Engineering Resilience for IT Systems: Chapter 0 – Introduction

Co-editors

Andy Newham, Charles Weir, Ed Steinmueller, Gill Ringland, Jacqueline Hogg, James Davenport, James Glover, Michael Crooymans, Paul Reason, Peter McElwaine-Johnn, Ran Tao, Rony Zaman, Sarah Greasley, Sathishkumar Palanisamy, Steve Sands, Sue Milton, Tarnveer Singh

Lead contributors: Paul Reason, Gill Ringland, Ed Steinmueller, Rony Zaman

August 2026

Acknowledgements

This Handbook is an outcome of extensive work hosted by the BCS IT Leaders Forum, and starting in 2022. Major reports published contain more than 1000 names of people who have contributed, through Round Tables, in interviews, or through contributing to working papers or reports. So rather than trying to acknowledge each by name, may we refer to the significant reports and books, each of which lists the contributors.

https://www.bcs.org/media/9679/itlf-software-risk-resilience.pdf

https://nationalpreparednesscommission.uk/publications/elephant-in-the-room/

https://www.bcs.org/media/3j1n1mhc/service-resilience-and-software-risk-2023.pdf

https://www.bcs.org/media/ku4fdukn/treat-it-resilience-like-we-treat-safety.pdf

https://www.bcs.org/media/tvudbfex/transparency-software-is-the-elephant-in-the-room-policy-brief-v5.pdf

https://www.bcs.org/media/rxdmjr5h/availability-6-report-nine-banks-data-roundtable-220425.pdf

https://www.bcs.org/media/eitldxyk/availability-service-resilience-summary-of-the-working-group-papers.pdf

Engineering Resilience for IT Systems: Chapter 0 – Introduction

Why this handbook is needed

Our economy and society depend on IT systems. These systems have been put together over 50 years. They rely on a complex global network of software and service suppliers. Their complexity and volatility mean that they are subject to failure. Cyber attacks may take down services and steal data. When this happens, it disrupts organisations and citizens. The impact can range from inconvenience to loss of life.

Organisations are increasingly being held accountable to customers for service outages and data loss, no matter the reason. So, boards and their teams must find new ways to operate to make systems more available, more resilient against disruptions. Tools include strategy and planning, operations, and the use of AI.

Organisations contain a number of silos with differing responsibility for the resilience of their IT systems.

CEO's and Board members need to be better informed in order to understand organisational risk and potential reputational damage.

CTO's, CSO's and CISO's bridge between the IT systems and the business. People in these roles may well already be aware of the risks and seek to implement processes and tools to minimise them

Supply Chain and Vendor Managers are increasingly central to the effective provision of IT services, and their role is evolving.

Programme Managers and Enterprise Architects define ongoing strategic, that is architectural, changes. These can help reduce the impact and frequency of outages.

Site and System Reliability Engineers and Availability Team roles are of increasing visibility, covering as they do the operation of the IT system.

Disaster Managers, Business Continuity and Risk Managers need to balance human-focused processes with automation.

We also recognise the need to brief entrants to the IT industry on the roles beyond computer science education. We suggest that they use this Handbook to browse the focus and accountabilities of key roles.

This Handbook provides advice and checklists for each silo, as well as emphasising the governance structures and language to manage the many cross-dependencies.

Engineering Resilience for IT Systems: Chapter 0 – Introduction

Chapters

Annexes

0. Introduction

0.1 Purpose of this Handbook

Organisations are increasingly being held accountable to customers for service outages, no matter the reason. So, boards and their teams must find new ways to operate to make systems more available, more resilient against disruptions. And when resilience – the ability to adapt – does not work, there is a need to recover from catastrophic failures.

Organisations contain a number of silos with differing responsibility for the resilience of their IT systems. While the responsibility for resilience spans the whole organisation, for our discussion of the resilience of IT systems, we have chosen to identify the key roles of:

- CEO's, NED's and the Board (the Board)
- CTO's, CIO's and CISO's (the CTO family)
- Supply Chain and Vendor Managers
- Programme Managers and Architects
- Availability and Resilience Managers eg Site Reliability Engineers
- Business Continuity and Disaster Recovery Managers.

This Handbook provides advice and checklists relevant to the role of each function, as well as emphasising the governance structures and language to manage the many cross-dependencies. The language of each chapter is oriented towards the needs of the main professional community with a role – but we hope that all chapters are structured so that the essence is available to all professionals. In particular, we recognise that Computer Science students and young professionals need to be aware of how current IT Systems function – we encourage them to browse the briefing material and skip the checklists and Appendices.

This Introduction explains why the Handbook is needed now, who it's for, and how to use it.

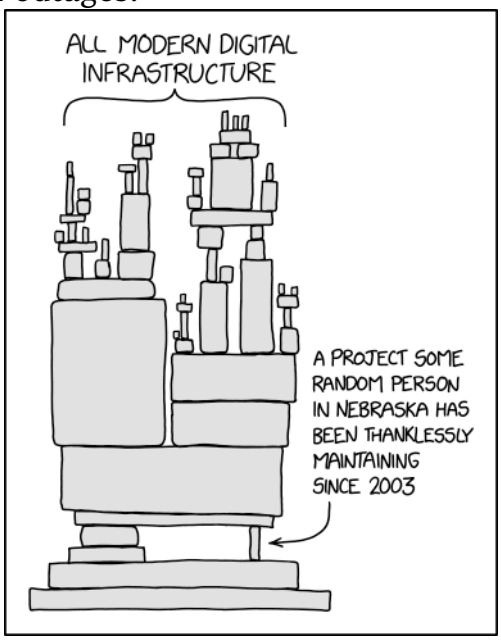
Engineering Resilience for IT Systems: Chapter 0 – Introduction

0.2 Why this Handbook is needed now

0.2.1 Economic and societal dependence on IT systems

Software malfunctions that cause failures and service outages can happen for many reasons.

- Cyber-attacks make headlines. They often result in damage to data and long standing disruption of services.
- Software flaws, user or operator errors, unexpected high demand, data which breaks the system, new vulnerabilities from technologies like AI, can all cause service outages.
- IT systems are subject to frequent change in the supply chain: upgrades/changes are frequent causes of outages.
- Most services are delivered by a complex set of software components from a variety of sources.



Access to services may be blocked. Data may be lost, corrupted, or looted. A service outage can be brief and impact just a few people, so it may be ignored or seen as random, like cosmic rays. However, an outage can also last a long time, affecting millions and causing significant damage.

IT system failures impact more people and various parts of life as we rely more on digital systems: they threaten prosperity, productivity, security, health, and welfare. Many people do not fully understand these risks linked to digitisation. This lack of awareness makes it harder for organisations to adapt and thrive in a changing environment.

0.2.2 Redefining responsibility for digital resilience

Organisations provide services to customers.
The availability of these services is a key measure of their viability and reputation. Boards worry about front-page news related to disruptions caused by their IT systems or those of their suppliers. However, this concern has not yet translated into effective action by most Boards.

There is growing attention in the UK Government to the impact of service outages on the UK economy. The key questions are: where is action needed (priorities), who will take action, and what levers can encourage it?

A clear priority is essential services. These include water, electricity, sewage, health, transport and communication services. All of these increasingly rely on digital services. It is not only the operators of essential services, but also those who provide digital services to these operators, that need to be resilient.

Parliament is reluctant to increase regulations and their costs. This is because it wants to boost growth and productivity. While-self regulation might be preferable to all, the prospects of *effective* self-regulation are dim. So, the UK Government has tabled regulatory solutions. These focus on increasing the liability of organisations for service outages that affect users. Mechanisms include fines, reputational damage, and litigation costs.

With every new outage in the headlines, the pressure for regulation is increasing. The boards of these organisations have the final say, but they struggle to determine the right actions. Therefore, the supporting management will be called on to propose and justify specific actions. Substantive discussion of how to improve resilience is increasing and this Handbook is part of that broader movement.

Resilience is an organisational endeavour: we identify the various responsibilities and accountabilities in the Annex: RACI for resilience deliverables.

0.2.3 Need for new governance and skills

Availability of services is increasingly a Board responsibility, with corresponding liability. Boards need to take control of governance of the risk from IT failures. They need to ask for assurance that essential services have the necessary availability.

Availability of services depends on resilient systems. Resilience Engineering describes the methods, tools, and approaches organisations can use to ensure “enough” availability.

This Handbook describes the new business environment and the characteristics of IT systems as they affect resilience. The way in which organisations buy and operate IT systems has radically changed over the last decade. This shift has been from mostly in-house services to a complex network of third-party providers. While large software and systems providers provide training on their products

and services, there is little training covering a wider view of this new environment – its challenges as well as its opportunities.

This Handbook shows how organisations can lessen the impact of outages. It covers planning to reduce scope and duration of outages. It covers the role of anticipating and avoiding disruptions. It also highlights the importance of quick and effective responses and recovery after an outage.

Organisations need new governance and skills in this new regime.

0.3 How this Handbook addresses the need for new governance and skills

This Handbook provides examples, advice and checklists which can help organisations improve the availability of IT systems. The chapters are organised by job role: however, it is important to understand that each job role needs information and co-working with other roles to achieve improved resilience.

0.3.1 Chapter 1: Why Resilience Matters describes the economic and societal role of IT systems and the impact of service outages. It underlines why it is important for organisations to be more resilient and why CEOs and Boards need to address this responsibility.

0.3.2 Chapter 2: Making resilience a reality provides advice and checklists for governance in the new regime. It is for use in discussion between Boards and CTO's, CIOs, and CISO's to improve clarity of responsibilities and support implementation.

0.3.3 Chapter 3: Resilience for Vendor and Supply Chain managers provides advice and checklists on standards and sources of support for these roles. They are central to improving resilience.

0.3.3 Chapter 4: Planning for Improved Availability provides advice and checklists on planning for improving service delivery. These resources for Enterprise Architects and Programme Managers help head off potential vulnerabilities.

0.3.4 Chapter 5: Service Delivery is intended to support Site and System Reliability Engineers and Availability Teams. It provides advice and checklists to reduce the frequency, impact and blast radius of service outages.

0.3.5 Chapter 6: Response and Recovery covers what to do after a failure which interrupts the organisation's service delivery. For both recoverable and catastrophic failures, we provide checklists and advice on the changing balance between automation and people-based processes.

0.3.7 Appendices cover

- AI Support for Enterprise Architecture
- Glossary
- Public Reports
- RACI for Resilience Deliverables
- Standards
- Review Guide

0.4 How to use this Handbook

0.4.1 Focus on Chapters by role in the organization

For each of the chapters, there is a primary audience:

1) CEO's and Board members need to be better informed in order to understand organisational risk and potential reputational damage. This is the focus of Chapter 1

2) CTO's, CSO's and CISO's bridge between the IT systems and the business. People in these roles may well already be aware of the risks and seek to implement processes and tools to minimise them. This is the focus of Chapter 2.

3) Vendor and Supply Chain managers – Chapter 3 provides advice and checklists on standards and sources of support for these roles as the need for informed procurement becomes clear.

4) Programme Managers and Enterprise Architects define ongoing strategic, that is architectural, changes. These can help reduce the impact and frequency of outages. This is the focus of Chapter 4.

5) Site and System Reliability Engineers roles are of increasing visibility, covering as they do the operation of the IT system. The roles are evolving rapidly: this is the focus of Chapter 5.

6) Disaster Managers, Business Continuity and Risk Managers need to balance human-focused processes with automation. This is the focus of Chapter 6.

0.4.2 Navigation

This Handbook is organised into Chapters 1 to 6 as described above. Under each are sections, e.g. this is in 0.4. Sections are further divided into *topics, e.g. 0.4.2.* Within each section there may be briefing, examples, advice and checklists.

Briefing text is in normal script.

Advice is in italic.

Checklists are in boxes

Examples are in boxes

Advice:

If you have a specific topic in mind,

The annotated Table of Chapter Contents above at 0.3 lists main topics in each Chapter.

If you are trying to formulate the question

First try the briefing, advice and checklists in the Chapter appropriate to your role. If that does not help, the Glossary may be a place to start.

0.4.3 Some housekeeping

Acronyms - the first time it's mentioned, it is spelled out in full, with the acronym then used from there on. But if you spot an acronym and you don't know what it is, it will also be in the Glossary.

The information we have provided is current at the time of publishing, August 2026. But we are aware that the status of legislation and standards changes over time, and that links disappear.

Where company or product names are given, it's for information, it is not an endorsement. And it may not be current after a few months of our publication!

Engineering Resilience for IT systems:
Chapter 1 – Why Resilience Matters

1. Why resilience matters

1.1 Purpose of this Chapter

This chapter aims to increase Board understanding of why resilience matters.

1.1.1 Key messages

IT systems underpin most services that organisations supply to their users. Users suffer – from inconvenience to life threatening events – when IT systems fail to be available. But IT systems do fail — not occasionally, but inevitably — because they are complex, tightly coupled, and under continuous change.

So, organisations need to ensure that their systems have the resilience to adapt and recover, to deliver services even though IT systems may fail. Availability now is becoming a focus of board and regulatory attention. AI has the potential to improve availability, although there few proven case studies as yet.

- Failure of IT systems is a design assumption, not an exception.
- The user impact of failure is now visible and increasingly regulated.
- Investment in resilience needs to be matched to the importance of each service — not applied uniformly.

Engineering Resilience for IT systems:
Chapter 1 – Why Resilience Matters

Questions CEOs and Boards should ask

Do we accept that IT failure is inevitable in complex, tightly coupled systems?

Do we recognise the growing impact of IT failures, from both cyber-attacks and from ineffective change management?

Have we identified our Important Business Services (IBS) and set Impact Tolerances and availability targets that define the maximum disruption we can absorb before intolerable harm reaches users?

Do we recognise that the UK Cyber Security and Resilience Bill is leading towards Board-level liability for service availability, not only for CNI and Operators of Essential Services?

Can we demonstrate to a regulator that resilience investment is proportionate to each service's importance, accepting that not all services need high availability and that over-engineering carries its own cost?

Have we understood that returning incorrect data can be a more severe failure than returning no data, and that reputational and customer-trust damage from a visible outage should be quantifiable, as in the NIS Framework?

1.1.2 Topics in this chapter

- The need for better availability of IT systems
- Characteristics of contemporary IT systems
- Availability – increased regulatory and legislative focus on user impact
- Standards, Guidance
- What the Board should expect to see

1.2 The need for better availability of IT systems

Our economy and society depend on digital systems. These are complex and have evolved over time, and are liable to unpredictable failures.

It is not easy to estimate the real cost of system outages. Published reports for both cyber incident costs and internal causes often use the four headings below.

1.2.1 "The True Cost of IT Incidents[1]" - publicity from the impact of cyber attacks

Boards and Governments are rightly concerned about cyber-attacks. The impacts of cyber incidents are widely reported:

- Financial Losses: The average cost of a data breach reached $4.88 million in 2024, reflecting a 20% increase since 2020. These costs stem from regulatory fines, legal fees, and remediation efforts in the attacked organisation.
- Market Value Decline: Publicly traded companies suffer an average 7.5% drop in share price following a cyber incident, as investors react to financial uncertainty and reputational damage.
- Customer Trust & Retention: 55% of U.S. consumers report they would be less likely to continue business with a company that has experienced a breach, underscoring the long-term brand damage.
- Operational Downtime: The 2024 cyberattack on Change Healthcare, which cost the broader UnitedHealth Group a total of $1.1 billion, disrupted payments across the U.S. healthcare system for weeks."

In Europe, outages due to cyber-attacks have included the British Library, Jaguar Land Rover, Marks and Spencer, and many affecting the UK's NHS.

1.2.2 Why IT failures will continue to happen

IT systems are complex and often tightly coupled systems. They rely on massive amounts of code, hardware with multiple possible points of failure; human operators may take unexpected actions. The net result is that IT systems must be expected to fail.

Modern IT systems are continuously expanded and revised to provide new services and to take advantage of new technological opportunities. The revision process itself is a major source of failures.

[1] Extracted from the Computer Weekly Report, https://media.bitpipe.com/io_33x/io_333140/item_2927565/Technology%20Risk%20Management%20-%20A%20Proactive%20Framework%20for%20the%20Digital%20Age-Whitepaper.pdf

The inevitability of failure necessitates a focus on resilience to achieve availability. Resilience is the organisation's ability to withstand and recover from disruptions. Resilience requires that organisations "anticipate, absorb, recover, and adapt"[2] to IT failures.

Efforts to create 'bullet proof' systems can make failures less frequent, or with a smaller "blast radius". But the systems are inevitably still subject to failure. And ransomware attacks are increasing and often focussed on infrastructure[3].

The methods and tools in this Handbook are aimed at averting a catastrophic, major, company-threatening, data and service loss. Many major outages are preceded by anomalous traffic or system behaviour. The emphasis in all resilience thinking is about anticipating and containing potential failures early on, before a major collapse.

1.2.3 Outages from causes other than cyber attacks

Data from the Nine Banks report[4] suggest that over two years, about 10% of the outages recorded by the top UK Banks were due to cyber-attacks. Many of the rest of the outages were while upgrading the system or due to user (mostly mobile) or operator error. So it is important to identify the cost estimates for outages from all sources, not just cyber attacks.

Considering IT outages more widely, the impacts of IT incidents are profound. They disrupt operations, erode trust, and cause financial damage that extends far beyond the initial incident [5]:

- Financial Losses & SLA Penalties: The financial impact of IT downtime is staggering. Over 90% of midsize and large enterprises report that a single hour of downtime costs more than $300,000[6]. These costs cover immediate revenue loss, Service Level Agreement breach penalties, and resource-intensive recovery efforts.

---

[2] The CEOs of the UK FTSE350 companies have been urged by the UK Government to look toward "resilience engineering", systems that can anticipate, absorb, recover, and adapt, in the event of IT threats.

