routine effort and helps teams coordinate. At its worst, its speed encourages the organisation to skip a slower step: establishing what happened and who is authorised to act. This volume follows one practical question across logistics, dispatching and night-shift hospitality: what happens when an automated output appears complete but the work is not? The reader will meet constructed scenes, published studies and documented work categories. They are not the same kind of evidence. An invented case can clarify a mechanism; it cannot prove that a named workplace suffered that failure. No guest, driver or employer is identified, and no private system procedure is disclosed.

Stiven Janaqi EDITOR · STIVEN CATALYST

HOW EVERY CHAPTER IS BUILT

1 The living scene

A shift, a ramp, a desk at 03:00: one piece of work followed as it actually happens, with the hand-off where it breaks. Scenes are constructed from real categories of work and say so.

2 The conceptual and scientific breakdown

The mechanism behind the scene, named with the model that explains it, and the research that tests it: the study, its population, its year and its limits, before it is applied anywhere else.

3 The operating protocol for action

What a working team can try on Monday: fields, questions, a cadence and a count, each with an owner, a next step and a return of information to the people who do the work.

THE READING CONTRACT

Field

Only verified background from real roles; no invented personal incidents.

Science

Researchers, dates, populations and limits appear beside every claim.

Action

An authorised next step, an owner and an observable result.

CONTENTS

The chapters,
in three layers

CHAPTER I The green dashboard, the unfinished job			05
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
The green dashboard, the unfinished job	05 From recorded to resolved	06 Verify before you escalate	07
CHAPTER II The irony of automation			08
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
The operator who was watching	08 Bainbridge's uncomfortable irony	09 Keep people inside the loop	10
CHAPTER III When every alarm is red			11
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
When every alarm is red	11 The base-rate problem	12 A better alarm review	14
	Two numbers, one design	13	
CHAPTER IV What the driver knows, the system does not			15
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
The route knows less than the driver	15 Use, misuse, disuse, abuse	16 Make local knowledge transferable	18
	What reviews actually found	17	
CHAPTER V 03:00: when the system is unavailable			19
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
When the training stopped	19 Out of the loop is not out of duty	20 Continuity without improvisation	22
	Confidence is not control	21	
CHAPTER VI The responsible override			23
1 SCENE	2 BREAKDOWN	3 PROTOCOL	
The decision between two systems	23 What exactly is automated?	24 Measure the decision system	26
	The responsible override model	25	

ALSO IN THIS VOLUME

04 RESEARCH NOTE
What this evidence can tell us

27 FIELD TOOL
The Human Override Card

28 CLOSING ESSAY
The system cannot inherit the responsibility

29 METHOD
Evidence before conviction

30 SOURCES
Sources and method

RESEARCH NOTE

What this evidence can tell us

Research on pilots, control-room operators and clinical decision support offers mechanisms, not local performance estimates.

Bainbridge (1983) examined the paradoxes of industrial automation. Endsley and Kiris (1995) studied performance after an automated navigation aid failed. Parasuraman and Riley (1997) distinguished reliance, over-reliance and the rejection of useful automation. Goddard and colleagues (2012) reviewed empirical studies across domains; Romeo and Conti (online 2025) focused on modern human-AI collaboration. Their settings differ from a loading ramp, an office dispatch desk or a night hotel reception. Transfer is defensible only at the level of a mechanism to test: degraded monitoring, stale records, over-trust, ignored alarms, missing recovery knowledge or unclear authority. It is not defensible to import a published error percentage as though it came from the author's workplaces.

01

Observed

Published study, measured outcome, population and year.

02

Inferred

A plausible operational risk, stated as a hypothesis.

03

Proposed

A practice or metric awaiting local validation.

OUR READING

The purpose of research here is to discipline what we claim, not to make a constructed scene look verified.

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995; Raja Parasuraman & Victor Riley, *Human Factors* 39(2), 1997; Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, *Journal of the American Medical Informatics Association* 19(1), 2012; Giuseppe Romeo & Daniela Conti, *AI & Society* (online 2025, volume 41, 2026), 2025

Full references on the sources page; every study is cited at its DOI and read in its published version or abstract.

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER I · THE GREEN DASHBOARD, THE UNFINISHED JOB

The green dashboard, the unfinished job

A screen can be accurate about the field it records while the overall service remains incomplete.

HYPOTHETICAL COMPOSITE, NOT A RECORDED INCIDENT

The dashboard marks a departure as complete. A vehicle has left the loading area; the timestamp is legitimate. But an unresolved discrepancy in the load had been discussed moments earlier. One colleague recorded an exception in one place, another sees only the successful departure status. The display is green because the departure occurred. The customer-facing process is not necessarily green.

The temptation is to argue about which employee made the wrong entry. A better investigation starts with the exact statement the screen makes. Does “departed” mean the vehicle moved? That all items were scanned? That the load was verified complete? Those are different claims. A status becomes misleading when its readers silently give it more meaning than its definition supports. Mixed trolleys, missing bags and loading discipline are documented categories from the author's last-mile background. This exact vehicle, sequence and discrepancy are invented.