[3] https://www.csoonline.com/article/4212157/ransomware-takes-aim-at-enterprise-resilience.html

[4] https://www.bcs.org/media/rxdmjr5h/availability-6-report-nine-banks-data-roundtable-220425.pdf

[5] As note 1 above

[6] https://thenetworkinstallers.com/blog/cost-of-it-downtime-statistics

- Market Value Decline: Publicly traded companies can suffer significant drops in share price following high-profile service disruptions[7]. Investors react swiftly to financial uncertainty and a perceived lack of operational reliability when core infrastructure fails.
- Customer Trust & Retention: System availability is synonymous with brand reliability. Research indicates that 65% of consumers report trusting a business less after experiencing a site error or service outage. This underscores the long-term reputational damage caused by unstable systems[8].
- Operational Downtime: In modern distributed cloud architectures, localized disruptions can rapidly cascade into global failures. For instance, the Crowdstrike July 2024 global IT outage was triggered by a faulty routine software update. It crashed millions of devices worldwide and disrupted critical infrastructure from airlines to healthcare, for days.

"These impacts serve as a reminder that technology risk is not an IT problem. It is an enterprise risk. It requires executive oversight, structured risk management and cross-functional resilience strategies. Managing these risks demands an integrated approach that accounts for the way that technology risks interact and amplify one another[9]."

The Annex: Public Reports describes some major outages, including in Europe:

- National Air Traffic System (NATS) in 2023[10]
- the power outage that closed London Heathrow[11] in 2025,
- the blackout that left most of Spain and Portugal without electricity for days[12] in 2025,
- major cloud disruptions at Amazon Web Services (AWS)[13]
- Cloudflare[14] in 2025.

1.2.4 Board's responsibility for the organisation's reputation

The responsibilities of board members include, regardless of business type[15]:

- establishing vision, mission and values
- setting strategy and structure

[7] https://medium.com/@euanw4872/global-it-outage-how-crowdstrike-navigated-reputational-challenges-and-stakeholder-confidence-in-b3a5183fd115
[8] https://queue-it.com/blog/cost-of-downtime/
[9] As note 1 above
[10] https://www.caa.co.uk/publication/download/23337
[11] https://www.thebci.org/news/the-kelly-review-lessons-from-heathrow-s-power-outage.html
[12] https://www.bbc.co.uk/news/articles/cvg0r3z3lvqo
[13] https://www.bbc.co.uk/news/articles/cev1en9077ro
[14] https://www.theguardian.com/technology/2025/dec/05/another-cloudflare-outage-takes-down-websites-linkedin-zoom
[15] https://www.iod.com/resources/company-structure/what-is-the-role-of-the-board

- delegating to management
- exercising accountability to shareholders and being responsible to relevant stakeholders. This includes eg legal compliance with regulations, risk considerations.

Risk is flagged by the IoD:

> *Determine the company's appetite for risk. Engage in the process of backing a robust risk management programme focused on the company's business and the area(s) of its activities.*

However, Boards in the private sector are often not yet aware of the liability and reputational risks of IT failures.

Responsibility to stakeholders includes legal compliance. This is necessary but not sufficient. Further, there is no generic legislation mandating the availability of services. Some sectors, e.g. telecoms, do have published Service Level Agreements (SLA's)[16] for services they offer to customers, and published penalties for not meeting these levels.

Board members wish to avoid nightmare risks. What are the nightmares keeping the Board awake at night? Frequent nightmares are about e.g. "Bad headline in the Daily Mail" or "viral postings on Social Media". These damage the organization's public image and reputation.

A number of potential causes of nightmares are IT related:

- Data corruption/loss
- Supply chain fiasco (including but not just software)
- Failure of a major strategy eg acquisition
- Disruption eg loss of site/people or Important Business Service.

1.2.5 Governance – Board responsibility

Strategic Governance is the responsibility of the Board. This includes:

- Executive and Board Leadership: Understanding technology risk, in the context of strategic awareness and appreciation of financial implications.
- Accounting Officer/CEO Accountability – see Chapter 2:
- Designated Board Member for Resilience: Organizations, especially those managing Critical National Infrastructure must appoint an accountable board member specifically for cyber and software resilience[17].

[16] https://www.bcs.org/media/034nuwdr/availability-the-role-of-slas.pdf
[17] https://www.gov.uk/government/publications/government-cyber-action-plan/government-cyber-action-plan

Additionally,

- Boards should engage in training (such as the BCS/Non-Executive Director Association (NEDA)[18] certification programme). This builds governance competence and avoids "cognitive blind spots."
- Boards should foster a "Just Culture" of openness and fairness when assessing incidents and near-misses. This ensures staff can report vulnerabilities or mistakes (like clicking a phishing link) without fear of retribution, see section 2.3.

1.2.6 Advice: requirements for resilience in organisations

Improved availability of services depends on increased resilience. The protection of the organisation's reputation is an increasingly important factor in setting risk appetite, strategies and approaches to resilience. But, making systems more reliable is costly and often there is no data to assess the cost benefit trade-off.

Boards need a better understanding of their important business services and the impact on users should they suffer outages, in order to define priorities. Cost and the risk to stability from introducing new systems are both a factor in the continuing use of legacy systems.

[18] https://www.bcs.org/qualifications-and-certifications/neda-training-and-certification-programme/

1.3 Characteristics of contemporary IT systems

This section highlights the ways in which IT systems are very different from even a decade ago.

1.3.1 Complexity

In recent brainstorming session a participant mused:
"In the good old days, when it was much more simple, you could say that, I've got my databases, I've got resilience, I have my business logic tier, my presentation layer. But now, we've got microservices, we've got things running in different environments, different sites, you've got some services running in the cloud, some services running locally, ... it's a whole mishmash of things."

Systems now include more lines of code, of variable quality, from many sources – Commercial Off The Shelf Software (COTS), Software as a Service (SaaS), open source, legacy software. Software is supplied through global supply chains: and updates that create new versions are frequent. The pressure for new features limits the ability to build in architectural resilience features. Service delivery systems are tightly coupled complex software systems. Natural Accident Theory[19] (NAT) predicts that in complex and tightly coupled systems, like today's IT systems, an accident type known as the "normal accident", is inevitable.

There is increasing use of external SaaS providers who provide access to complex functionality outside the organisation's control. These systems are often integrated into critical business processes, but where they are hosted and how they are constructed and maintained is often not visible.

A common feature of these services is that they are often 'evergreen' so that the supplier undertakes to patch and maintain them. However, they may also be enhanced to use services you are not aware of, in order to provide better availability.

Managing the complexity of a hybrid multi-cloud service across multiple tools, systems and processes was the primary challenge faced by IT leaders surveyed in a recent study[20]. It was cited by 60% of respondents. The other two leading concerns were excessive complexity and cost, and lack of insight into IT health and warning of any problems.

[19] https://www.sciencedirect.com/science/article/pii/S0265964624000444
[20] See https://www.kyndryl.com/content/dam/kyndrylprogram/cs_ar_as/AIOps_AS_USEN.pdf

1.3.2 Longevity

Many organisations have applications containing legacy software dating back to the 1970's. A legacy system is an older (but not obsolete) part of the IT landscape that's still in use today. Legacy systems often support mission-critical operations and business processes, since they were built first. Lack of the skills used to create them, and often undocumented links to other systems, makes updating or replacing them risky and complex.

Example: Legacy system: Equifax Data Breach[21]

The Incident
In 2017, Equifax experienced one of the most devastating data breaches in history, exposing sensitive information of 147 million Americans. The breach occurred because Equifax had failed to patch a known vulnerability in an Apache Struts web application, which was part of their legacy system.

Impact of Compromise
The breach led to nearly $700 million in fines and settlements, damaging Equifax's reputation and trustworthiness as a credit reporting agency.

What Could Have Prevented It
Had Equifax kept its legacy systems up-to-date with the latest patches, the breach might have been avoided. Regular vulnerability scanning and patch management are essential components of IT security.

1.3.3 Changed user behaviour

The basic structure of the Internet includes protocols such as TCP/IP and DNS which were designed in a context where users were identifiable, and largely trusted, members of the research community. Users operated under social norms of responsible behaviour and they were identifiable and accountable for their behaviour.

The public Internet has introduced a much more diverse group of users. They include fewer expert users, and also larger numbers of hackers who engage in predatory behaviour. Hackers and malign attacks have created myriad problems for the security of data and communication. Hackers seek fame from 'breaking the system'. Malign attacks may promulgate harmful software for private gain (e.g. ransomware) or as part of cyber warfare.

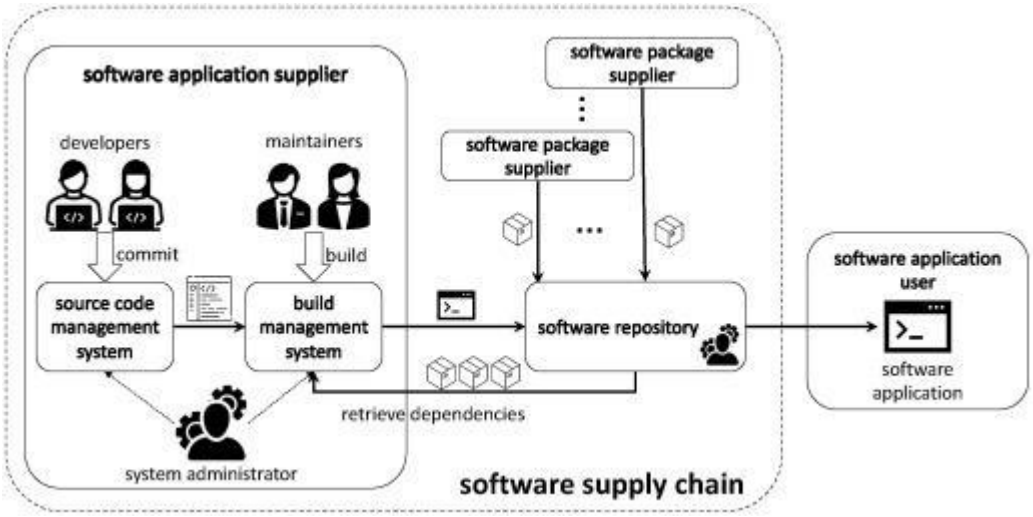
[21] Top 5 Instances When Legacy Systems Were Compromised: DBot Journal

Risks from vulnerable software include leaking credentials or confidential data, corruption of data, installation of malware, and service outages.

1.3.4 Supply chains and volatility[22]

"To identify potential threats to your software, you need to understand the software development process, specifically the components of the software supply chain. It can be broken down into two categories: the people who create the software (including vendors) and the automated systems used to develop the software."[23]

An end-to-end diagram shows the modern software supply chain all connected in a single flow: source code, open-source dependencies, developers, CI/CD pipelines, build systems, artifact registry, cloud/runtime, and third-party services. The software supply chain also includes all the different parts of the system that contribute to creating the software application. These include proprietary code written by the vendor, the vendor's use of open-source libraries, tools for continuous integration and continuous delivery (CI/CD), build systems, cloud services, APIs, and third-party vendors.



Source:
https://www.sciencedirect.com/science/article/pii/S2214212625003606

Unlike traditional supply chains, the software supply chain is dynamic and changes every day. In addition to new dependencies being added daily, pipelines are constantly updated through automation, and software application updates

[22] Excerpted from https://cycode.com/blog/software-supply-chain/
[23] https://cycode.com/blog/software-supply-chain/

are released multiple times a day. Understanding how each of these elements interacts with the others in today's applications is crucial.

1.3.5 24/7 operation

"Many systems delivering digital services are required to operate 24/7. This requires for instance:

- Ability to do maintenance and apply upgrades with little or no interruption in service levels; traditional methods take systems down for upgrades or changes.
- A trained and responsive staff with specialist skills and understanding of business objectives; staff working shifts are often less skilled than day staff.
- A supply chain for complementary and linked services that can operate 24/7; lengthy supply chains, even when supported by contractual SLAs, may fail to provide needed support.
- Automation of alerts of service outages or unexpected behaviour that connects when necessary to humans able to take action; experienced operators may not want to work out of hours shifts.
- Systematic scheduling of testing – including stress testing for impact tolerances and monitoring of installation of new releases; this requires new skills in software testing staff. [24]

1.3.6 Operators/Users

IT operations[25]
A recent study by Futurum Research, in partnership with Nokia, shines a light on the "human factor" in reliability. There's a common saying in IT: "The biggest risk is between the chair and keyboard."

When asked how frequently do operator or configuration mistakes impact service, nearly 99% acknowledged that at least on occasions, an outage or issue has resulted from a manual slip-up. The majority view such incidents as infrequent but 17.5% admitted that human errors occur frequently and are among their top causes of disruption.

The survey asked what percentage of all reliability issues stem from operational/human mistakes; most respondents estimate that human error

[24] Excerpted from Chapter 4 of https://londonpublishingpartnership.co.uk/books/resilience-of-services-reducing-the-impact-of-it-failures/
[25] Excerpted from https://www.nokia.com/blog/people-process-and-pitfalls-tackling-human-error-and-skill-gaps-in-the-data-center-network-operations/

accounts for a minority of incidents. Only 2% felt that over half of the issues are human-caused. This suggests that while people are a known source of risk, many other factors (hardware failures, software bugs, etc.) also cause outages.

View from users – complexity reduces productivity

Research by Workspace 365[26] suggests that overcomplicated IT systems are causing frustration and reduced productivity among UK employees. The study surveyed 1,000 UK workers employed in organisations with more than 250 people.

Six out of ten employees stated that a simplified digital workplace should be a priority for their company's tech investments. Nearly four in ten admit to not using the programmes provided by their employers, and nearly a quarter expressed a need for more training to maximise the use of their digital tools.

A third of hybrid workers identified the burden of using multiple apps and programmes as their foremost challenge, with inefficient communication due to switching between various platforms. Excessive notifications were highlighted as a disruptive factor for a quarter of them.

The study also noted the rise of generative AI among younger employees. Almost 38% of those aged 24-35 reported using personal tech tools like ChatGPT for work. Respondents raised concerns about security of data held by the firm if these tech tools had access to corporate systems.

[26] https://itbrief.co.uk/story/complex-it-systems-decrease-productivity-in-uk-workplaces

1.4. Availability – increased regulatory and legislative focus on user impact

1.4.1 Legislation and NIS

The UK Government is planning a Cyber Security and Resilience Bill. It applies to Critical National Infrastructure (CNI)[27] and Other Essential Services (OES) organisations. It is planned to be enforced using the Network and Information Systems (NIS) framework.

Checklist: NIS Framework metrics	
Parameter	Metric
Availability	Your service was unavailable for a number of affected users for a duration of 60 minutes (lost user hours).
Integrity, authenticity or confidentiality	The incident resulted in a loss of integrity, authenticity or confidentiality of the data your service stores or transmits, or the related services you offer or make available via your systems.
Risk	The incident created a risk to public safety, to public security or of loss of life.
Material damage	The incident caused material damage to at least one user.

The UK Information Commissioners Office (ICO) is responsible for regulating legislation including relevant digital service providers under the NIS regulation. While there are questions about enforcement, it is important in the longer term for an indication of a direction of travel, towards Board liability for non-delivery of services to users.

1.4.2 Regulation in Financial Services and IBS

Two main two frameworks applying to financial services in the UK are DORA[28] from the EU and the UK's Operational resilience[29].[30]

DORA: targets the digital operational resilience of entities within the EU's financial sector. It focuses on Information and Communication Technology (ICT)

[27] CNI are national assets that are essential for the functioning of society, such as those associated with energy supply, water supply, transportation, health and telecommunications.
[28] https://www.eiopa.europa.eu/digital-operational-resilience-act-dora_en
[29] See https://www.bankofengland.co.uk/financial-stability/operational-resilience-of-the-financial-sector
[30] Extracted from https://www.linkedin.com/pulse/how-does-dora-compare-uks-operational-resilience-framework-simon-cox-p81se/

risk management, incident reporting, digital operational resilience testing, third-party risk management, and information sharing. Included are detailed requirements to be taken when implementing the Act. It applies directly across all EU member states. It provides a harmonised set of requirements. It is overseen by various EU regulatory bodies, including the European Banking Authority, the European Securities and Markets Authority, and the European Insurance and Occupational Pensions Authority.

UK Operational Resilience: The UK framework focuses not only on digital resilience but also on the ability of firms to withstand, adapt, and recover from operational disruptions regardless of their source or cause. It covers a wider range of operational risks. It is a pragmatic and sequential approach that can be followed by financial or other institutions seeking to achieve compliance. It is implemented by the Bank of England (BoE), the Prudential Regulation Authority (PRA), and the Financial Conduct Authority (FCA).

The BoE process is summarised:

- Identify important business services (IBS) which will have measured availability.
- Set impact tolerances for these services which define how much disruption can be absorbed before intolerable harm is inflicted on the users of the services.
- Undertake regular testing against severe but plausible operationally disruptive scenarios to identify vulnerabilities.
- Take mitigating action so that services can remain within tolerance.

This four-stage process describes “what” but does not specify “How” the Important Business Services are to be identified.

Example: Assigning Important Business Services at an Insurance Company

Defining Important Business Services is something which must be done with business executives; the key tests for important business services are customer impact, business impact, and wider systemic impact.

Many organisations have prioritised their sales applications over service when it comes to digitisation, frictionless customer experience and process automation.

However, as, for example, many insurance companies have determined, it is the claims and repair services which are the important business services, enabling existing customers to access the services for which they have paid,

and minimising the impact on them by having a resilient 24x7 service available.

The focus on resilience and Important Business Services brings these applications into sharper focus – it tests an organisation's holistic perspective on resilience and customer service. In terms of prioritisation of resources and funding, the focus on important business services (and the regulation behind them) raises the importance of these services, and leads to productive discussions.

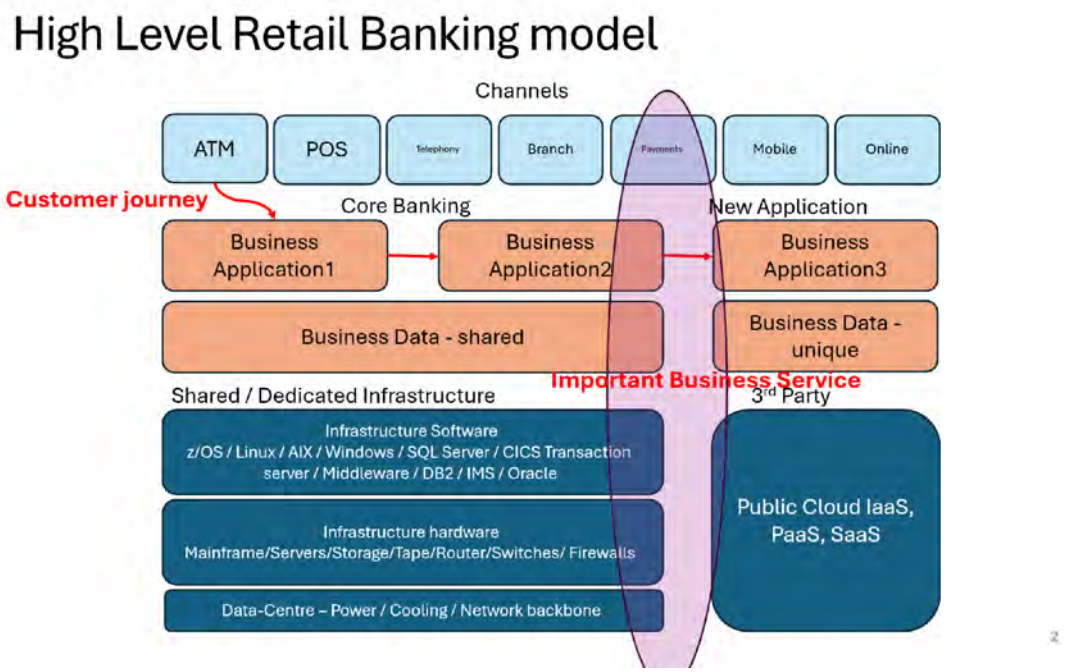
Checklists for defining an IBS in financial services are described for instance by The Investing and Saving Alliance.[31] Although some aspects may not apply in other sectors, we include the checklist here.

Checklist: defining an IBS

- PR01. Business services should be defined consistently to allow an assessment of systemic impact and to facilitate cross-industry comparisons.
- PR02. A business service must have a clearly defined external end user / set of users ("customers & consumers") which allows for the identification of both distinct services and "instances" of such services where needed.
- PR03. A business service must be provided to an entity external to the firm or group. Shared and internal services that are fundamental to the provision of the business service should also be captured, mapped, and tested.
- PR04. Business Services can be distinguished from supporting services or capabilities if they could be considered as providing value to consumers on a stand-alone basis. If a service has no consumer value on its own then it is part of another business service.
- PR05. Business services should be described in a way that is agnostic of the means of accessing the business service.
- PR06. For a business service to be valid the firm must be accountable for the provision of the service delivery. If there are any activities in which the firm acts only as an introducer, broker, or intermediary, regardless of the branding of the service, then the activity does not need to be included as a business service.

[31] See https://www.tisa.uk.com/wp-content/uploads/2022/02/TISA-Important-Business-Services-Guide-November-2021.pdf

The definition of IBSs by the Board allows the organisation to prioritise focus and investment. The diagram below illustrates that an IBS will often incorporate software and systems from a range of internal and 3rd party sources,



1.4.3 IBS and Business Continuity

The definition of IBSs allows for cross-organisational focus on the continuity of provision of key business services and with defined extent of maximum outages.

In some organisations, the definition and planning for achievement of continuous delivery of crucial services has been the domain of the Business Continuity function. In other organisations, Business Continuity functions have been focussed on operation of periodical simulation or practice exercises.

The use of IBS as the unifying factor for preventing service outages, or minimising their duration, ensures a focus on user impact of any outages, and provides an umbrella for wider engagement across the organisation.

1.4.4 Other regulatory frameworks

CRA

One regulation having increasing effect is the EU's Cyber Resilience Act (CRA)[32]. As IT systems integrate an increasing array of other subsystems, some are

[32] CRA - EU's Cyber Resilience Act - aims to safeguard consumers and businesses buying software or hardware products with digital elements.

covered by the CRA Act. This covers all the electronic devices that communicate data. This includes the Internet of Things, ATMs, and other data communication devices. The CRA requires reporting and disclosure of known vulnerabilities of such systems. These devices will incorporate software, and it is often not possible to establish the provenance of the embedded software in order to satisfy the regulator.

AI

AI adds complexity, but does not change principles. Under UK law, AI itself cannot be held liable. Instead, accountability sits with the individuals and organisations under existing legal frameworks, including negligence, contract, data protection and equality law. So, for instance if an organisation's IT system violates GDPR, that organisation is liable whatever the cause of the breach. The Equality Act 2010 applies where an AI system produces discriminatory outcomes, including in recruitment, lending, or service provision.

The direction of UK law is clear. In its draft Legal Statement on Liability for AI Harms,[33] published in January 2026, the UK Jurisdiction Taskforce (UKJT) reaffirmed that AI has no legal personality under English law and cannot be held liable for harm. Instead, responsibility rests with the individuals and organisations that design, deploy and use AI. In practice, courts are likely to treat AI as a tool, with liability determined under existing principles of negligence, duty of care and other applicable laws[34].

The EU has also passed new directives and amended existing ones to address AI liability, while Canada has drafted a new law to address this issue[35].

Conflicting regimes

There are different regulatory regimes covering automotive, shipping, and many other industries. As IT becomes more pervasive, conflicts of regulatory regimes will occur.

[33] Legal Statement on Liability for AI Harms,
[34] https://www.oneadvanced.com/resources/who-is-responsible-for-ai-mistakes/
[35] See https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2026.1709945/full

1.5 Standards, Guidance

1.5.1 Why standards are important

Standards provide a basis for sharing expectations about a product, process or service. They provide a common language, essential for instance in supply chains. They can also be the basis of certification, and applied to products, processes, or people.

Links to relevant standards are in the Annex. Note that the status of the standards and regulations here and in the Annex is correct at the time of going to press – August 2026.

1.5.2 Why standards are not a magic bullet

Standards play a key role in setting expectations and scoping the issues that underpin resilience.

However, implementing standards alone won't create a strong organisational response to resilience challenges for two reasons:

- No set of standards is comprehensive.
- No standard totally fits the unique resilience needs of each organisation.

Issues in practice include:

- Getting all stakeholders to reach a common understanding of specific standards is a costly and tricky process.
- Many organisations may lack skills, so they will depend on short-term consultants.
- Even after the organisation puts standards in place, keeping them up will need extended effort.
- In many organisations, figuring out the payoff from these efforts can be hard. So, they often don’t get enough resources when needed.
- Few organisations depend on outside groups to certify their compliance. This weakens the push behind implementing or maintaining standards.

In summary, standards provide useful guidelines and frameworks. They help improve organisational and technical resilience. However, they do not fully address the issues outlined in this Handbook.

1.5.3 Standards bodies

International Standards Organisation (ISO) and International Electrotechnical Commission (IEC) are international. ISO defines standards which are often the

basis of national standards, incorporated in national legislation. IEC prepares and publishes international standards for all electrical, electronic and related technologies. The IEC also manages global certification for equipment, systems and components.

1.5.4 ISO standards

Relevant ISO standards cover

- Software product certification
- Process certification
- Quality standards and guidelines – operational performance:

1.5.5 People certification

The IT profession is anomalous among the engineering professions in that, in the UK and elsewhere, practitioners do not have to be licenced. It is also the case that professional and academic qualifications in digital subjects quickly become out of date.

The rapid development of AI has raised the visibility of concerns about software failures and their consequences. The concerns include being able to identify responsibility and accountability for systems, about safety, 'bias' and ethics. Legislation targeted at regulation of AI as a whole[36] could lead to licencing of AI software engineers.

A number of professional associations, IT suppliers and training organisations provide certification on IT related topics: some are listed under *2.3.6 Skills*

1.5.6 Advice

Successful use of standards relies on[37]

- *Senior management commitment for time, budget, targets and audits*
- *Knowledge of the standard by eg talking to other organisations who have used it*
- *Stakeholder engagement and communications*
- *Keeping adequate records on processes and metrics*
- *A joined-up approach across affected silos*
- *Adaptation of internal processes where needed.*
- *Ongoing continual improvement after first wave.*

[36] https://www.techtarget.com/searchitoperations/feature/The-promises-and-risks-of-AI-in-software-development
[37] https://amtivo.com/us/resources/insights/7-common-iso-9001-standard-implementation-mistakes/

1.6 What the Board should expect to see

Boards, leadership, and engineering teams should expect the following answers to the questions posed at the beginning of this chapter, following the descriptions in this chapter:

Do we accept that IT failure is inevitable in complex, tightly coupled systems?

> Channels to engage the C-suite on business priorities and risk, and communicate technical exposure in business terms, for example a cost / time-to-fail curve, see Chapter 2.

Do we recognise the growing impact of IT failures, from both cyber-attacks and from ineffective change management?

> Lines-of-defence model: day-to-day controls (ISO 27001, ISO 22301), internal audit, management controls/remediation, and independent external audit, see Chapter 2.

Have we identified our Important Business Services (IBS) and set Impact Tolerances and availability targets that define the maximum disruption we can absorb before intolerable harm reaches users?

> IBS Register: mapping for every IBS to its supporting systems, dependencies, and data flows, with named owners and defined external user groups, reviewed at least annually.
>
> Impact Tolerance Statements: mapping Impact Tolerances for every IBS and underpinning systems into engineering targets for each service and it's data. Internal operating level agreements for networks, server infrastructure, storage database services and telephony, or supply contracts are built around the impact tolerance requirements.

Do we recognise that the UK Cyber Security and Resilience Bill is leading towards Board-level liability for service availability, not only for CNI and Operators of Essential Services?

> See section 1.4.1

Can we demonstrate to a regulator that resilience investment is proportionate to each service's importance, accepting that not all services need high availability and that over-engineering carries its own cost?

> Regulatory Applicability Assessment: a documented determination of which instruments apply to UK Cyber Security and Resilience Bill / NIS, DORA, FCA/PRA Operational Resilience and include the obligations/risks arising.
>
> Standards Conformance Baseline: a mapping of current practice to ISO 22301, ISO/IEC 25010 and 5055, and ISO 31000, with identified gaps, optionally include remediation plans to achieve compliance.

Have we understood that returning incorrect data can be a more severe failure than returning no data, and that reputational and customer-trust damage from a visible outage should be quantifiable, as in the NIS Framework?

> Availability Profile Map: a per-IBS assessment against the four degradation modes — latency, error rate, hard outage, and data integrity.

Engineering resilience for IT systems: Chapter 2 – Making resilience a reality

2.1 Purpose of this chapter[1]

This chapter is about the prerequisites to resilience: how to align the organisation to meet the need articulated in Chapter 1. The CTO family is normally responsible for the planning and implementation of resilience measures. They link to strategy and business analysis, customer service and marketing, HR and finance, as well as IT.

2.1.1 Key messages

Resilience is an organisational discipline, before it is a technical one. Systems will fail: how the organisation absorbs the failure depends on

- governance,
- culture,
- skills,
- human responses prepared in advance,
- as well as technical factors.

Throughout, the chapter has aimed to clarify accountability, ensure reputation is treated as a measurable risk, and prepare people and systems for the day the unthinkable happens.

Questions the CEOs and Boards should ask of CTOs

- Are our governance structures fit for purpose to achieve resilience for our IBSs? Are they documented and widely understood?
- What are our risks on the IT-related "nightmares"; data corruption or loss, supply-chain failure, loss of an Important Business Service?
- Have we established a Just culture in which staff can report mistakes and near-misses without fear of blame?
- Do our AI adoption policies treat AI systems as requiring the same scrutiny as any mission-critical software?
- How can we improve our skills for resilience?
- Are we missing any silver bullets of investment in IT to improve resilience?

[1] See https://www.gov.uk/government/publications/government-cyber-security-strategy-2022-to-2030/government-cyber-security-strategy-2022-to-2030-html

2.1.2 Topics in this chapter

- Responsibility, governance
- Culture, skills and communication
- Trade-offs between risk and cost
- Implementing resilience in IT systems
- AI opportunities
- What the Board should expect from the CTO
- Key takeaways.

2.2 Responsibility, governance

In Chapter 1 we discussed the Board's role in the oversight of risk. The organisation needs to operate governance at three levels: Strategic, Operational and Tactical.

2.2.1 Strategic Governance: Board

In Chapter 1 we identified the core responsibilities of the Board for resilience. Supporting the Board, the CTO family should ensure:

- A team including CTO family and Business Operations with the authority to manage organization-wide digital resilience.
- Business Continuity Steering Committee: A committee of decision-makers and technology experts chartered by the executive team. It provides strategic oversight of the Business Continuity Strategy, Disaster Recovery, and overall risk mitigation.

2.2.2 Operational Governance:

Management and cross-functional integration: translating board strategy into processes and policies. This is a key role of the CTO family. This involves a "Defend as One" approach[2] that transcends organizational silos.

For historic reasons, operational governance is often implemented by several teams with overlapping and sometimes conflicting functions, e.g.

- Business Continuity Management Team: An operational group (often synonymous with the Emergency Management Team) of senior managers from all functional areas. Their role is execution of continuity and recovery plans.
- Secure and Resilient by Design Team: this team establishes common profiles and architectural patterns to ensure that new services, support, and standards are "secure by design". For complex organizations this includes compartmentalization to localize the impact of IT failures and limit access to potential bad actors.

Other functions are embedded across the organisation eg

- Supply Chain and Third-Party Risk Management: see Chapter 3
- Risk Categorisation and Important Business Services: The enterprise architecture map (see Chapter 4) shows interdependencies and allows

[2] https://assets.publishing.service.gov.uk/media/61f0169de90e070375c230a8/government-cyber-security-strategy.pdf

organisations to focus operational resilience efforts on Important Business Services.

2.2.3 Tactical operations across silos:

The tactical tier involves the on-the-ground capability to anticipate, detect, protect against, and recover from disruptive events. Co-ordination across silos is a CTO family function. Topics include:

- Business Continuity Teams and Plan Administrators: Designated individuals responsible for the administration and on-the-ground execution of specific services, departmental or facility Disaster Recovery and Downtime plans.
- Downtime Procedures and Interoperability: Tactical teams to maintain standardized, downtime procedures that enable critical business functions (e.g., patient care, payments) to continue safely during a system outage. They also cover the recovery systems, plans and workflows to prevent data loss, see Chapter 6.
- Testing, Exercising, and Rehearsal: Regular "dry run" exercises to test response and recovery plans using scenarios drawn from threat intelligence, see Chapter 6.
- Continuous Threat Detection and Hunting: Maintaining robust capabilities (e.g., through Security Operations Centres) to actively monitor networks, detect cyber events, and anticipate probable attack methods.

2.2.4 Governance across silos: RACI

Governance across silos relies on assignment of roles and ensuring the implementation of plans. One basic tool for assigning roles across functions or departments is RACI.[3] RACI stands for the four roles in relation to a target or plan:

- Responsibility for completing the task of implementing the plan or hitting the target.
- Accountability - answerable for the correct completion of the deliverable or task, ensuring the prerequisites of the task are met, and delegating the work to those responsible.
- Consulted - those whose opinions are sought, such as subject-matter experts.
- Informed - those who are kept up-to-date e.g. on completion of the task or deliverable, if their work processes change.

[3] https://www.bcs.org/media/03hn1dju/availability-the-role-of-raci.pdf

RACI is used extensively by consultants and suppliers working with organisations, as a tool to understand who is tasked to deliver what; and hence to protect their reputation. In-house use can be helpful for cross-silo working.

Responsibility

Responsibility for implementing resilience lies across the organisation, and may be led by Business Continuity or the CTO family.

Metrics for resilience are discussed below.

Accountability

Accountability for setting the organisation's required resilience capability is with the Board. The Board will often be briefed by CTOs or other senior people who will describe options, after consultation with various departments - IT, business continuity, finance, and more - under a unified approach. This depends on a 'common language', for instance IBS and Impact Tolerances, aligned with strategic objectives.

Accountability, unlike responsibility, cannot be split across departments.

Implementation may be monitored through establishing controls at various levels: eg

- First: Control frameworks and day-to-day controls via for example implementation of Information Security / Business Continuity Management Systems e.g. ISO 27001, ISO 22301[4].
- Second: Impartial regular internal audits by competent auditors e.g. IRCA[5] Certified Lead Auditors. These review the effectiveness of implemented frameworks and controls.
- Third: Prompt remediation by top management of internal audit report findings.
- Fourth: Independent external audits by competent auditors e.g. IRCA Certified Lead Auditors. These use knowledge of industry practice not generally available internally.
- Fifth: Prompt remediation by top management of external audit report findings.

[4] https://www.iso.org/standards.html

[5] https://www.imsm.co.uk/blogs/what-is-irca-and-why-is-it-important/

Example: UK public sector plan for Accountability

In addition to the Cyber Security and Resilience Bill, the UK Government has issued a Cyber Action Plan for the UK public sector which states:

"Government-wide cyber risk will be owned by the DSIT Permanent Secretary as Government Technology Risk Owner".

And for Organisational cyber risk ownership:

"The Accounting Officer is the senior official (Permanent Secretary or CEO) with overall accountability for an organisation. This includes personal accountability for the cyber risk of that organisation.