OUR READING

A true event can still be an incomplete explanation.

No actual incident is reconstructed; the essay named below tells the same shape from the dispatch desk.

[More in the essay](#)

The report a machine wrote

CHAPTER I · THE GREEN DASHBOARD, THE UNFINISHED JOB

From recorded to resolved

Three meanings of “done” can quietly collapse into one colour.

01

Recorded

The system captured an event.

02

Verified

The event and the required condition were checked.

03

Resolved

The customer-facing obligation was fulfilled.

The categories above are an editorial diagnostic model, not a three-stage taxonomy from one published study. They help a manager ask what the system truly claims. A scan may be evidence of a transaction; a status may record that an operation passed a designated point. Neither alone proves that every requirement downstream has been satisfied. Parasuraman, Sheridan and Wickens (2000) distinguish automation of information acquisition, information analysis, decision selection and action implementation. The distinction matters because the same dashboard may combine measured facts with inferred assessments or recommended actions. Operators deserve to know where recording stops and inference begins.

01 **Observation**

What was sensed, entered or timestamped?

02 **Interpretation**

What rule transformed it into a status?

03 **Decision**

What action is warranted, by whom, with what authority?

OUR READING

Before disputing the colour, define the condition it is supposed to represent.

Sources: Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, IEEE Transactions on Systems, Man, and Cybernetics A 30(3), 2000; John D. Lee & Katrina A. See, Human Factors 46(1), 2004

The four classes of automated function are Parasuraman, Sheridan and Wickens (2000); appropriate reliance after Lee and See (2004). Recorded, verified, resolved is editorial.

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER I · THE GREEN DASHBOARD, THE UNFINISHED JOB

Verify before you escalate

One signal, one verified condition, one accountable next decision.

1

Name the signal

Record the indicator, source, timestamp and its official definition.

2

Check freshness

Establish whether the reading is live, delayed, aggregated or final.

3

Compare the work

Use permitted independent evidence from the person or process closest to the outcome.

4

Classify the mismatch

Is it a recording gap, a business-rule problem or an actual process exception?

5

Assign authority

State who may decide on the correction or escalation under current policy.

6

Return the result

Document what was confirmed and what the team needs to know next.

A status mismatch is not permission to bypass a safety, privacy or compliance control: an immediate threat follows the emergency path, a routine discrepancy the local review route. Verification exists to produce a better decision, not an unofficial second system.

OUR READING

A manager should be able to say what the number means, when it was true, and what it does not establish.

Sources: Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, IEEE Transactions on Systems, Man, and Cybernetics A 30(3), 2000; John D. Lee & Katrina A. See, Human Factors 46(1), 2004

Editorial protocol after the function classes of Parasuraman et al. (2000) and the reliance framework of Lee and See (2004); not a procedure certified for a named employer.

CHAPTER II · THE IRONY OF AUTOMATION

The operator who was watching

Being responsible for an automated process is not the same as remaining prepared to operate it.

HYPOTHETICAL ILLUSTRATION, NO CLAIMED WORKPLACE FAILURE

For hours, the plan proceeds routinely. Stops are arranged, exceptions are displayed and the person at the desk deals with the cases that the software highlights. Then an unusual condition appears. The system has not supplied a useful next move. The operator must reconstruct what has happened, what the route depends on and which changes remain permitted.

It is tempting to call this a simple need for better technical training. But the difficult part may be the transition itself. Routine automation invites people to monitor results rather than build and continually update the whole operational picture. When the software cannot complete the task, the missing picture has to be rebuilt under time pressure. This is a constructed scene. The author has worked in last-mile dispatch and operational coordination, but no particular route-planning outage or mistaken takeover is asserted here.

OUR READING

The most difficult human task may be the one the system leaves for last.

A constructed illustration of a general risk; its significance rests on Bainbridge (1983) and Endsley and Kiris (1995), read in the settings where they were written.

[More in the essay](#)

The system is faster than you

CHAPTER II · THE IRONY OF AUTOMATION

Bainbridge's uncomfortable irony

Automation can remove routine activity while leaving people responsible for the exceptional cases.

In a short 1983 paper, Lisanne Bainbridge challenged the belief that technical sophistication automatically eliminates human-factor problems. Industrial automation may expand the difficulties left to the operator, especially when abnormal conditions need diagnosis. The point is not that automation is undesirable; it is that the remaining human role is changed by the way automation is designed.

01

Routine relieved

Fewer repetitive manual decisions.

02

Context harder

Less direct participation in the work that creates familiarity.

03

Exception retained

People still need to interpret unusual, consequential conditions.

A logistics route, a scan trail and a hotel report are not industrial control loops of the kind Bainbridge analysed. The transfer is narrower: if a process expects a person to handle anomalies, it must also preserve that person's access to context, authority, practice and usable feedback. A named owner without those conditions may carry responsibility without control. The most dangerous design question is therefore not “What can we automate?” but “What will the person still need to understand when it stops helping?”

OUR READING

Removing a routine task does not remove the need to learn the system.

Source: Lisanne Bainbridge, *Automatica* 19(6), 1983

Bainbridge, “Ironies of Automation”, *Automatica* 19(6), 1983: a conceptual paper on industrial process control, not an experiment; the three cards are an editorial summary.

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER II · THE IRONY OF AUTOMATION

Keep people inside the loop

The response to a rare abnormal event should be rehearsed before it arrives. In Endsley and Kiris's navigation experiment (1995), lower situation awareness went with slower decisions after the expert system failed; the level of control moderated the loss. That does not measure a hotel or a dispatch desk, but it makes the design concern concrete.

1

Explain the boundaries

Show what the system handles, what it cannot infer and who owns exceptions.

2

Practice independent orientation

Periodically reconstruct the workflow from permitted evidence, without assuming the recommendation is correct.

3

Rehearse safe escalation

Know how to get help, pause an unsafe action and request an authorised decision.

4

Check recovery, not memory

Use scenario-based training and confirm the operator can describe the next action.

5

Update after learning

When an unfamiliar failure appears, amend approved training after a documented review.

OUR READING

A trained operator knows not only how to use the tool, but how to recognise when the tool is no longer enough.

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995

Editorial training protocol after Bainbridge (1983) and Endsley and Kiris (1995), whose participants worked a laboratory navigation task, not a dispatch desk.

CHAPTER III · WHEN EVERY ALARM IS RED

When every alarm is red

A red indicator deserves attention. It is not, by itself, a diagnosis.

HYPOTHETICAL DISPATCH SCENE, NO RECORDED ALARM EVENT

The display is full of red. Some routes are late to a threshold, some readings are old and others reflect problems already being handled. The coordinator does not know which condition is new, which is serious and which is no longer current. Every signal competes for the same limited attention.

A manager may call this poor discipline: why is the team not responding to each red row? Yet an effective response depends on more than colour. The alert must represent a defined condition, arrive at a useful time, indicate severity and give someone authority to act. Otherwise an alert system may become a second source of work: investigating what the system meant before investigating what happened. There is no documented case in the author's remote-dispatch role of a specific red alert that disappeared without intervention. This scene illustrates the decision problem; it does not present invented alert statistics as experience.

OUR READING

If every change demands the same attention, the screen has not decided what matters.

Constructed scene from the category of remote dispatch work; the essay named below tells the same morning from the author's desk.

[More in the essay](#)

[When the dashboard turns red before you](#)

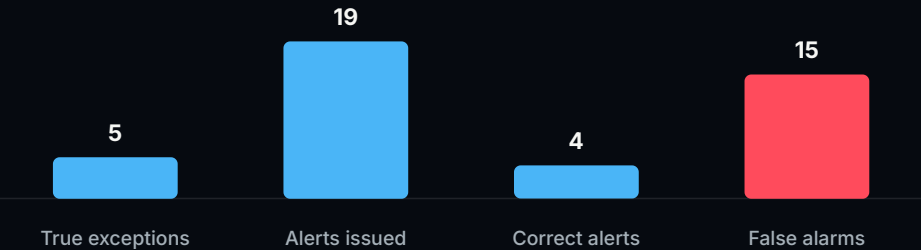
CHAPTER III · WHEN EVERY ALARM IS RED

The base-rate problem

A system can detect many genuine exceptions and still produce mostly unhelpful alerts. Assume five of 100 cases really require attention. The system identifies four of those five, misses one, and incorrectly alarms for 15 of the remaining 95. It issues 19 alerts: four correct and 15 false positives.

FULLY HYPOTHETICAL EXAMPLE: 100 CASES, NOT COMPANY DATA

ONE HUNDRED INVENTED CASES, FIVE OF THEM TRUE EXCEPTIONS



Only $4 \div 19 = 21.1\%$ of the alert messages correspond to real exceptions in this example. The system detected $4 \div 5 = 80\%$ of true exceptions. Both values describe the same hypothetical design, from different angles. Parasuraman and Riley (1997) note that false alarms can promote disuse of automation. The example makes their base-rate warning intuitive: alert usefulness depends on how often the condition occurs, not only on how well the software detects it. No workplace false-alarm rate is estimated here.

OUR READING

Count the alarms that need action, not only the alarms the system can produce.

Source: Raja Parasuraman & Victor Riley, Human Factors 39(2), 1997

The 100 cases are invented so the arithmetic can be checked; disuse after false alarms is Parasuraman and Riley (1997).

CHAPTER III · WHEN EVERY ALARM IS RED

Two numbers, one design

The same alert design looks good or bad depending on which denominator is used. Precision answers the question of the person who receives the alerts; sensitivity answers the question of the person who fears the missed case. A review of alarms needs both, and the cost of each kind of error.