For LGD's, that accountability also extends to the ALB 's and sectors which fall under their financial responsibility. For all government organisations, it extends to appropriate assurance of the cyber security and resilience of their suppliers."

Consulted
People consulted may be subject experts, stakeholders inside the organisation, customers, suppliers, regulators.

Effective consultation is essential in setting achievable and desirable availability characteristics for the organisation.

Informed
Effective communication with customers can prevent unrealistic assumptions. It can also mitigate the impact of IT outages. This might be by advising of alternate ways of getting the service, or expected times of resumption of service.

The Annex: RACI for Resilience Deliverables shows a possible allocation of each of tasks in support of resilience, across departments. It is intended to provide a basis for evolution of specific role assignments in any given organisation, rather than being a final prescription.

2.3 Culture, skills and communication

2.3.1 Communication between management and IT people[6]

An organisation's resilience depends on how people, organisations and incentives support learning from failure, and approach risk. This requires teams to talk openly about uncertainty, risk and failure, in a culture of safety and trust.

IT leaders are aware of the immediate risks of digital failures. This makes them potential initiators of discussions of resilience to engage other stakeholders. This needs recognition of two things.

- Firstly, however much resilience planning we do, the possibility of a failure remains.
- Secondly, people's judgement and experience are invaluable tools in responding to a failure.

The key cultural challenges that need to be addressed are as follows:

- Blame vs learning: many organisations still default to asking 'who is at fault' which prevents reporting, discussion and collaboration. This is particularly relevant where the system is supplied and/or maintained by multiple parties.
- Low psychological safety; not only should resilience not be seen as purely an IT Ops responsibility, but all teams should be encouraged to ask themselves the questions 'what happens if it fails? And how will I know if it is likely to fail?' This should be seen as a way of providing an early warning system and contributing to the overall resilience of the systems.
- Legacy beliefs about heroics; some cultures celebrate firefighting and individual heroics, rather than systematic prevention and adaptation. This can lead to a relative lack of appreciation of design activities and choices which reduce the need for heroics.
- Silos within the IT organisation as well as across wider functions are real. They exist in both large and small organisations. While large organisations can offer education and training to support communication, this is less likely in small organisations. This Handbook looks to communicate across silos, for instance by cross references between chapters.

[6] Excerpted from Chapter 7 of Resilience of Services, https://londonpublishingpartnership.co.uk/books/resilience-of-services-reducing-the-impact-of-it-failures/

2.3.2 Culture to support resilience

Leaders need to encourage curiosity, transparency about error, and willingness to learn to support resilience.

In addition, leaders need to understand that resilience costs money and effort, and need to be prepared to make trade-offs between new business functions and resilience.

Organisations can promote the culture of 'safe spaces' for people to openly discuss failures, outages and the early signs indicating instability or risk. One model may be to draw on some of the practices common in health and safety where there is a positive obligation to call out issues.

Organisations can establish a 'What If?' approach to planning for future potential scenarios to ensure they have adequate protections in place. One way is to define and conduct 'pre-mortem' examinations of large-scale failure to address managing organisational risks.

2.3.3 Communication tools

Some of the approaches that can build an appetite for broader engagement:

- Fictional case studies are widely used in management education to highlight specific elements of business practice. They reflect actual experience but simplify the context. This reduces the coincident factors that are ordinarily present in case studies of actual companies and may deflect from the main point.
- The CTO or CIO can devise a plausible simulation of failures. This allows consequences to be identified and traced to current processes, as well as simulating realistic measures to cope with the crises and ultimately recover. This can be implemented with less senior staff and the participants can generate feedback and ideas for improvement.
- The CTO or CIO can have in place a 'prepacked' workshop plan to offer to the C-suite immediately after a major service outage hits the organization or a competitor. This could also be triggered by the media's flagging of major data breaches or service outages and their impact.
- Graphs are a useful way to communicate. IT Leaders could use a Cost/time to fail graph. This shows that greater investment in service resilience buys a lowering of the risk of service failure. It also shows that the risk can never go to zero. This visibility of the organisation's calibration of risk could reduce insurance premiums and could in the future be a requirement to obtain any insurance at all.

2.3.4 Thinking the unthinkable – anticipating catastrophic failures

Much of this Handbook is about avoiding IT failures and/or reducing their impact on services to users.

Organisations also need to explore the unthinkable. What will happen when a major disaster occurs? Your systems will be down; there has been a major cyberattack or system corruption, or something catastrophic has happened to one of your key service providers, or a facility taken out by extreme weather with total loss of data. Whatever you will do to restore normal operations will take days, weeks and even months to carry out. Yet you will still have responsibilities to your customers and your stakeholders both within the company and outside.

What will you do? And, more to the point, what will you wish you had done to improve the outcomes you're facing now? How can you prepare your organization and systems for the consequences of a software disaster? This is important enough that some examples of how organisations coped are included in Chapter 6 on Response, recovery and learning.

2.3.5 CIOs Body of Knowledge- what has changed

A classic "Body of Knowledge" guide for CIOs by Dean Lane, published in 2011[7], had 28 chapters, sub-divided into People, Process and Technology. The responsibilities of an IT Leader still include these three but in addition

- the need to manage people, process and technology now extends to third parties;
- and IT Leaders need to explicitly manage Availability, through improved resilience.

People:
The chapters on People in the Body of Knowledge are still relevant, covering as they do teamwork, recruitment, people development, skill building, staff retention. However, as the emphasis has changed from inhouse development teams to external suppliers for new systems, the management task is to manage across organisations and silos. For instance:

- creating channels and processes to engage with the C-suite to understand business priorities and risk, as digital services become ever more important to operations;
- managing the people and activities in the software supply chain;
- managing procurement and contracts;

[7] Dean Lane, 2011. The Chief Information Officer's Body of Knowledge, John Wiley & Sons

- developing and maintaining good external and industry networks to share information on suppliers and products, as the supply chain becomes more complex.

Process:
Chapters on strategic alignment, project requirements, development, quality and project management and review, compliance, security and hackers, outsourcing and offshoring are still relevant. Additional skills and processes are needed for

- information sharing as staff work from home and information is shared with workers not covered by the organisation's employment contract;
- management of third parties – outsourcing, and procurement and contracts through a complex supply chain.

Additional process capabilities are needed for managing Availability

- support for the definition of IBS and Impact Tolerances;
- definition of stress testing scenarios and stress testing against Impact Tolerances;
- preventive system maintenance in a 24/7 operation;
- proactive upgrades to improve resilience;
- processes for recovery – from partial or total outages.

Technology:
The chapters on technology cover information security, business continuity planning and "Why being involved in your web strategy matters". This last heading is a useful reminder of a major change in the role of IT systems since 2010. IT is central to most B2C services to customers, taxpayers or citizens. This creates further demand for resilience as customers, taxpayers and citizens come to rely on services.

Information security and business continuity planning have heightened importance as the focus on Availability increases.

2.3.6 Skills

Many of the new skills above are not yet included in most education and training courses for IT professionals. The role involves values about doing the right thing rather than performing to nominal goals; and an ability to move between larger system perspectives and details of implementation. The characteristics of people to deliver availability management are not easily inferred from a CV or specific qualifications.

Improving internal capacity involves a process of upgrading of learning and skills to gain 'soft' skills, address competencies, and provide mentorship. This often benefits from the use of external standards and qualifications.

Education, Skills, and Professionalism: To sustain governance, the organization must invest in its people, aligning with frameworks like the Skills Framework for the Information Age (SfIA[8]).

Competency Frameworks: There is a need to add "resilience and availability" competencies to standard IT profiles (Strategy, Architecture, Supplier Management, and Service Support). Useful baselines are published by Prospects[9].. BCS is providing input to SfIA on the extensions needed to SfIA.

Sector-Wide Learning: There are a number of sources of specific training on for instance

- cyber security[10],
- risk management for leaders, boards, and the wider workforce[11],
- business continuity[12],
- IT service management[13],
- Leadership[14].

The major service suppliers (Microsoft, Amazon, Google, SAP, Oracle) provide training on their products.

2.3.7 Certification

Many universities award degrees in computing, while increasing numbers of entrants use entry routes through certification by suppliers and by a range of training organisations. These also provide specialist training relevant to Resilience Engineering. Some are listed below

Check list: Bodies providing relevant certifications

1. Amazon Web Services (AWS)
2. BCI
3. BCS – CITP
4. CompTIA
5. CSSP
6. GIAC
7. Google

[8] https://sfia-online.org/en
[9] https://www.prospects.ac.uk/job-profiles/browse-sector/information-technology
[10] https://www.ncsc.gov.uk/cyberessentials/overview
[11] https://www.theirm.org/training/
[12] https://www.thebci.org/certification-training.html
[13] https://www.itil.org.uk/
[14] See https://www.jbs.cam.ac.uk/

8. ITIL
9. ISACA
10. Microsoft
11. UK Resilience Academy[15]

2.3.8 AI –skills needed to use it effectively

Building an effective AI Enabled Operations team needs a blend of operations, data/ML, automation, and collaboration skills in addition to the classic skills and expertise available in many resilience engineering teams today.

Checklist: skills and expertise of an AI enabled Operations team.

1. IT operations and observability
Strong understanding of infrastructure and applications: servers, networks, cloud, containers, microservices, and how they fail in production.
Incident and problem management, SRE practices, and observability basics (logs, metrics, traces, SLIs/SLOs, alerting, post-incident reviews)
2. Data and analytics skills
Data literacy: ability to work with large operational datasets, clean and normalise logs and metrics, and recognise patterns, spikes, and anomalies.
Practical data analysis and visualisation using tools such as Kibana, Grafana, Power BI, or similar
3. AI and machine learning fundamentals
Understanding of key ML concepts relevant to operations and familiarity with common ML libraries and platforms to experiment with and operationalise models,
4. Programming, scripting, and automation
Proficiency in at least one general-purpose language as well as scripting and automation skills to connect monitoring systems, build workflows, and implement self-healing actions
5. Platform and tool familiarity
Hands-on experience with monitoring/observability platforms and AIOps tools e.g. understanding how to design telemetry pipelines and integrations
6. Soft skills and ways of working
Strong problem-solving and analytical thinking to diagnose complex, cross-system issues.
Cross-team communication and collaboration with SRE, DevOps, security, and business stakeholders; ability to explain AI-driven insights and influence process changes.
Change-readiness and continuous learning mindset.

[15] https://ukresilienceacademy.org/develop/professional-pathways/resilience-engineering/

2.4 Trade Offs Between Risk and Cost

2.4.1 Risk

Three factors describe the risks that an organisation is exposed to.

- risk assumptions, eg[16]
  - Key customer says "This invoicing package is risky because it did not work at my last company"
  - Create an assumption "The selected package will deliver the required functionality"
  - Tease out more assumptions from the key customer, and teams implementing or supporting the package and the supplier.
  - In general, it is possible to reduce each of these risks by further investments in negotiating user assumptions and requirements.
- risk constraints, eg time, scope, cost, resources/skills.
  - Shortening development and deployment time, increasing scope of services and their availability and improving resources/skill will generally require increases in investment.
- risk tolerances, assessed by calculating metrics such as
  - Value at Risk (VaR),
  - stress testing results,
  - the potential financial loss in different risk scenarios
  - Higher VaR and prospective financial loss generally contribute to a need for greater investment while testing is itself a cost that is justified to the extent that it can mitigate risk.

The aim is to help organizations understand their capacity to withstand adverse events financially. The risk may be different for each of the business services within an organisation and different business organisations may value risks and the costs of its mitigation very differently.

[16] https://www.bcs.org/media/4665/assumption-based-risk-management-50319.pdf#:~:text=%EF%82%B4%20Given%20each%20risk%20determine%3A,company%E2%80%9D

2.4.2 Risk priorities.

Many organisations publish a list each year: e.g.[17] for 2026

Checklist of Risk priorities for 2026
Evolving cyber security threats
Zero trust security
Technology resilience and incident response
Cloud governance and security
Generative and autonomous AI
Evolving digital regulation and compliance
Critical third-party supply chain risk
Transformation
Data governance
Deepfakes and disinformation threats

For any particular organisation, it may be helpful to prioritise this list according to the best estimates of the benefit to cost ratio of making improvements.

2.4.3 Translating Impacts into internal metrics

Some of the most commonly tracked internal metrics are

- Mean time between failures (MTBF),
- Mean time to recovery, repair, respond, or resolve (MTTR),

User impacts
These can be estimated using the NIS Framework metrics.

Availability is measured in lost user hours: so, for a business service this will be the sum of all the outage times (MTTR), each multiplied by the number of users. This may depend on which business service is unavailable:

- A banking application serving thousands of customers vs one serving millions;
- An outage of a water system may affect millions of customers, but how often do they switch on the tap?
- An outage of a booking system may affect millions of potential customers though only a few get tickets – what is the right number of users? how can this be estimated?

[17] https://www.grantthornton.co.uk/insights/technology-risk-trends-your-key-priorities-for-2026/

The term availability is also used in some contexts to cover timeliness, correctness, and reliability of the services to users, not just when the service is totally offline.

Integrity, authenticity or confidentiality of data is measured in the number of data items involved. What is the cost to the person involved? It could be argued that the most significant cost is to the organisation if it loses data[18]. But customers also suffer damage from unauthorised access to their personal data.

Risk to public safety, to public security or loss of life: the costs of these are used extensively by insurers. For instance, the Health and Safety Executive publishes estimated unit costs for work related illness[19].

Material damage is the financial loss to a (large) users of the service.

2.4.4 Costing and budget – how to construct and justify

Resilience has a cost. This is apparent in budgets. Risk is more difficult to quantify. Quantification of risk involves assigning numerical values to

- probability - the likelihood of the risk behaviour occurring;
- impact - the financial or operational consequences if the risk occurs.

Probability likelihood is typically communicated through qualitative metrics such as red, orange, green or “very low,” “low,” “moderate,” “high,” and “very high.” Assigning numeric probabilities to IT risks can use ranges of MTTR and MTBF, if the service is dependent on a single system that is measured.

A common practice in the medical device industry is to use a conservative value of 1 as the probability of software failure[20].

This example uses a bridge analogy to calculate impact[21].

Calculation of Impact Tolerances: Firm owns a bridge
A firm owns a bridge across a river that enables people to drive from one side to the other. The firm identifies an important business service it provides as: crossing the river. Each vehicle crossing the bridge represents

[18] e.g.https://conosco.com/in-the-news/marks-and-spencer-what-it-means

[19] https://www.hse.gov.uk/statistics/economics/eauappraisal.htm

[20] https://naveenagarwalphd.substack.com/p/ama-4-how-to-estimate-the-probability

[21] See https://www.pwc.co.uk/forensic-services/assets/documents/white-paper-on-impact-tolerances-feb-2020.pdf

a transaction, and drivers pay a fee (a toll) to do so. The total capacity of the business service is limited to four lanes and is represented by how many vehicles can cross at any given time.

Disruption to the bridge may force the firm to close one or more lanes, impacting the ability of vehicles to cross and limiting the capacity of the service. The firm can communicate with customers in various ways including smart motorway signs and local media.

The firm is regulated by two authorities:
Regulator A focused on the potential harm to consumers;
Regulator B focused on the firm's financial viability and the stability of the market.

To calculate impact tolerances, they need to address both regulators. This is done in 5 stages.

1. The bridge is a key piece of infrastructure for the region with no real alternatives offering a comparable service. The firm breaks the service down into process steps.

- Payment requested of the driver
- Driver's payment is processed
- Vehicle approaches bridge barrier
- Bridge barrier is raised
- Vehicle completes crossing

It can then identify the resources that enable the firm to deliver the service and monitor performance, and impact, as follows:

- Technology: Traffic flow sensors, CCTV, card payment facility, number plate recognition
- People: Maintenance crew, toll collectors, traffic flow controllers
- Facilities: Road bridge, toll booths, remote control centre
- Information: Number plate registration, payment card details
- Third parties: POS payment provider, maintenance contractor firm, driver and vehicle licensing agency.

2. The firm identifies a series of metrics which describe the functioning of the business service:
Measures of direct impact of operational disruption

- Total number of vehicle crossings completed.
  a. Of which, number of emergency services vehicles completed.
  b. Of which, number of trade vehicle crossings.
- Average time to complete a crossing (including queueing at the toll point).

- Average wait time to pay at toll point.
- Total revenue collected at toll point.

Measures of indirect impact of operational disruption

- Loss data from fraudulent transactions.
- Regulatory fines or contractual penalties applied as a result of breaches (e.g. on availability of service).
- Social media sentiment (positive and negative).
- Newspaper coverage (column inches).

Key risk indicators to help predict future operational disruption

- Historic incident trend data including root cause analysis.
- Average speed of vehicles crossing the bridge.
- Volume of payments by type: cash, debit card, and credit card.
- Average payment processing time (and variability) by type.

3. Disruption to the bridge could have a range of consequences:

- Citizens are unable to complete journeys
- Vulnerable lives are put at risk where emergency services are impeded.
- The flow of goods is disrupted and services are not completed.
- The firm's profitability is hit by lost revenue and additional costs through restorative repairs, penalties and fines on a low-margin business service.
- The firm's cashflow is jeopardised as toll receipts are interrupted.
- The firm's reputation is damaged, impairing its ability to win future contracts.
- The region's reputation is damaged, impairing its ability to attract investment in businesses and trade, which has wider socio-economic implications (e.g. job growth).

To define impact tolerance limits, the firm assumes that the business service is unavailable and estimates the maximum tolerable level against the key metrics: 1) number of hours of delays to emergency services vehicles; and 2) loss of revenue during the disruption.