FULLY HYPOTHETICAL EXAMPLE: 100 CASES, NOT COMPANY DATA

THE INVENTED 100 CASES, READ FROM TWO SIDES

Alerts that were real (precision)



True exceptions caught (sensitivity)



Alerts that were false



21.1% positive predictive value: $4 \text{ correct alerts} \div 19 \text{ alerts issued}$

A team that receives 19 alerts and finds four real cases will, over weeks, learn to wait before reacting. That learning is rational, and it is the mechanism Parasuraman and Riley call disuse. The remedy is not to tell people to trust the screen more. It is to change the condition the alert represents, its timing or its threshold, under control, and to count the result.

Source: Raja Parasuraman & Victor Riley, Human Factors 39(2), 1997

The percentages follow from the invented counts on the previous page; precision and sensitivity are standard definitions, not figures from any study.

CHAPTER III · WHEN EVERY ALARM IS RED

A better alarm review

Treat alert design as a quality problem with measurable costs on both sides.

- 1 Define the outcome**
Which condition really needs a decision, not merely a colour?
- 2 Capture true and false signals**
Use independently verified cases with a clearly defined observation window.
- 3 Measure missed exceptions**
Silence can be more consequential than noise; define misses explicitly.
- 4 Separate urgency from inconvenience**
A safety-critical condition cannot be grouped with a routine timing exception.
- 5 Design a response**
Specify owner, maximum review delay and safe escalation path.
- 6 Test after changes**
Compare downstream outcomes, not just the count of red rows.

Never improve the dashboard by raising thresholds just to make the screen greener. A threshold change is a controlled change to risk detection: it needs permission, a documented rationale and a review of both false alarms and missed cases, with the time the data was last refreshed.

OUR READING

An alarm is good when it helps the right person take the right action in time.

Sources: Raja Parasuraman & Victor Riley, Human Factors 39(2), 1997; John D. Lee & Katrina A. See, Human Factors 46(1), 2004

Editorial review protocol after Parasuraman and Riley (1997) on disuse and Lee and See (2004) on appropriate reliance; no threshold tested in the author's workplaces.

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER IV · WHAT THE DRIVER KNOWS, THE SYSTEM DOES NOT

The route knows less than the driver

Software knows its inputs. The worker encounters what is happening now.

COMPOSITE ILLUSTRATION; THE EXACT EVENT DID NOT OCCUR AS REPORTED

A route appears reasonable on the screen. A person who has delivered in the area before notices that part of the plan depends on a condition that is no longer reliable. The person asks whether the route can be reviewed. At the same time, another system has a reassuring scan history, yet a physical load still needs checking. The coordinator faces two kinds of evidence: a record that may be formally correct and local knowledge that is not yet verified.

The wrong choice is to treat the software as automatically authoritative or to accept every human objection on trust. The working choice is to examine the claim, preserve the evidence and follow the authorised procedure for changes. The employee's observation has value even when it turns out to be mistaken: it identifies a condition worth testing. The author's verified background includes coordination of drivers, missing bags, mixed trolleys and scanning rules. The specific route disagreement, timing and outcome here are illustrative, not remembered events.

OUR READING

A route can be mathematically consistent and operationally incomplete.

No driver, route or employer is identified; the essay named below tells the same disagreement from the dispatch side.

More in the essay

The tour the driver does not understand

CHAPTER IV · WHAT THE DRIVER KNOWS, THE SYSTEM DOES NOT

Use, misuse, disuse, abuse

The problem is not simply to make people trust automation more.

01	Use	Deliberately engage or disengage automation when appropriate.
02	Misuse	Rely too strongly on its output, reducing checking or judgment.
03	Disuse	Neglect a useful system, potentially after repeated false alarms.
04	Abuse	Design or impose automation without sufficient regard for human consequences.

These terms come from Parasuraman and Riley (1997). Their model refuses an easy binary between “believe the computer” and “believe the worker”. The intended outcome is appropriate reliance. Lee and See (2004) emphasise that trust is shaped by context and by the system's characteristics. A trustworthy design helps people calibrate what the tool is actually good at, and under which circumstances. When a driver flags an unusual road condition or a colleague questions a scan event, the response must not depend on seniority or confidence alone. What independent evidence is available? Does the input reflect the current situation? Who has authority to change the plan? Can the observation improve the system for the next user?

OUR READING

The goal is calibrated trust: neither blind reliance nor reflexive rejection.

Sources: Raja Parasuraman & Victor Riley, Human Factors 39(2), 1997; John D. Lee & Katrina A. See, Human Factors 46(1), 2004

The four terms are Parasuraman and Riley's; the four questions at the end are editorial.

CHAPTER IV · WHAT THE DRIVER KNOWS, THE SYSTEM DOES NOT

What reviews actually found

Automation bias has an empirical literature, but the evidence samples must stay attached to their contexts.