Regulator A: Harm to vulnerable consumers 'Delay to journeys' is a tangible impact with potentially severe consequences. Emergency services are forced to find alternative methods of crossing, adding one hour to journey times. The threshold is determined at a point where the risk of loss of life is deemed to have increased substantially based on analysis of available data. The resulting impact tolerance = 2 hours of total disruption (i.e. zero capacity) and >50 hours of delayed emergency services journeys.

Regulator B: Safety and soundness of firm Financial loss is captured through lost revenue levied as a result of contractual or regulatory breaches. The threshold is determined at a point where the financial loss will have a detrimental impact on the firm's viability.

4. Scenario testing
On Monday 20 May 2026 at 07:00 two lorries collide and cause a pile-up with three other cars. The wreckage spans all lanes of the bridge with no way for other vehicles to get through. One of the lorries partially ruptures the wall on one side of the bridge. An unidentified liquid chemical has started leaking from the cargo of one of the lorries. Several of the passengers are trapped in their vehicles. At this point it is not known how serious the injuries are.

Assumptions made by the firm:
- The incident was identified by the control centre via remote cameras on the bridge
- There have been no explosions or fires
- There is a weather warning for high winds.

Actions the firm could take:
- Contacting emergency services to alert them of the incident
- Closing the toll gates remotely to prevent further vehicles from entering the bridge.
- Alerting local media to discourage further travel via this route
- Dispatching engineers to assess structural damage to the bridge
- Arranging for lifting equipment (e.g. crane) to remove wreckage
- Arranging alternative crossing routes to support customers (e.g. chartering a ferry)
- Identifying customers who are trapped on the bridge, especially those that are vulnerable
- Seeking to reopen individual lanes at the first opportunity.

Regulator A: *Harm to consumers* The initial impact is not easily mitigated as vehicles cannot cross the bridge until the wreckage has been partially cleared. Contingency measures therefore take several hours to become effective, reducing the overall impact significantly but still breaching the tolerance threshold.
Regulator B: *Safety and soundness of firm* As a result of the disruption, the firm does not collect vital toll receipts. Costs for chartering a ferry initially increase the financial impact but as lanes start to open the overall impact is managed within tolerance.

5. Ongoing governance
Having completed the first cycle for an important business service, setting and testing the impact tolerances, the board signs off the resulting documentation and agrees updates to the firm's governance framework to ensure clear individual and collective accountabilities.

This would include discussion and agreement of:

- Self-assessment document to share with regulators (on request)
- Updates to existing risk reports
- Terms of reference for key committees.
- Summary of the operational resilience of each important business service.

2.5 Implementing resilience of IT systems

2.5.1 Resilience depends on more than IT

Operational resilience is the ability of firms to prevent, adapt and respond to, recover and learn from operational disruption.

A recent Delphi survey[22] on learning from failures in complex systems identified new skills needed across the organisation in order to underpin resilience. These covered maintenance, design, communication, leadership, quality management, risk management, training, compliance, human factors, organisational learning, decision-making, employee wellbeing, processes, resource management, change management, incident response, and continuous improvement. Not a very helpful list! But it does emphasise, as above, that a culture of cooperation across the organisation is an essential basis for resilience.

The also ignores what is clearly a major gap in skills: those needed to manage commercial relationships with suppliers in a way which supports the aims of the organisation, for instance for resilience. This is of crucial importance to the ongoing delivery of services, and is covered in Chapter 3, Supply Chain and Vendor Management.

2.5.2 Which services need what availability? Impact Tolerances

In Chapter 1 we introduced the concept of IBS and the associated Impact Tolerance. A tolerance metric is useful for all services, for assessing the impact of service outages on users, as in 2.4 Trade-offs above.

In regulated sectors, the regulator may set reporting thresholds which organisations could use to set Impact Tolerances.

Once Impact Tolerances are agreed, the duration of outage can be used to set the targets for the IT system metrics, for instance Recovery Time Objective (RTO) or MTTR:

- RTO is the target amount of time in which a feature must be restored to avoid unacceptable disruption. This will vary wildly between features depending on their importance and can be a useful metric in determining what level of disruption classifies as an incident.

[22] https://engineeringx.raeng.org.uk/media/p3no1hj1/learning-from-failures-in-complex-systems.pdf

- MTTR[23] Mean time to recovery, repair, respond, or resolve.

2.5.3 Planning and architecture

Given the characteristics of current IT systems, there is no silver bullet to ensure resilience.

A useful starting point to 'the ability to absorb and adapt in a changing environment' is looking to ensure graceful degradation: if one part of the system fails, the entire house doesn't fall down. This means:

- Defensive programming,
  Software which behaves in a predictable manner despite unexpected inputs or user actions, or the use of Application Programme Interfaces (API's) which serve as break points should unexpected data arrive. However, new ranges or volume of data entering the best of software can cause failure.

- Data validation
  A basic defence! For instance:
  - Check digit eg Barcode readers in supermarkets
  - Format check eg National Insurance number LL 99 99 99 L
  - Length check eg password as specified
  - Lookup table eg only seven possible days of the week
  - Presence check eg a key field cannot be left blank
  - Range check eg hours worked less than 50 and more than 0

- Minimizing the "Blast Radius"
  Isolating failures. Microservices or Modular Design ensure that for instance a failure in the "Payment" service doesn't crash other services such as the "Product Catalog" or "User Login" services.

- Using hardware with high availability (Zero Downtime)
  Workloads are spread across multiple geographical regions. If a data centre in London goes dark, traffic automatically shifts to New York without the user noticing.

These and other architectural approaches are discussed further in Chapter 4.

[23] https://medium.com/vodafone-uk-engineering/the-importance-of-software-resiliency-and-best-practices-to-improve-user-experience-87a6ec96ec97

2.5.4 Anticipation and avoidance

In addition to graceful degradation, resilience depends on anticipation and avoidance of outages in live systems, and the ability to recover after outages.

Anticipation and avoidance of outages requires

- Systems mapped against business services
- Defined priorities eg Important Business Services and Impact Tolerances.
- Monitoring, management and analysis – what data to collect, the use of dashboards, setting alert thresholds.
- Upgrading to new versions, adding new functions and users.
- Preventive maintenance, stress testing and chaos engineering

This is discussed further in Chapter 5.

2.5.5 After the failure outs services

Recovery plans are dependent on an understanding risk including liability, building on business analysis and mapping as for Anticipation.

Business continuity focuses on keeping the lights on and the business open in some capacity after outages. Incident response is all the activities that an organization takes when it suspects a security breach or other threat to the organisation. Both increasingly rely on Self-Healing & Automation, with the role of AI increasing.

- Auto-scaling adds more servers automatically during a traffic spike.
- Health Checks that automatically kill and restarting a "sick" instance of an application.

If the worst happens, cross organisational Disaster Recovery or Business Continuity Plans are called up. Preparation for implementing these include:

- War gaming, regular reviews and updates of the Plan, training of staff
- Regular Reviews and Updates
- Root cause analysis of failures, learning

An IT responsibility is maintaining backup applications, data and systems in secure off-site locations.

Recovery is discussed in Chapter 6.

2.6 AI opportunities

AI is becoming central to management of complex IT systems. In Chapters 3,4,5,6, we discuss the use of AI in the different roles involved in achieving resilience: some topics that span all aspects of resilience are captured here.

2.6.1 New systems/replacing legacy systems

Coding:
There is a school of thought that it will be faster, and more cost effective, to use AI to rewrite existing applications than to maintain existing systems. Much has been said about the ability of Claude to support COBOL modernisation: this is unpacked, and clearly articulated in a Thoughtworks post[24]. The current status in late 2026 appears to be that AI coding is currently of a quality delivered by a relatively new programmer, with known vulnerabilities visible[25].

The reality of resilience is that any software component has to be engineered into an end-to-end system, considering the architecture and operational aspects of the software as well as the characteristics of the code. Rewriting legacy applications can be part of a plan, but is not a complete plan.

2.6.2 Documentation of architecture

Documentation of legacy modules may be out of date or lost, with limited experience and expertise of them still in the organisation. AI can support the re-coding of the system, regenerating documentation, analysing the code, inputs and outputs, functionality and regenerating code. This is an area which is evolving rapidly with huge focus across the industry on not just cyber security, but identification of code vulnerabilities and code quality.

Tools can effectively map different aspects of systems. One program to map out the configuration of the existing estate is SolarWinds[26] . On the physical hardware, it gives an appreciation of CPU and disk spin consumption, etc. It's feasible, but it's costly[27].

Use of tools to document Enterprise Architecture is important in underpinning compliance demands for IBS.

---

[24] https://www.thoughtworks.com/en-gb/insights/articles/claude-code-cobol-modernization-reality

[25] Based on private discussions with the team

[26] https://www.solarwinds.com/

[27] Based on private discussions with the team

2.6.3 User and operator manuals

Some of the most extended outages[28] have been where the initial problems were exacerbated by untrained operators. Their actions took days to resolve/roll back. Several case studies appear in the Annex: Public Reports.

ServiceNow are pioneering the use of Generative AI to summarise manuals and recommend actions in the case of an outage[29].

2.6.4 Testing, QA and compliance:

The use of AI to automate testing is becoming widespread and broad, covering many of the associated processes. It has obvious application in filtering alerts and taking routine decisions such as engaging back-up systems.

AI can streamline regulatory compliance by automating checks, monitoring changes and providing evidence of due diligence. AI-based compliance extends its utility into real-time data analytics, which is critical for maintaining up-to-date and accurate compliance reporting. By continuously analysing transactions and operations, AI tools can immediately identify discrepancies or non-compliance issues as they arise.

The Annex AI Support for Enterprise Architecture discusses Testing in more detail.

2.6.5 AI-enabled operations

AI and AIOps (autonomous remediation and AI-assisted resilience engineering) can strengthen recovery by improving early warning, triage, and runbook execution. It is a lever for safeguarding availability and constraining the "blast radius" of future incidents. Early reports are that incident detection can be increased and MTTR decreased by automation, while human intervention remains essential for instance to move to degraded operation (DMOP, see Chapter 5 on Service Delivery).

Commercial observability platforms often offer broader integration across application, infrastructure, and security domains. Many now include AIOps capabilities to help correlate signals, reduce noise, and accelerate triage and

[28] See https://www.spiedigitallibrary.org/conference-proceedings-of-spie/13813/1381317/Handling-human-errors-and-management-fails-in-telecommunications-and-data/10.1117/12.3092122.full

[29] https://www.servicenow.com/company/how-servicenow-uses-servicenow.html

response—subject to appropriate controls, testing, and governance. Many consultancies are developing portfolios of applications to deliver combined anticipation and recovery functions, with the aim of reducing catastrophic outages.

2.7 What the Board should expect

Boards, leadership, and operational teams should expect the following to be produced and maintained as outputs of the governance, culture, and human-response disciplines described above, to answer the questions posed at the start of this chapter.

Are our governance structures fit for purpose to achieve resilience for our IBSs? Are they documented and widely understood?

Resilience Governance Charter: a documented three-tier governance model (Strategic, Operational, Tactical) with a designated Board Member for Resilience and a chartered Business Continuity Steering Committee.

Board strategy translated into operational governance: a “Defend as One” model that crosses silos rather than leaving teams with overlapping, conflicting remits.

Common language established and in use e.g. IBS and Impact Tolerances, that lets IT, business continuity, finance, and risk operate under a unified approach.

RACI Matrix for Resilience: roles assigned across IT, business continuity, finance, and suppliers, with the lines of defence defined.

Lines-of-defence model: day-to-day controls (ISO 27001, ISO 22301), internal audit, management controls/remediation, and independent external audit.

What are our risks on the IT-related “nightmares”; data corruption or loss, supply-chain failure, loss of an Important Business Service?

Risk and Cost Model: a documented method translating probability and impact into budget decisions, using NIS metrics and Impact Tolerance logic.

Channels to engage the C-suite on business priorities and risk, and communicate technical exposure in business terms, for example a cost / time-to-fail curve.

Have we established a Just culture in which staff can report mistakes and near-misses without fear of blame?

Just Culture Policy: a stated commitment to blameless post-incident learning, safe-space reporting, and near-miss capture.

Communications Toolkit: prepared simulation and wargame materials, a "prepacked" post-incident workshop, and a cost / time-to-fail visualisation for board engagement.

Do our AI adoption policies treat AI systems as requiring the same scrutiny as any mission-critical software?

Govern AI deployment across documentation, manuals, testing, release management and AIOps, capturing the uplift while assigning clear human accountability for AI outcomes.

AI Adoption Policy: Treat AI as software, deployment approvals align with internal change and release processes, human-in-the-loop for high-consequence decisions, maintained manual fallbacks, adversarial testing, and assigned accountability.

AI Enablement Register: documented AI use cases — documentation, manuals, testing/QA/compliance, AIOps, and contract management — with assigned human accountability.

How can we improve our skills for resilience?

Skills and Competency Plan: SfIA-aligned resilience and availability competencies added to IT role profiles, plus a defined skills baseline for an AI-enabled operations team.

Are we missing any silver bullets of investment in IT to improve resilience?

Architectural Principles Statement: a documented commitment to graceful degradation, blast-radius minimisation, and observability as design defaults is not a sliver bullet but could be a useful start point.

Engineering Resilience for IT Systems:
Chapter 3 – Supply Chain and Vendor Management

3.1 Purpose of this chapter

3.1.1 Key Messages

Many organisations do not consider the risks from temporary or long term failure of a software supplier.

For instance, if your supplier is based on services in the cloud:

- Could you get access to your data?
- How would you continue operating?
- How quickly could you be up and running again?

In Chapter 6 we discuss recovery in more detail – here we describe the processes to improve awareness and to defend the organisation. When an Important Business Service fails, the organisation fails with it, and the regulator, the customer, and the press will hold the organisation, not any suppliers, to account. No modern IT service is built, run, or recovered by one organisation. It relies on a chain of vendors, cloud platforms, open-source components, and managed services, most of them invisible from the boardroom and many of them changing daily.

Whilst the software supply chain is one risk among many, its management is a discipline in its own right. This chapter assumes that the organisation has in place a framework for supply chain management covering such aspects as concentration risk, alternative sourcing and supplier accreditation. It focuses on the threats to resilience highlighted by software and digital service supply chains and vendor management. These factors can be summarised as:

- Anatomy and volatility of the software supply chain
- Accountability spanning organisational silos
- Upgrades/changes and Software Bill of Materials (SBOM) management
- Supplier engagement during incident management and recovery.

Questions CEOs and Boards should ask

- Do we understand that supply chain software updates are a Board-level risk requiring contractual and governance controls?
- Do our third-party contracts include SBOM disclosure, staged rollout requirements, and rollback obligations and is every contract that lacks them escalated to us?
- Which single supplier's failure would take down the largest share of our Important Business Services and have we accepted that concentration knowingly?

- Can we evidence supplier assurance? For public sector Accounting Officers, supplier cyber resilience assurance is a personal accountability.
- Do we have the change governance in place to measure the daily risk of software supply chain dependencies, pipelines, and release changes?
- Is AI-generated code and autonomous AI agents, as a new supply chain channel, subject to the same inventory, validation, and rollback disciplines as vendor software?

3.1.2 Topics in this chapter

This chapter walks the full lifecycle of a vendor relationship through selection, contracting, onboarding, in-service management, incident response, and review or exit, and then connects that lifecycle to the artefacts that make it governable: the Software Bill of Materials (SBOM), the Service Level Agreement (SLA), the vendor register, the dependency map, and the contract schedule.

Topics are:

- Understanding the modern software supply chain
- Governance, accountability, and regulation
- The vendor management lifecycle
- SBOM and the extended supply chain artefacts
- AI in supply chain and vendor management
- What the Board should expect to see

Engineering Resilience for IT Systems:
Chapter 3 – Supply Chain and Vendor Management

3.2 Understanding the modern software supply chain

The software supply chain divides into two categories: the people who create the software (including vendors) and the automated systems used to develop it.[1] An end-to-end view connects the whole in a single flow: source code, open-source dependencies, developers, continuous integration and continuous delivery (CI/CD) pipelines, build systems, artifact registries, cloud and runtime platforms, and third-party services. It includes proprietary code written by the vendor, the vendor's use of open-source libraries, CI/CD tooling, build systems, cloud services, APIs, and third-party vendors.

Unlike traditional supply chains, the software supply chain changes every day. New dependencies are added daily, pipelines are updated continuously through automation, and applications are released multiple times a day[1]. Systems now include more lines of code, of variable quality, from many sources — Commercial Off The Shelf software (COTS), Software as a Service (SaaS), open source, and legacy software. Natural Accident Theory[2] predicts that in complex, tightly coupled systems of this kind, the “normal accident” is inevitable. The supply chain is where much of that coupling now lives.

Two characteristics deserve particular attention:

- Evergreen services: SaaS providers undertake to patch and maintain their services continuously. The benefit is currency; the cost is that the service may be enhanced to use further services you are not aware of, and where it is hosted and how it is constructed is not visible to you.
- 24/7 dependency: services operating around the clock need a supply chain for complementary and linked services that can also operate 24/7. Lengthy, complex supply chains, even when supported by contractual SLAs, may fail to provide needed support[3].

Managing the complexity of hybrid multi-cloud services across multiple tools, systems, and processes was the primary challenge cited by IT leaders in a recent industry survey, raised by 60% of respondents[4]. Complexity that cannot be seen cannot be governed; Section 3.5 addresses the tools that give visibility of the supply chain.

---

[1] https://cycode.com/blog/software-supply-chain/

[2] https://www.sciencedirect.com/science/article/pii/S0265964624000444

[3] https://londonpublishingpartnership.co.uk/books/resilience-of-services-reducing-the-impact-of-it-failures: Chapter 4

[4] Results of Kyndryl survey quoted with permission.