13,821

papers retrieved

74

studies met the criteria

35

peer-reviewed studies reviewed

2015–25

publication window assessed

Goddard, Roudsari and Wyatt (2012) searched several fields, with a focus on clinical decision support, and kept 74 studies from 13,821 retrieved papers. Trust and confidence, task experience, workload, task complexity and time constraints mediated automation bias; training, user accountability and design choices could mitigate it. Those counts are papers in a search, not the frequency of bad decisions at work.

Romeo and Conti (online 2025) examined 35 peer-reviewed studies of human-AI collaboration from January 2015 to April 2025 and warn that explanations alone do not reliably prevent over-reliance; prompting independent verification still matters. The two reviews use different inclusion rules and periods and cannot be compared as if they counted the same phenomenon over time.

OUR READING

A bigger literature gives a stronger reason to test a mechanism, not permission to invent local results.

Sources: Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, *Journal of the American Medical Informatics Association* 19(1), 2012; Giuseppe Romeo & Daniela Conti, *AI & Society* (online 2025, volume 41, 2026), 2025

The counts are from the two abstracts: "Of 13 821 retrieved papers, 74 met the inclusion criteria" (JAMIA 19(1), 2012) and "35 peer-reviewed studies ... published between January 2015 and April 2025" (AI & Society, online 2025).

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER IV · WHAT THE DRIVER KNOWS, THE SYSTEM DOES NOT

Make local knowledge transferable

Do not ask workers to choose between silent improvisation and blind obedience.

1

Accept a report

An easy, approved way to flag a route, status or scan reading that may be wrong.

2

Describe the observation

Condition, time, impact, and whether it can be checked independently.

3

Protect the standard

No change to routes, data or compliance steps without delegated authority.

4

Verify jointly

The original record and the real condition, with the relevant team.

5

Decide and document

Update an authorised source, or explain why the current rule stands.

6

Tell back

Frontline users hear what was confirmed and what is still unknown.

What a street, a ramp or a desk teaches should not stay in one person's memory; nor should an unverified anecdote become a new rule. A feedback channel works when it separates observation from approval and returns a reasoned decision to the people affected.

OUR READING

A system improves when a correction becomes shared knowledge, not a private workaround.

Sources: Raja Parasuraman & Victor Riley, Human Factors 39(2), 1997; John D. Lee & Katrina A. See, Human Factors 46(1), 2004

Editorial protocol after Parasuraman and Riley (1997) and Lee and See (2004); the site's Shift Handover tool carries the fields of the last two steps.

CHAPTER V · 03:00: WHEN THE SYSTEM IS UNAVAILABLE

When the training stopped

A real experience can demonstrate dependency without becoming a fabricated incident.

DOCUMENTED EXPERIENCE, DETAILS LIMITED FOR OPERATIONAL SECURITY

During an early hotel training shift, a planned system update left the trainee unable to practise normal system tasks for several hours. A colleague could explain the work, but the usual live demonstration could not proceed. The training did not become a dramatic emergency. It became an extended lesson in how much procedural learning depended on access to the interface.

This is narrower than a “night when the entire hotel system failed”. The evidence supports an interruption associated with maintenance and limited hands-on training; it does not support claims about damaged data, incorrect charges, guest consequences, emergency access or a failed recovery. Those facts are not known and do not belong in a public narrative. The episode connects to the author's later effort to organise approved operating knowledge into searchable, visual guidance, the Front Desk Control project. Such a guide can support learning, but it must not be presented as a substitute for the official system, the local operating procedure or an authorised contingency plan.

OUR READING

When the interface becomes unavailable, the quality of preparation becomes visible.

The author's own training shift, told without the property, the system or the date; the essay named below and the project page tell the rest.

[More in the essay](#)

[The night the system went down](#)

CHAPTER V · 03:00: WHEN THE SYSTEM IS UNAVAILABLE

Out of the loop is not out of duty

Less active work can mean a weaker working picture at the moment intervention is needed. Endsley and Kiris (1995) investigated operators using an expert system for a navigation task. They linked lower situation awareness under automation to worse decision-time performance after the expert system failed; the level of control available to the operator moderated the effect. This was an experiment, not an observation of hotel night auditors or delivery dispatchers.

01

Perceive

What is happening now?

02

Understand

What do the
observations mean?

03

Anticipate

What could happen next
if nothing changes?

The three prompts above are an editorial teaching device built around situation awareness, not a substitute for its complete research literature. In operational practice, a handover that records only an end status can deprive the next worker of the path by which that status was reached. When the system is unavailable, the worker also needs to know the limits of their role, how to obtain authorised guidance and which actions must wait. The test of resilience is not whether a person can reproduce every screen from memory. It is whether the organisation has preserved a safe, authorised ability to orient itself.

OUR READING

A person who is accountable for a system needs a way to understand its state.

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995
Endsley and Kiris, *Human Factors* 37(2), 1995, a laboratory study; Bainbridge (1983) for the loss of practice; the three prompts are editorial.

CHAPTER V · 03:00: WHEN THE SYSTEM IS UNAVAILABLE

Confidence is not control

A transparent explanation can help, but it does not automatically produce better judgment.