Engineering Resilience for IT Systems:
Chapter 3 – Supply Chain and Vendor Management

3.3 Governance, accountability, and regulation

Responsibility for implementing resilience lies across the organisation; business operations, internal IT, and third-party suppliers. Therefore, the RACI discipline described in Chapter 2 must extend across organisational boundaries, with suppliers appearing as named parties in the resilience RACI matrix, not as a footnote.

The management task has shifted accordingly. As the emphasis has moved from in-house development teams to external suppliers, the IT leader's task is to manage across organisations and silos: managing the people and activities in the software supply chain, managing procurement and contracts, and maintaining external and industry networks to share information on suppliers and products as the chain grows more complex.

> Example: UK public sector — supplier assurance is personal accountability
>
> The UK Government Cyber Action Plan[5] makes the Accounting Officer, the Permanent Secretary or CEO, personally accountable for the cyber risk of their organisation. That accountability explicitly "extends to appropriate assurance of the cyber security and resilience of their suppliers." Supplier assurance is not delegated away by outsourcing; it is retained at the top of the organisation.

Regulatory obligations. The 2026 regulatory environment places supply chain duties directly in scope. The table summarises the frameworks imposing supply chain and third-party obligations relevant to this chapter. (see Chapter 5.1.1 for the full-service delivery view).

For financial services firms, DORA's Register of Information converts vendor management from an internal practice into a supervisory submission: a complete, structured, continuously maintained inventory of all ICT third-party service arrangements. Non-financial UK organisations adopting these controls position themselves favourably under the UK CSRB and the FCA/PRA Unified Framework, which have equivalent expectations.

5 https://www.gov.uk/government/publications/government-cyber-security-strategy-2022-to-2030/government-cyber-security-strategy-2022-to-2030-html

Framework	Scope	Key supply chain obligation
UK CSRB (2026) [6]	UK CNI, Managed Service Providers, data centres	Security and resilience duties; supply chain risk management; NCSC CAF alignment; 24h/72h incident reporting
EU NIS2 (Oct 2024) [7]	EU 18 critical sectors	Risk management; executive accountability; supply chain security
EU DORA (Jan 2025) [8]	EU financial sector and critical ICT providers	ICT third-party risk management; Register of Information for all ICT third-party arrangements; Article 25 change management
FCA/PRA SS1/21 and Unified Framework (2026) [9]	UK financial services	Impact Tolerances for IBS including third-party dependencies; tested recovery
ISO 22301[10]	Cross-sector	BCMS: continuity strategies and exercised plans covering supplier dependencies

[6] European Union (2022). Directive (EU) 2022/2555 — NIS2 Directive. Official Journal of the European Union, L 333/.

[7] European Union (2022). Regulation (EU) 2022/2554 — Digital Operational Resilience Act (DORA). Official Journal of the European Union, L 333/1.

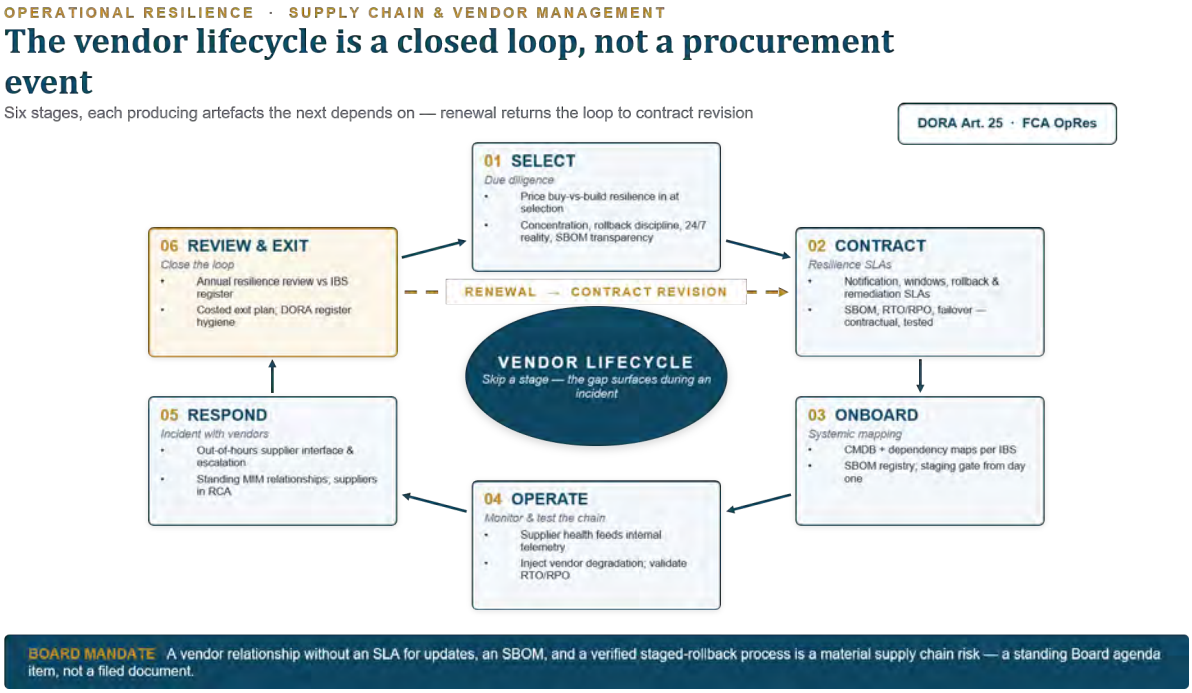
[8] Bank of England, Prudential Regulation Authority (2021). Supervisory Statement SS1/21 — Operational Resilience: Impact Tolerances for Important Business Services. London: Bank of England.

[9] Financial Conduct Authority and Prudential Regulation Authority (2026). Unified Cyber and Operational Resilience Framework. London: Bank of England.

[10] International Organization for Standardization (2019). ISO 22301:2019 — Security and Resilience: Business Continuity Management Systems. Geneva: ISO.

3.4 The vendor management lifecycle

Vendor management fails when it is treated as a procurement event rather than a lifecycle. Six stages structure the discipline: Select → Contract → Onboard → Operate → Respond → Review and Exit. Each stage produces artefacts that the next stage depends on. Skip a stage, and the gap surfaces later — usually during an incident.



Source: https://trustedtech-consulting.com/resilience.html

3.4.1 Selection and due diligence

The first resilience decision is whether to have a vendor at all. The buy-or-build trade-off (see also Chapter 4) is not only a cost and capability question: building retains control of the change pipeline; buying imports another organisation's release discipline, security posture, and failure modes into your critical path. Neither answer is wrong, but the resilience consequences must be priced in at selection, not discovered in service. Additional reliance features are considered in Chapter 4.4.3

Selection criteria must therefore extend beyond function and price:

- Concentration Risk: manage multi-region and provider diversity to mitigate concentration risk from single cloud provider failures. Ask at

selection: what proportion of our IBS estate would this supplier's failure take down?
- Deployment discipline: apply the Chapter 3.4.3 and 3.4.4 onboarding discipline. Human error and misconfiguration during software delivery remain primary vectors for outages.
- 24/7 support reality: validate that the supplier's support model genuinely operates at the hours your service does — contractual SLAs alone may fail to provide needed support.[11]
- Transparency: willingness to disclose an SBOM, hosting arrangements, and sub-supplier dependencies is itself a due diligence signal. Opacity at selection becomes blindness in service.

3.4.2 Contracting for resilience

The contract is the only control you hold over a supplier-driven update. Supplier-driven upgrades; security patches, content updates, evergreen enhancements — are applied remotely and are not under the organisation's direct control (Chapter 5),

Checklist for SLAs:

- Notification requirements for all update types including content and signature updates, not only version releases;
- Maintenance windows within which change may be introduced;
- Rollback and remediation obligations, with defined SLAs;
- SBOM disclosure as a contractual term in all software procurement;
- Recovery and RTO/RPO expectations, built into third-party SLAs at design time, before services enter production (Chapter 5);
- Failover and recovery-site arrangements e.g. GSLB or warm-standby regions, contractually available and tested, not assumed (Chapter 5);
- Source code escrow for critical supplier-owned software deposits covering code, build instructions, and dependency metadata, with defined release triggers (e.g. supplier insolvency or support withdrawal) and periodic deposit verification;
- Support availability, including out-of-hours escalation paths and named contacts.

Escalate the gaps. Where these controls are absent, the organisation's exposure is material and should be escalated to the Board as a supply chain risk. A supplier

[11] Excerpted from Chapter 4 of https://londonpublishingpartnership.co.uk/books/resilience-of-services-reducing-the-impact-of-it-failures/

contract without SBOM, staged deployment, and rollback obligations is an active Board agenda item, not a filed document.

Contract lifecycle management is an area well served by AI-assisted systems; the major professional services networks view AI as essential for automating contract lifecycles and legal knowledge management (Chapter 2)

3.4.3 Onboarding and integration

Advice: A vendor should be onboarded when its failure modes are mapped, not when its invoice is paid. Three integration disciplines apply before go-live:

- Dependency mapping: *extend the Configuration Management Database (CMDB) and dependency maps (see Chapter 4) to include the new service, its APIs, and its sub-dependencies, so the critical path per IBS remains accurate.*
- SBOM registration: *enter the supplier's components into the SBOM registry; flag open-source components against CVE databases automatically via CI/CD pipeline integration.*
- Staging gate: *establish the rule from day one, no third-party update enters production without validation in a mirrored environment. Define rollback procedures for all automated update mechanisms before, not after, first deployment.*

3.4.4 In-service management: updates, monitoring, and testing

Updates can cause failures; this is the documented primary cause of outages in production IT systems. A Treasury Select Committee report[12] found that in nine major UK banks, over two years, nearly half of all outages were due to the introduction of new software versions into existing systems. The CrowdStrike incident of July 2024[13] demonstrated the same vulnerability at global scale — delivered through the supply chain.

[12] . Treasury Select Committee (2019). IT Failures in the Financial Services Sector. London: House of Commons.

[13] CrowdStrike (2024). Falcon Content Update Incident: Technical Summary. Austin, TX: CrowdStrike

Example: CrowdStrike Falcon Update — July 2024 (the supplier-driven update)

What happened: a content update, deployed simultaneously and globally via the Falcon platform, crashed up to 8.5 million Windows endpoints across critical sectors, rendering devices unusable without manual intervention.

Vendor management failures: no staging gate for content updates; no automated rollback; customer organisations had no contractual control over the timing or gating of the update; no degraded-mode procedure (see Chapter 4) for systems dependent on the endpoint agent.

Lesson for this chapter: mandatory staging gates and rollback obligations apply to all update types, including content and signature updates — and they must be contractual, because the deployment decision sits with the supplier.

Example: Cloudflare Bot Management Outage — November 2025

What happened: a database permissions change caused a feature file to double in size, exceeding a hardcoded limit. Traffic routing crashed globally, affecting 2.4 billion users across ChatGPT, X, Spotify, and Shopify for six hours[13]. For every one of those dependent businesses, this was a supply chain incident they could neither prevent nor fix.

Resolution: Cloudflare's "Code Orange: Fail Small" programme[14] now stages and size-bounds configuration changes before global propagation.

Lesson: your vendor's configuration discipline is your availability. Ask for evidence of it at selection and review.

Advice:

Monitor the chain, not just the estate. *Monitoring must extend beyond owned components to supplier health. Commercial services such as DownDetector[15] provide monitoring of outages in the supply chain (Chapter 6); vendor status pages and API health endpoints should feed the same dashboards as internal telemetry, so that a supplier degradation is detected as a service event, not discovered from customer complaints.*

[14] Cloudflare Outage on November 18, 2025. Cloudflare Blog, 21 November.

Test the dependency. *Given the prevalence of SaaS and third-party integrations; payment providers, identity services, CDN providers, KYC vendors, resilience testing must include vendor degradation scenarios (Chapter 5.3.7):*

- *Implement synthetic transaction testing against vendor APIs;*
- *Execute contract tests to validate integration assumptions;*
- *Inject vendor slowness or outage conditions to validate timeout and circuit-breaker logic.*
- *Validate vendor performance against contracted non-functional requirements (NFRs) - latency, throughput, and capacity under load;*

Testing can confirm that posted RTO/RPOs remain valid if the vendor is unavailable, then results feed Service Level Management and the next SLA negotiation. Annual recovery testing is a regulatory requirement (DORA Article 24(6), ISO 22301 Clause 8.5, FCA Operational Resilience)[15,16]; the vendor dimension of that testing extends through to partner network services.

3.4.5 Incident response with vendor

When the incident involves a supplier, the supplier relationship becomes an operational capability. Chapter 6 indicates three vendor requirements that are part of the vendor management:

- Availability of the interface role: is the person who interacts with third parties, the supplier manager, available outside of working hours? (Chapter 6). If the answer is no, the recovery clock runs on the supplier's schedule, not yours.
- Standing relationships: the Major Incident Management team maintains close working relationships with suppliers alongside technical recovery managers, customer service, and architects (Chapter 6), established before the incident, exercised in simulations, not improvised at 3am.
- Suppliers in the diagnosis: root cause analysis explicitly includes suppliers in the 4 Ss of fishbone analysis (Surroundings, Suppliers, Systems, Skills) make the supply chain a first-class candidate cause, not an afterthought (Chapter 6).

[15]European Union (2022). Regulation (EU) 2022/2554 — Digital Operational Resilience Act (DORA). Official Journal of the European Union, L 333/1.

[16] International Organization for Standardization (2015). ISO/IEC 19770-2:2015 — IT Asset Management: Software Identification Tag. Geneva: ISO.

Engineering Resilience for IT Systems:
Chapter 3 – Supply Chain and Vendor Management

Maintain an up-to-date supplier contact schedule including out-of-hours support escalation paths; this is a deliverable of this chapter.

A vendor relationship that cannot be exited is a hostage situation with an SLA.

Checklist: The final lifecycle stage that closes the loop

- Annual resilience review: conduct annual resilience reviews of critical third-party dependencies aligned to the organisation's IBS register; evidence of staging discipline, incident history, SBOM currency, and support performance.
- Concentration re-assessment: re-test the concentration risk position at each renewal; dependency accumulates silently as usage grows.
- Exit and substitution plan: for each critical supplier, document the substitution path, data egress method, and realistic transition timeline. An exit plan that has never been costed is an assumption, not a plan.
- Register hygiene: update the vendor register and (for financial services) the DORA Register of Information on every material change/. It must be maintained continuously for supervisory submission, not rebuilt annually.

3.5 SBOM and the extended supply chain artefacts

3.5.1 Why the SBOM is needed

You cannot govern components you have not inventoried. The SBOM, as defined in ISO/IEC 19770-2[17], is a real-time inventory of every software component in the production system, with version, provenance, and known vulnerability status. In the modern supply chain, APIs, open-source libraries, CI/CD pipeline components, third-party SaaS dependencies can change dynamically on a daily basis; Each change is a potential injection point for failure, and the SBOM is what makes that change visible (Chapter 5.2.2).

For regulated financial services firms, DORA Article 25 mandates ICT change management controls and changes to production systems must be tested, validated, and subject to rollback procedures, FCA/PRA SS1/21[18] applies equivalent expectations to the broader UK-regulated population. The SBOM is the evidentiary artefact that demonstrates compliance. It also anchors the legacy improvement baseline: a complete configuration along with the SBOM is the starting point for any legacy gap analysis (Chapter 5.2.2).

Checklist: SBOM Audit (per ISO/IEC 19770-2)

- Maintain a current SBOM for all production systems, updated on each deployment
- Flag open-source components against CVE databases automatically via CI/CD pipeline integration
- Require vendor SBOM disclosure as a contractual term in all software procurement
- Monitor all SBOM updates; risk-assess proposed changes
- Establish a staging gate: no third-party update enters production without validation in a mirrored environment
- Define rollback procedures for all automated update mechanisms
- Add “AI-generated or AI-assisted code” as a line item in the SBOM audit

[17] International Organization for Standardization (2015). ISO/IEC 19770-2:2015 — IT Asset Management: Software Identification Tag. Geneva: ISO.

[18] Bank of England, Prudential Regulation Authority (2021). Supervisory Statement SS1/21 — Operational Resilience: Impact Tolerances for Important Business Services. London: Bank of England

- Review SBOM completeness quarterly as part of the resilience governance cycle.

3.5.2 The new supply chain channel: AI-generated code and agentic processes.

The SBOM and deployment governance disciplines were designed for a supply chain in which the organisation is a consumer of known software components.

However, when a business user builds an application with an AI coding tool, the resulting code contains dependencies, libraries, and architectural patterns selected by the AI model, not the organisation's architecture team. It enters the operational environment without passing the SBOM pipeline or staged deployment gates. SAP/Oxford Economics research[19] found 68% of UK organisations report employees using unapproved AI tools; Microsoft UK[20] found 62% of UK businesses have deployed AI agents without full governance visibility; IBM[21] traced 20% of breaches directly to shadow AI.

Two consequences follow. First, code that bypasses the SBOM pipeline also bypasses IBS classification. Unassessed is not the same as non-critical — and mapping obligations (FCA/PRA SS1/21; DORA Art. 8) apply whether the asset is registered or not. Second, the EU AI Act does not ask whether a system is an IBS. It asks whether the system is high-risk. A shadow-built application that meets that test attracts obligations in full — and can make the organisation a provider, not merely a deployer.

Advice: The rule is simple: AI-generated code is not exempt from supply chain governance; it is a new supply chain channel. All vibe-coded or AI-assisted applications should be subject to the same SBOM inventory, staged deployment, validation, and rollback requirements as vendor software; treat every AI-generated dependency as an unverified third-party component until validated.

Autonomous AI agents are architecturally equivalent to a new service and a new supplier in the infrastructure. Agentic drift, where behaviour changes because the provider updates the underlying foundation model without the organisation's knowledge or consent, is a supplier-driven update with no staging gate: precisely the risk category mentioned above in 3.4.1. Regulatory exposure is already live:

[19] . SAP and Oxford Economics (2026). AI in the Enterprise: UK Adoption and Governance Survey. February 2026

[20] Microsoft UK (2026). UK Cyber Pulse Report: AI Agent Deployment. March 2026.