Lee and See (2004) describe trust in automation as consequential when complexity or unexpected conditions prevent a complete understanding of a system. The goal is appropriate reliance under the circumstances, not maximum trust. A confident user can still rely on the wrong output; a cautious user can also reject useful assistance. Romeo and Conti's review (online 2025) found that explanations in human-AI decision-making do not reliably eliminate automation bias. Complex explanations may overload people; simplified explanations may make a suggestion seem more credible without supporting independent checking. Their evidence concerns AI-assisted decisions, not every ordinary enterprise application.

01	Explanation	What does the system claim and on what inputs?
02	Verification	What can an authorised user independently confirm?
03	Authority	What change can this role approve, and what needs escalation?
04	Recovery	If the tool is unavailable, where is approved guidance?

OUR READING

Good assistance increases the user's ability to judge, not merely the user's willingness to agree.

Sources: John D. Lee & Katrina A. See, Human Factors 46(1), 2004; Giuseppe Romeo & Daniela Conti, AI & Society (online 2025, volume 41, 2026), 2025

Lee and See, Human Factors 46(1), 2004; Romeo and Conti, AI & Society, online 2025. The four rows are editorial.

CHAPTER V · 03:00: WHEN THE SYSTEM IS UNAVAILABLE

Continuity without improvisation

A system-interruption plan should be knowable, authorised and safe before it is needed.

- 1 Know the approved source**
Find the approved continuity instructions during training, not during the outage.
- 2 Define limits**
Which work proceeds, which pauses, who authorises exceptions.
- 3 Agree the communication**
Designated channels; no contradictory promises to customers or colleagues.
- 4 Keep an accountable record**
Open issues documented by approved, privacy-protecting means.
- 5 Confirm controlled return**
Authorised staff verify readiness and reconcile open issues officially.
- 6 Learn afterwards**
Review gaps in training and documentation without exposing sensitive controls.

This volume deliberately gives no instructions about guest access, payment control, credentials or recovery mechanics: preparedness and responsibility can be examined without publishing a playbook that could weaken security.

OUR READING

Preparedness is an organisational capability, not a secret workaround remembered by one person.

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995; John D. Lee & Katrina A. See, *Human Factors* 46(1), 2004

Editorial continuity protocol after Bainbridge (1983), Endsley and Kiris (1995) and Lee and See (2004); not a contingency plan of any hotel.

1 SCENE

2 BREAKDOWN

3 PROTOCOL

CHAPTER VI · THE RESPONSIBLE OVERRIDE

The decision between two systems

There is an especially difficult moment when everyone can point to a record, but nobody can identify the decision owner.

HYPOTHETICAL SCENE, AN EDITORIAL TEST CASE

An automated recommendation conflicts with a verified observation. The operator recognises the discrepancy but hesitates: the software is treated as authoritative, while the procedure for challenging it is unclear. The manager asks whether the team should “override” the recommendation. Yet override is not a magic button. It may mean requesting review, recording an exception, pausing a non-critical action or escalating to a person with the appropriate authority.

If a company treats the event as a personal contest between experienced staff and a machine, it misses the design question. What happens when both sources disagree? Does the employee have permission to stop? Who makes the next decision? How will the system be corrected if the observation proves reliable? No specific user of Front Desk Control, no hotel incident and no route correction are claimed in this scene. It is a test case for designing human accountability, not evidence that an override has been executed.

OUR READING

The right to question a recommendation is meaningful only if someone can decide what happens next.

Constructed test case; the essay named below argues the same point for the scan record.

[More in the essay](#)

The scanner is a witness, not a judge

CHAPTER VI · THE RESPONSIBLE OVERRIDE

What exactly is automated?

A system can automate information without being authorised to automate the final decision. Parasuraman, Sheridan and Wickens (2000) distinguish four classes of functions that can be automated at different levels. This is a useful antidote to the vague claim that “the system decided”. A route planner may acquire and analyse information, then suggest a plan. Another system may execute a pre-approved action. Those are different responsibility arrangements.

01	1 · Acquisition	Collect data: events, positions, times, entries.
02	2 · Analysis	Transform data: calculate, compare, flag an exception.
03	3 · Selection	Recommend or choose a course of action.
04	4 · Implementation	Carry out the action under the chosen authority level.

The central managerial task is to make those handoffs explicit. A status badge may express a rule-driven conclusion, not an independent observation. A recommended change may require a different role's approval. Even when implementation is automated, an organisation remains responsible for policies, training, controls and review.

OUR READING

Ask which function the software performs before saying it made the decision.

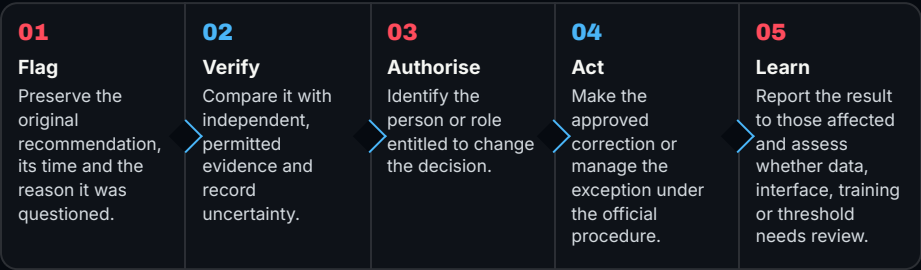
Source: Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, IEEE Transactions on Systems, Man, and Cybernetics A 30(3), 2000

Parasuraman, Sheridan and Wickens, IEEE Transactions on Systems, Man, and Cybernetics A 30(3), 2000; the operational examples are editorial.

CHAPTER VI · THE RESPONSIBLE OVERRIDE

The responsible override model

A five-stage operating cycle for exceptions that the normal display cannot resolve.



The cycle is an editorial synthesis, not an ISO standard, a certification or a control validated on the author's data. And “human” does not mean “unlimited discretion”: some decisions are legally constrained, some demand immediate safety escalation, others a second authority.

01	System output	What does it actually recommend or record?
02	Human evidence	What independent condition has been verified?
03	Decision rights	Who may correct, pause, or escalate?
04	Closure	What proves the outcome, and who hears back?

OUR READING

The purpose of a human override is accountable correction, not a hidden route around safeguards.

Sources: Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, IEEE Transactions on Systems, Man, and Cybernetics A 30(3), 2000; John D. Lee & Katrina A. See, Human Factors 46(1), 2004; Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, Journal of the American Medical Informatics Association 19(1), 2012

Editorial synthesis after Parasuraman et al. (2000), Lee and See (2004) and Goddard et al. (2012); no stage has been validated as a control.

CHAPTER VI · THE RESPONSIBLE OVERRIDE

Measure the decision system

A controlled pilot should measure customer and safety outcomes, not how often humans disagreed with software.

1	Start small	One low-risk, permitted workflow, with its customer outcome defined.
2	Write down definitions	Observed incorrect outputs apart from merely contested recommendations.
3	Sample cases fairly	Accepted and challenged recommendations alike; otherwise the review is biased.
4	Track follow-through	Verification, owner, final disposition and time to communicate the outcome.
5	Audit for unintended effects	New delays, missed cases or unauthorised actions created by the change.
6	Decide to keep or stop	The whole result against the baseline, and what users say improved or worsened.
01	Verification rate	Cases independently checked / cases sampled.
02	Resolution lead time	Time from verified issue to authorised decision.
03	Feedback closure	Decisions communicated to affected staff.
04	Exception quality	Challenges with documented reason and outcome.

Sources: John D. Lee & Katrina A. See, Human Factors 46(1), 2004; Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, Journal of the American Medical Informatics Association 19(1), 2012; Giuseppe Romeo & Daniela Conti, AI & Society (online 2025, volume 41, 2026), 2025

Editorial pilot design and metrics after Lee and See (2004), Goddard et al. (2012) and Romeo and Conti (2025); proposed, not tested for validity.

FIELD TOOL

The Human Override Card

One disputed output. One verified condition. One authorised action. Never place personal, payment or proprietary data on a public copy.

01 System output what exactly did it report, recommend or mark complete?

02 Current evidence what did a person or independent source actually verify?

03 Uncertainty what remains unknown? is the information stale or incomplete?

04 Safety and authority may work proceed? who is authorised to decide?

05 Action and owner what approved action is next, by whom, and by when?

06 Result / tell back what changed? who receives the verified update?

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, *IEEE Transactions on Systems, Man, and Cybernetics A* 30(3), 2000; John D. Lee & Katrina A. See, *Human Factors* 46(1), 2004; Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, *Journal of the American Medical Informatics Association* 19(1), 2012

Original editorial tool, synthesising Bainbridge, Parasuraman et al., Lee and See and Goddard et al. If the cause is uncertain, write the next verification step rather than a conclusion; the tool named below keeps the same fields for a shift.

Open the tool: [Shift Handover](#)

stivencatalyst.com/tools/shift-handover/

CLOSING ESSAY

The system cannot inherit the responsibility

The future of operational management is not a contest between faster software and wiser people.

A dashboard can reveal a pattern that no one standing beside a loading ramp could see. A route planner can calculate possibilities at a speed no dispatcher can match. A hotel system can preserve a consistent record across shifts. These are real advantages. The lesson of this volume is not to distrust them. It is to understand their limits precisely enough to use them well. An organisation begins to lose that precision when it treats a data field as a complete event, a red indicator as a diagnosis or a recommendation as a final instruction. It loses even more when the colleague who questions a result is treated as an obstacle instead of a possible source of evidence. The opposite error is equally real: local experience can be wrong, and “I know the area” cannot replace a verified rule. Good management has to hold both claims up to the same discipline of checking.