[21] IBM Security (2025). Cost of a Data Breach Report 2025. Armonk, NY: IBM Corporation.

under the EU AI Act[22], anyone who places an AI system on the market under their own name, including a business user wrapping an API in a custom User Interface is classified as a provider, with full conformity obligations; the ICO's January 2026 report on agentic AI[23] confirms active UK monitoring.

Advice: an AI Asset Register could be set up alongside the SBOM — a living inventory of all AI-generated code, AI-assisted applications, and agentic processes operating within the organisation, classified by risk tier, with named ownership and regulatory applicability assessment.

[22] European Union (2024). Regulation (EU) 2024/1689 — Artificial Intelligence Act. Official Journal of the European Union, L 1689

[23] Information Commissioner's Office (2026). Agentic AI: Data Protection and Privacy Risks. London: ICO, January 2026.

3.6 AI in supply chain and vendor management

AI is both an opportunity and source of risk simultaneously. This section lists the areas of supply chain management that are using AI, not specifically for the software supply chain.

Checklist: applications of AI in supply chain and vendor management

Contract lifecycle automation: AI-assisted systems automate contract lifecycles and legal knowledge management; extraction of SLA obligations, renewal alerting, and clause comparison across the vendor portfolio. Route AI-flagged contract actions through human sign-off; the register of AI use cases carries named human accountability (Chapter 2).

Continuous vendor risk sensing: predictive analytics over audit, reporting and logging data, incidents, vendors, and external signals all act as an early-warning system, surfacing patterns that historically precede material disruption, feeding resilience planning with dynamic risk views so scenarios and playbooks update as the supply chain evolves (applied to software risk in Chapter 4).

Scenario planning for combined failures: AI-mined disruption scenarios help quantify likelihood and impact for the combinations that actually matter, for example cyber-attack combined with vendor outage and surge demand rather than a few manually imagined cases (applied to software risk in Chapter 4).

Advice: any AI system that calls external services or acts autonomously has crossed the internal-vs-integrated boundary and becomes part of the supply chain itself; inventory it, gate it, and give it a non-agentic fallback path.

3.7 What the Board should expect to see

The deliverables below would support Reports to the Board to answer the questions.

Do we understand that supply chain software updates are a Board-level risk requiring contractual and governance controls?

- Vendor and Third-Party Register: a complete inventory of third-party service arrangements mapped to the IBS they support; for financial services, maintained as the DORA Register of Information for continuous supervisory submission

Do our third-party contracts include SBOM disclosure, staged rollout requirements, and rollback obligations and is every contract that lacks them escalated to us?

- SBOM Registry: current software bill of materials for all production systems — including AI-generated and AI-assisted code, updated on each deployment and reviewed quarterly.
- Mandate SBOM disclosure and staged rollout for all third-party software entering production; no third-party update enters production without validation in a mirrored environment.

Which single supplier's failure would take down the largest share of our Important Business Services and have we accepted that concentration knowingly?

- Concentration Risk Assessment: documented provider-diversity position per IBS, reviewed at each renewal and after material architecture change.
- Extend the CMDB and dependency maps to cover vendor services, their APIs, and sub-dependencies on the critical path of each IBS.

Can we evidence supplier assurance?

- Supplier Contract Resilience Schedule: per-contract record of notification requirements, maintenance windows, staged rollout, rollback and remediation SLAs, and support availability, with gaps escalated to the Board
- Third-Party Dependency Test Results: synthetic transaction, contract test, and injected vendor-outage results per IBS, feeding Service Level Management and SLA negotiation.

- Supplier Contact and Escalation Schedule: named contacts and out-of-hours escalation paths for every critical supplier, validated in incident simulations.
- Exit and Substitution Plans: documented substitution path, data egress method, and costed transition timeline for each critical supplier.
- Ensure supplier health monitoring feeds the same dashboards as internal telemetry, so vendor degradation is detected as a service event.

Do we have the change governance in place to measure the daily risk of software supply chain dependencies, pipelines, and release changes?

- Align third-party change management with DORA Article 25 and FCA/PRA SS1/21 obligations.

Is AI-generated code and autonomous AI agents as a new supply chain channel subject to the same inventory, validation, and rollback disciplines as vendor software?

- Treat AI-generated code and agentic processes as a supply chain channel: inventory in the SBOM/AI Asset Register, gate, validate, and define non-agentic fallbacks.
- AI Asset Register: living inventory of AI-generated code, AI-assisted applications, and agentic processes, classified by risk tier with named ownership, maintained alongside the SBOM.

Engineering Resilience for IT Systems:
Chapter 4 - Planning for Improved Availability

4.1. Purpose and scope

This chapter sets out how to prioritise investment by business impact, how to build and maintain the architectural views that make failure visible, and how to select the patterns that contain damage and speed recovery.

- Failure is a planning assumption. Architect services so they fail in contained, recoverable ways and can be restored within agreed objectives.
- Prioritise by business impact, not technical complexity. IBSs and their impact tolerances determine where resilience investment is justified; not every service needs the same availability.
- You cannot recover what you have not mapped. Accurate, current configuration, dependency, and architectural views are the foundation of both anticipation and recovery; treat them as living artefacts under change control.
- Make objectives measurable, then prove them. RTO, RPO, and SLO targets, with error budgets, replace aspiration with evidence; a recovery capability that has not been validated through testing is an assumption, not a capability.
- Plan for the estate you have, not the one you wish you had. Inherited and legacy systems are the harder problem; apply lifecycle strategies, compensating controls, and explicit retirement decision points.
- Treat AI as both a planning tool and a planned component. AI strengthens scenario design and early warning, but AI embedded within services carries its own failure modes and needs the same governance as any other software.

4.1.1 Key messages

Planning for resilience is the responsibility of programme managers and enterprise architects, delegated from the CTO family, working with

- strategy, customer services, risk management and marketing to assess priorities
- Site Reliability Engineers and Business Continuity Resilience to assess and implement improvements.

4.1.2 Questions the CEOs and NEDs will ask

Boards and senior management expect answers from the enterprise architects and those managing systems to questions including:

- Are we confident that, when (not if) an incident occurs, we can restore IBS and/or critical services within agreed tolerances and continue operating?

- Do we have a map showing IBS and/or critical services and their supporting systems and infrastructure?
- Which IBS or critical services are at risk of not meeting Impact Tolerances?
- What more can we do to compartmentalise in case of cyber-attack?
- How are we using AI to improve availability?

Adopting standards in defining processes for managing risk creates consistency and improves understanding – relevant standards are described in the Annex: Standards. In this chapter we refer extensively to

- ISO/IEC 27001 on Information Security Management (ISMS)[1]
- NIST CF 2.0 Cyber Security Framework[2]
- NIST 800-53 Security and Privacy Controls for Information Systems and Organizations[3].

Additionally, the following standards are relevant to the resilience planning process:

- ISO/IEC 27031:2025 Cybersecurity — Information and communication technology readiness for business continuity[4]
- ISO/IEC 9837:2026 Systems and software engineering — Systems resilience concepts[5]

4.1.3 Topics in the chapter

This chapter covers planning for availability under the headings:

- The resilience planning toolkit
- How enterprise architecture can contribute to resilience
- Planning for graceful degradation
- Planning for recovery
- Application of AI to planning for availability
- Governance
- Key deliverables

[1] ISO/IEC 27001 on Information Security Management is detailed in Annex: Standards, https://www.iso.org/standard/27001

[2] NIST CF 2.0 Cyber Security Framework - alignment of resilience governance is detailed in Annex: Standards

[3] NIST 800-53 Security and Privacy Controls for Information Systems and Organizations, https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final

[4] https://www.iso.org/standard/27031

[5] https://www.iso.org/standard/83604.html

4.2 The resilience planning toolkit

4.2.1 Measuring costs and benefits

The Enterprise Architecture function (EA) is responsible for providing a clear view of the total IT estate, including resilience measures and where resilience falls short. This chapter describes the tool kit that EA uses to do this. Programme Managers rely on the information on each of these to manage a portfolio of services.

Chapter 1 details how IBSs are defined. If they have not been formally defined for the organisation, start with the most visible business services and apply the same prioritisation logic. Chapter 2.4.1 discusses the assessment of risk and costing of outages.

The following questions should drive prioritisation:

- What is the business priority of this service?
- What is the MTTR requirement?
- If a system has data loss, is it an acceptable level (within RPO) or do we need to recover the data?
- What is the Impact Tolerance for the service?

The answers to these questions expose the services which need to have higher levels of resilience and expose where the current estate falls short in terms of current resilience.

A Statement of Applicability (SoA) is used in frameworks like ISO 27001 and ISO 42001. It documents decisions, including control selection and rationale. It acts as a bridge between your risk assessment and your chosen maintenance and prevention methods. The SoA document will generally include:

- standard controls,
- whether or not they apply and their justifications,
- their implementation status,
- related documentation (procedures, evidence, etc.),
- all those additional data that may be considered necessary to record.

External and internal factors will also impact risk appetite and resilience planning. The external factors that are most commonly encountered include:

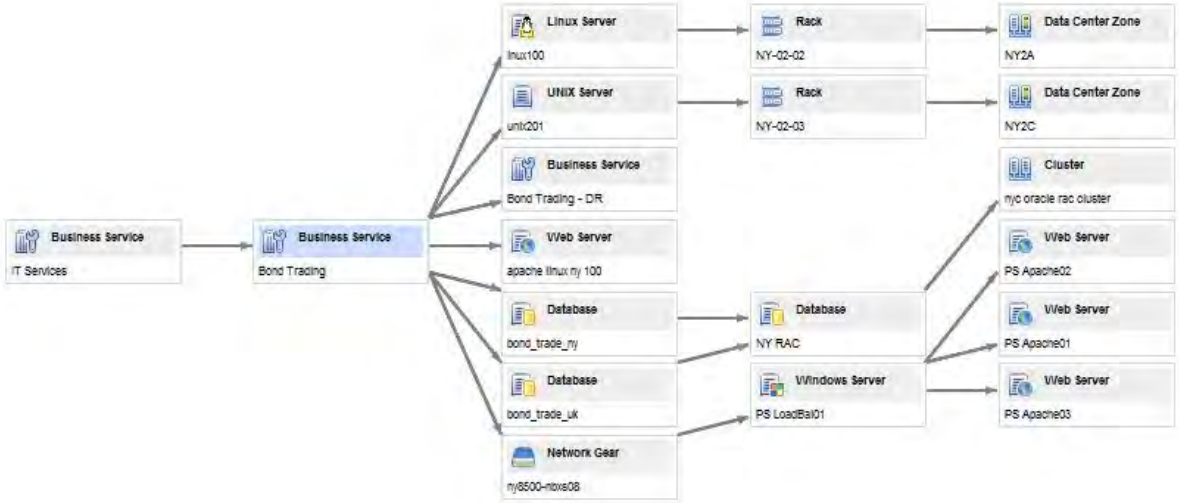
- Evolving Digital Regulation (DORA & UK Frameworks): Aligning technical anticipation with EU-harmonised ICT risk management (DORA) and UK-specific BoE/FCA operational resilience mandates.

- Cloud Security Governance: Managing multi-region diversity and provider diversity to mitigate "Concentration Risk" from single cloud provider failures. (Chapter 3.4.1)
- Cybersecurity resilience lens: Apply segmentation, identity and access management, encryption, secure configuration, and continuous monitoring to reduce blast radius and improve detection and recovery.

4.2.2. The system map

Mapping infrastructure and application dependencies across on-premises and cloud environments is the basis of knowing the risks to business services. System maps are the basis of maintenance and preventive action, and of effective incident response. They support risk assessment and help teams focus on the most material risks.

The following diagram illustrates both the configuration and dependencies form with a CMDB tool, in this case ServiceNow[6].



Source: https://servicenowguru.com/cmdb/walking-servicenowcom-cmdb-relationship-tree/

Application Dependency Mapping maps infrastructure and application elements and services across an organisation's environment. Given the volatility of many software configurations, the requirement is for a comprehensive, real-time, fast, and repeatable application dependency mapping tool that does not affect performance. Monitoring can be set across each of the inter-process communication layers for alerts if values go out of normal range. It is important that threshold levels are set so that spurious alerts do not cause "alert fatigue".

[6] https://www.servicenow.com/uk/

IT infrastructure mapping is the foundation of many operational processes, including performance monitoring, incident management, IT asset management, business continuity and disaster recovery, cybersecurity, and micro-segmentation. Where recovery requires rebuilding environments (for example, after a regional loss), inventories should extend beyond assets to include build standards, configuration baselines, and documented recovery site approaches (cold, warm, and hot standby)[7].

Legacy systems are often embedded in an IBS. They often accumulate add-ons and bolt-ons, so that suddenly they are doing much more than they used to. A reliable configuration map can help identify critical paths and potential failure points, and so, where monitoring and alerting should be prioritised. In order to improve a legacy system, baseline data includes:

- complete configuration along with SBOM (detailed in Chapter 3.5) known today,
- how the service behaves today

From there, a gap analysis can compare what exists today with the targeted level of availability and recoverability.

Advice
For legacy estates, adopt an explicit lifecycle strategy (including obsolescence management) that is risk-based. The strategy should define supportability criteria, upgrade paths, compensating controls, and decision points for retirement or replacement. We need to make difficult choices when our resilience objectives cannot be met cost-effectively and these choices may only be taken in relation to well-defined risk management.

4.2.3 Exposing vulnerabilities

A Failure Mode and Effects Analysis (FMEA) can be used to proactively identify and mitigate potential failures in a product, service, or process before they happen. It[8] is a structured reliability evaluation process that identifies failure points in advance by examining how components fail, potential failure modes, and their impacts at the system level, including interactions among subsystems and components.

[7] See ISO/IEC 19770-2 for maintaining relevant software component inventories.

[8] FMEA see https://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis

In a resilience context, FMEA should be used to assess which failure points are most critical, how likely they are to affect availability or integrity, and for how long or how often they might occur.

An FMEA process should be undertaken for a business service. This ensures that prioritisation is grounded in service impact rather than technical issues. Practical outputs include a maintained map of all Important Business Services and their dependencies, and a documented record of architectural deficiencies.

4.2.4 Mapping of services

A static systems map is a prerequisite to getting a view, which maps business services to Applications, data and Infrastructure. A tool to display this in real-time is maintained by SREs, and the data captured enables shared understanding of what is going on.

A comprehensive view maps the dependencies from the business process down to the physical hardware.

- Business Impact Layer: Map critical business processes (e.g., “Process Payroll”) to their required uptime. Define RTO (How fast?) and RPO (How much data loss?) for each.
- Application Dependency Map: A visual graph showing which apps depend on which APIs. This identifies "hidden" single points of failure.
- Infrastructure Dependency Map: A visual representation of all the infrastructure components and the availability features incorporated, covering servers, storage, networking.
- Data Sovereignty & Replication: Documentation of where data lives, how often it is backed up, the latency between the primary site and the recovery site and the operation of a kill switch.

Service maps, generated by tools such as Datadog[9], Dynatrace[10] or AWS Cloud Map[11], can show live traffic flow and reveal unexpected dependencies or choke points. Network topology diagrams highlight redundant and non-redundant paths, and the impact of load on both services and the network. Modern IT environments increasingly use cloud and virtualised architectures. These can improve scalability and recovery options, but they also require clear governance over configuration, identity, segmentation, and backup/restore so that recovery assumptions remain valid as the platform evolves.

[9] https://www.datadoghq.com/

[10] https://docs.dynatrace.com/docs/observe/application-observability/services/service-map

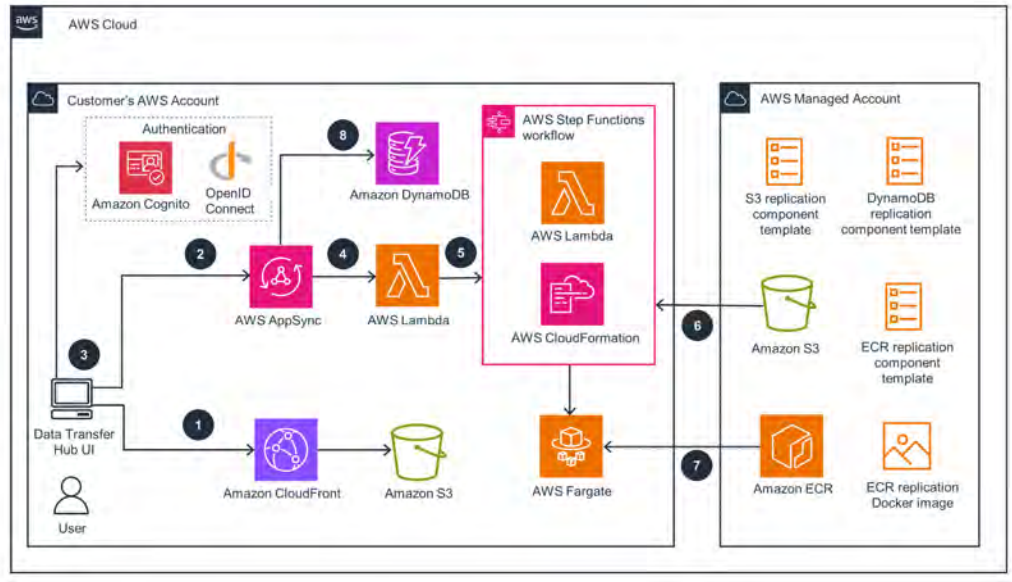
[11] https://aws.amazon.com/cloud-map/

Engineering Resilience for IT Systems:
Chapter 4 - Planning for Improved Availability

Observability - the landscape of system monitoring has shifted from simple “uptime” checks to full-stack observability. The objective is no longer just to know if a server is up, but to understand why a specific user request failed across a complex web of microservices, APIs and infrastructure—and what the impact of such a failure is. Discovery of hidden dependencies, unvalidated recovery paths, and unacknowledged Single Points of Failure (SPOFs) is critical as they often remain invisible until they cause incidents. The use of AI and other tools will help in the discovery.