Research offers a warning rather than a magic formula. Bainbridge showed that automation can reshape rather than eliminate human work. Endsley and Kiris demonstrated a form of lost situation awareness under particular experimental conditions. Reviews of automation bias reveal why trust, workload, training and explanation design matter. None of these papers measures the author's workplaces. They give us reasons to ask better questions there. The practical answer is not a dramatic “override”. It is a modest chain of responsibilities: preserve the signal, verify the condition, establish authority, make a proportionate decision and return the learning. The aim is to let technology perform what it does reliably, while ensuring people can recognise and address what remains outside its view.

OUR READING

The system can recommend the decision. It cannot inherit the responsibility.

Editorial conclusion. No improvement is attributed to the author's workplaces without an independently documented measurement.

METHOD

Evidence before conviction

How the claims, comparisons, fictional scenes and proposed controls were handled.

01	Research basis	Seven peer-reviewed works or scientific reviews: 1983, 1995, 1997, 2000, 2004, 2012, online 2025.
02	Verified background	Operational categories from last-mile, remote dispatch, hotel night auditing and the Front Desk Control knowledge project.
03	Constructed scenes	All invented events are explicitly labelled hypothetical, composite or illustrative.
04	Proposed practice	No field protocol, alert threshold, pilot result or approval is claimed as tested in an employer's operation.

Study types are not conflated. Bainbridge (1983) is a conceptual industrial automation paper. Endsley and Kiris (1995) report an experimental navigation task. Parasuraman and Riley (1997) and Lee and See (2004) offer interpretive frameworks; Parasuraman and colleagues (2000) classify automation functions. Goddard and colleagues (2012) include 74 studies after screening 13,821 records. Romeo and Conti reviewed 35 studies from January 2015 through April 2025; their article appeared online in 2025. Limits: the studies do not establish the rate of bad routing, false scans, weak training or AI error in the author's roles. The 100-case alert exercise is entirely synthetic. “Human override” is an editorial name for authorised review and correction, not a software feature or permission to bypass controls. Editorial rule: when evidence stops, the sentence must stop claiming.

Sources: Lisanne Bainbridge, *Automatica* 19(6), 1983; Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995; Raja Parasuraman & Victor Riley, *Human Factors* 39(2), 1997; Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, *IEEE Transactions on Systems, Man, and Cybernetics A* 30(3), 2000; John D. Lee & Katrina A. See, *Human Factors* 46(1), 2004; Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, *Journal of the American Medical Informatics Association* 19(1), 2012; Giuseppe Romeo & Daniela Conti, *AI & Society* (online 2025, volume 41, 2026), 2025

Every source is cited at its DOI on the next page; the two counts of the reviews were read in their abstracts. There are no names of guests, colleagues, employers or private systems.

SOURCES AND METHOD

Where the claims come from

The figures in this volume come from the sources below, each with its year. The scenes are constructed; the protocols are proposals; what is research is named as research.

Ironies of Automation

Lisanne Bainbridge, *Automatica* 19(6), 1983
[https://doi.org/10.1016/0005-1098\(83\)90046-8](https://doi.org/10.1016/0005-1098(83)90046-8)

The Out-of-the-Loop Performance Problem and Level of Control in Automation

Mica R. Endsley & Esin O. Kiris, *Human Factors* 37(2), 1995
<https://doi.org/10.1518/001872095779064555>

Humans and Automation: Use, Misuse, Disuse, Abuse

Raja Parasuraman & Victor Riley, *Human Factors* 39(2), 1997
<https://doi.org/10.1518/001872097778543886>

A Model for Types and Levels of Human Interaction with Automation

Raja Parasuraman, Thomas B. Sheridan & Christopher D. Wickens, *IEEE Transactions on Systems, Man, and Cybernetics A* 30(3), 2000
<https://doi.org/10.1109/3468.844354>

Trust in Automation: Designing for Appropriate Reliance

John D. Lee & Katrina A. See, *Human Factors* 46(1), 2004
https://doi.org/10.1518/hfes.46.1.50_30392

Automation bias: a systematic review of frequency, effect mediators, and mitigators

Kate Goddard, Abdul Roudsari & Jeremy C. Wyatt, *Journal of the American Medical Informatics Association* 19(1), 2012
<https://doi.org/10.1136/amiainl-2011-000089>

Exploring automation bias in human-AI collaboration: a review and implications for explainable AI

Giuseppe Romeo & Daniela Conti, *AI & Society* (online 2025, volume 41, 2026), 2025
<https://doi.org/10.1007/s00146-025-02422-7>

EDITORIAL METHOD

Each figure was checked for its year, its publisher and what exactly it measures, at the full text where it could be opened. Where the publisher's page could not be opened, the figure was checked against independent summaries and is marked "via". The editors' interpretation is marked "Our reading". No figure of the author's own workplaces is claimed.



MANAGEMENT REVIEW

Management without theatre.

Every volume, one question about operations, followed from the floor to the protocol
and tested against research.

ALL VOLUMES

stivencatalyst.com/magazine/operational-treatise.html