More details are provided in Chapter 5.4.1

Here is an illustration of a Service Map.



4.2.5 Keeping the map current

One of the biggest risks to successfully managing resilience is that the architecture and the system changes, but the recovery plan stays the same. Avoiding that drift requires a living-document approach in which every significant project updates the resilience view before it goes live.

Deployment pipelines, the standard procedures employed in introducing new or updated system components, are also useful artefacts to support Infrastructure as Code (IaC) and enable environments to be rebuilt from scratch in a new environment. These artefacts are more than operational aids; they also act as living architectural evidence which can be reviewed, tested, and challenged as part of the approach to resilience.
In addition, automated discovery tools should be used periodically across both cloud and on-premise environments to align the documented architecture with what is actually deployed. Differences should trigger a review of associated resilience plans, as this is often where resilience blind spots emerge.

As updates are applied, the architecture may change – affecting resilience. Detection of drift can be automated: for example, by alerting architects when a new resource appears without the disaster recovery tags or the replication controls expected by policy. More mature organisations may extend this approach through controlled chaos engineering to validate that systems really fail in the contained ways that the architecture assumes.

4.3 How Enterprise Architecture can support resilience

Checklist – supporting resilience

- An authoritative asset inventory, maintained by automated discovery where appropriate, so that critical components and their failure domains are visible and can be governed.
- Document critical data flows that support business continuity, including whether services can operate safely in a constrained or isolated mode during upstream failures. This clarifies recovery sequencing, integrity requirements, and operational workarounds during disruption.
- Ensure critical systems are hardened using secure configuration baselines and verified build standards. Where benchmarks (e.g., CIS[12]) are used, treat them as implementation guidance and ensure deviations are risk-assessed and approved, with evidence retained for assurance.
- Use ISO/IEC 27001 to define practices, ensure they are implemented and evidenced through risk-based control selection and operational control.

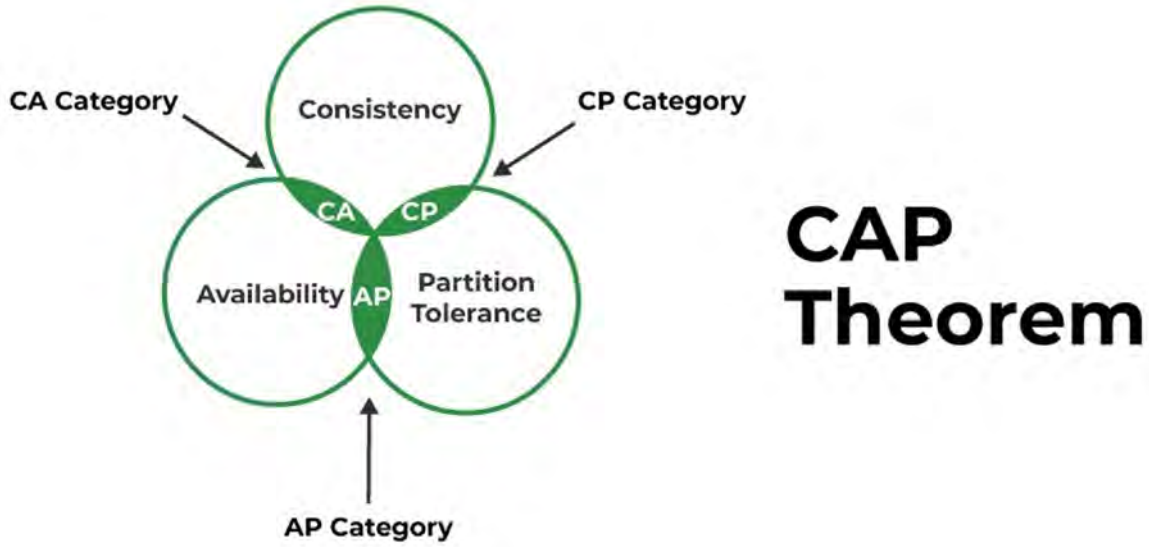
4.3.1 Resilience in distributed, networked systems

Modern IT systems are usually solutions assembled from many platforms, services, and SaaS providers. That increases flexibility and delivery speed, but it also multiplies inter-dependencies and makes resilience harder to achieve.

State management, and the recovery of state, needs to be considered as part of resilience. The CAP theorem[13] - Consistency (all nodes see the same data), Availability (every request receives a response), and Partition Tolerance (the system continues to operate despite network failures) is a useful example of the architectural thinking involved.

[12] https://www.cisecurity.org/cis-benchmarks

[13] https://www.geeksforgeeks.org/operating-systems/pacelc-theorem/



Any architecture that distributes state across nodes must deal with the trade-off expressed by the CAP theorem. During a network partition, a system must choose between consistency and availability. Because partitions are unavoidable in real distributed systems, the practical issue is not whether to tolerate partitions, but which of consistency or availability to sacrifice when one occurs.

Advice:
The right choice depends on context. If the cost of a wrong answer is high, such as in ledgers or regulated records, consistency should dominate. If no answer is worse than a slightly stale one, availability may be preferable. The feasibility of reconciliation after partition and user expectations around latency also matter.

An extension to the CAP theorem known as PACELC states that in a partitioned system (P), one has to choose between availability (A) and consistency (C), but else (E), even when the system is running normally in the absence of partitions, one has to choose between latency (L) and loss of consistency (C).

Advice:
Understanding a 3rd party's platform's CAP and PACELC posture (or appetite) is essential for setting realistic Service Level Objectives, designing recovery patterns, and explaining system behaviour during incidents.

4.3.2 Principles of enterprise architecture for resilience

Systems designed for resilience may implement one or more of these principles.

- Redundancy removes or reduces single points of failure by duplicating critical components across failure domains. If a database is a critical component, this requires an architecture that keeps data in sync, either synchronously or asynchronously. Critical components can exist in all layers of the architecture.
- Loose coupling prevents one system from depending on another's immediate success, often through asynchronous messaging. This may be through message queues, asynchronous streaming or transient data stores.
- Defence in depth layers security and resilience controls so that one failed barrier does not expose the core service, whether that barrier is a network access control layer, a firewall, or key management.
- Graceful degradation allows a system under stress to preserve core functions while disabling non-essential features. A good example would be that of an e-commerce site disabling 'recommended for you' suggestions when the site is handling a high volume of sales.
- Design for recovery shifts attention from Mean Time Between Failures to Mean Time to Repair, favouring approaches such as immutable infrastructure and automated rebuilds.

Whether a system is built or bought these patterns should apply to how the service is deployed and should be considered for any external service we are using (ref 3.4.1)

Advice:
Explore the Annex: Public Reports to understand how to ensure that the implementation of the principle has the desired effect.

4.3.2 System Compartmentalisation

System compartmentalisation is an important method for implementing the principles of redundancy and loose coupling. System compartmentalisation localises the impact of IT failures to prevent a single fault from cascading across the entire organisation.

In operational terms, compartmentalisation means that the failure of one service does not take down adjacent services. This requires explicit design: services fail independently, dependencies are documented and bounded, and circuit breaker need to be implemented to prevent a failing downstream service from blocking the entire request chain.

The Bulkhead Pattern, documented in Microsoft Azure Architecture Centre guidance[14], isolates service resource pools so that exhaustion in one partition

[14] Microsoft Azure Architecture Centre (2023). Bulkhead Pattern. learn.microsoft.com

cannot starve adjacent services. Netflix open-sourced Hystrix[15] following its 2012 outage, a circuit breaker library preventing cascading thread exhaustion.

Advice: These are strongly recommended architectural controls for any IBS in a distributed architecture. They represent recognised good practice under DORA Article 9's requirement to limit ICT incident propagation.

4.3.3 Feature Flag Hygiene and Kill Switches.

When a feature causes cascading failures, the fastest remediation is often a remote kill switch, disabling the feature without a code deployment. Engineering teams should conduct a Feature Flag Audit for every new feature, ask whether it can be toggled off without a release. Maintaining both existing and new codebases in parallel increases regression testing overhead before each release. That cost should be recognised and budgeted.

Case Study: Cloudflare Bot Management Outage — November 2025

Now looking through the lens of prevention (Case study from 3.4.4)

What happened: (see section 3.4.4)

Key failures: No staging gate for configuration propagation; hardcoded memory limit with no graceful fallback; circular service dependencies (Turnstile required for Dashboard login, but Turnstile was down, so engineers could not login); and critically, no remote kill switch, meaning the only recovery path was a full redeployment, extending the outage window beyond what a flag toggle would have required.

Lesson: Without a remote kill switch, remediation reverts to the slowest available recovery path. The Cloudflare incident illustrates the operational cost of this control's absence directly, a feature flag toggle would have isolated and disabled the offending component without a full redeployment. Every new feature should consider if the extent of its blast radius and if therefore it requires a toggle-off-available without a release.

4.3.4 Stateful Data Compartmentalisation

Shuffle sharding distributes workloads across thousands of tiny, isolated partitions, limiting the blast radius of a noisy neighbour to a negligible percentage of users. Engineering teams should evaluate shuffle sharding as a

[15] Netflix Technology Blog (2012). Fault Tolerance in a High Volume, Distributed System: Hystrix. Netflix Engineering.

structural control for any IBS where noisy-neighbour propagation presents an impact tolerance risk.

> Case Study: Netflix - Shuffle Sharding Adoption
>
> What happened: Netflix adopted shuffle sharding[5] to address the noisy-neighbour problem in its distributed infrastructure. Workloads were distributed across thousands of tiny, isolated partitions, limiting the blast radius of any single infrastructure failure to a negligible percentage of users.
>
> Key features: Shuffle sharding was implemented as a structural compartmentalisation control rather than a reactive fix. The architecture ensured that failures could not propagate across partition boundaries, containing impact before it reached scale.
>
> Lesson: Events which previously caused visible degradation to a material user population became operationally invisible to the majority of users. Shuffle sharding reframes infrastructure failure from a service availability event into a contained limited impact fault, which is precisely the outcome that compartmentalisation is designed to achieve. For any IBS where noisy-neighbour propagation presents an impact tolerance risk, this is an architectural control, not an optimisation.

4.3.5 Diverse Validation Logic

Redundancy eliminates single points of failure in infrastructure, but not in logic. When primary and secondary systems share the same validation algorithm, a common-mode defect defeats the redundancy entirely. Engineering teams should specify that primary and secondary IBS systems use independently developed validation logic. This pattern, known as N-Version Programming or Diverse Validation Logic, is an architectural design requirement, not a regulatory obligation. It belongs in the NFR specification for each IBS, not in the compliance register.

> Case Study: NATS Outage - August 2023
>
> What happened: Redundant systems processed the same malformed data through the same validation logic, producing the same erroneous output. The entire UK air traffic management network ceased operation.
>
> Key failure: The redundancy provided no resilience because the fault was in the shared logic, not the infrastructure. Identical logic across primary and secondary systems meant that they both processed the logic in the same manner and redundancy amplified rather than contained the failure.

Lesson: Redundant systems sharing a common validation algorithm are not redundant against logic faults. Primary and secondary IBS systems need to consider independently developed or cross-checked validation logic, N-Version Programming, or heartbeat check for the parallel process availability, to prevent common-mode failures from propagating through architecturally separate components.

4.3.6 Data Validation and Logic Guardrails

This case study of the NATS outage of August 2023 also demonstrated how bad data kills systems, a single logic error, two waypoints sharing the same name caused a lookup table to return an impossible result, triggering a critical exception that shut down the primary system, then the secondary, then the entire UK air traffic management network. The failure was not in the infrastructure. It was in the data validation logic.

Data validation is a resilience control, not a data quality measure. It belongs in the architecture and in the acceptance testing criteria, not as an afterthought. (See Chapter 2.5.3 for basic types of validation)

4.3.7 Data Validation and Logic Guardrails

Implementation of check digits, format checks, range checks, and lookup table validation should be incorporated as non-negotiable acceptance testing criteria for every IBS. The NATS incident demonstrates how a logic-based shutdown can propagate through redundant systems; validation is the first line of defence. With appropriate data validation, the system could have flagged the error and not processed this data and moved on to the next record to continue processing. Should there be several errors the application should stop processing after a pre-defined threshold has been reached.

Case Study: NATS Outage — August 2023 (Revisited)

This case study revisits the NATS incident through the lens of data validation controls, complementing the diverse validation logic analysis above.

What happened: Software created a flight plan including two waypoints with the same name, Deauville (France) and Devils Lake (North Dakota). A downstream system rejected the plan as the UK exit point was interpreted as the North Dakota location. The primary flight processing system flagged

a critical exception and shut down. The secondary system made the same error. The entire UK air traffic management network ceased operation.

Key failures: No lookup table deduplication; no cross-validation of waypoint coordinates against geographic plausibility; no degraded operating mode; staff not trained to manually enter flight plans; no single post-holder with overall incident management accountability.

Lesson: Input validation is not a back-office data quality concern. It is a front-line resilience control. Validation failures should be a monitored, alerted, and SLA-governed event category.

Checklist: Data Validation Rules

- Implement check digit validation on all critical reference codes and identifiers
- Enforce format checks on all structured data inputs (dates, postcodes, NI numbers, account codes)
- Apply length checks to prevent buffer overflow and truncation vulnerabilities
- Maintain and version-control all lookup tables; test updates before deployment
- Enforce presence checks on mandatory fields at point of entry
- Apply range checks to all numeric and date inputs; define acceptable bounds per field
- Include data validation failures as a monitored, alerted event category in the SIEM
- Determine how a system will deal with one or more data errors
- Introduce honey trap records - synthetic, fictitious entries in lookup tables and reference datasets that should never appear in live transactions; any hit triggers an immediate alert for audit and investigation.

4.3.8 Provision for Load Shedding

Provision for load shedding is a key design feature. Load shedding involves deliberately dropping non-essential background tasks or lower-priority requests during periods of saturation in order to protect critical path services. The principle is part of graceful degradation in architecture mentioned in previous subsection (4.3.2): preserve the most important function at the cost of less important ones. Automated load shedding may be built into systems but its

operation needs to be monitored and further provision for the SRE to raise the level of shedding in response to operating condition according to a pre-agreed plan needs to be part of the architecture.

For any regulated sector operating under DORA, FCA/PRA, or ISO 22301 certification the critical governance requirement is that load shedding decisions must be pre-defined, not improvised. An operator under pressure at 3 AM should not be deciding which workloads to drop, that decision should already have been made, documented, and assigned. This is discussed in Section 4.5 on Degraded Mode Operating Procedures (DMOP).

4.4 Planning for anticipation and graceful degradation of live systems

4.4.1 Use of monitoring data

The definition, capturing and analysis of monitoring data depends on cooperation between the planners, operations, and recovery teams.

Service Level objectives (SLO's) provide metrics to align your operational uptime and security metrics with mandatory EU cybersecurity and critical infrastructure regulations using the NIS Framework. SLO's are often expressed as error budgets for operational use.[16]

Defining SLOs

An error budget expresses a SLO in terms of how many minutes per time period the service may be down for the agreed availability %. The Error budget creates a measure in terms of a failure or breakage count. An example of the use of an Error budget is given in 5.4.2.

4.4.2 Assessing services for deployment or supplier selection

When assessing developing or acquiring services for deployment the following additional resilience features are worth considering. These are additional to the basic criteria discussed in Chapter 3.4.1.

[16] See https://oneuptime.com/blog/post/2025-09-03-what-are-error-budgets/view

Checklist: for supplier selection

Suppliers should be able to provide high resilience features, such as:

Quorum systems and modular redundancy
As systems scale across multiple nodes and availability zones, network partitions can cause nodes to lose contact with each other. This can lead to "split-brain" scenarios where different parts of a system attempt to process conflicting data.

To prevent this, critical state and business logic should utilise Quorum Systems or Triple Modular Redundancy (TMR). In a quorum-based architecture, a cluster of components must vote and reach a majority agreement before a transaction is committed or a new cluster leader is elected. If an enterprise operates a cluster of three nodes, at least two must agree to establish the truth.

Asynchronous decoupling and dead-letter queues (DLQs)
Tightly coupled synchronous APIs are fragile; if downstream System B fails, upstream System A is blocked. Resilient architectures utilise message brokers to decouple system components.

By introducing a buffer, System A can instantly return a 200 OK success signal to the client as soon as a request enters the queue, abstracting the processing latency. System B can then consume messages at its own pace, providing critical "load levelling" that prevents backend databases from being overwhelmed during sudden traffic spikes.

Event sourcing and append-only journaling
Traditional relational databases manage state via in-place updates, continuously overwriting historical data. If a database record becomes corrupted, determining exactly when and how the corruption occurred as rolling it back is a highly complex and potentially destructive process.

Event Sourcing inverts this paradigm. Instead of storing the current state, the system writes every state change as an immutable domain event to an append-only journal (the Event Store). Because this journal is strictly append-only, it eliminates the performance overhead and locking contentions of in-place updates, allowing systems to handle massive write throughput during traffic spikes.

4.5. Planning for recovery

Architecting for recovery is the practical subset of resilience concerned with returning to a known-good state after total or partial failure. Resilience focuses on staying up under pressure, whereas recovery focuses on how quickly and safely the service returns when pressure drives a failure. The role of planners is to create the capability for recovery. In Chapter 5 we describe the role of the recovery team.

Advice:
The mindset is to expect failure as a statistical certainty and anticipate the means that will be used to handle it gracefully. Recovery architecture should therefore be designed deliberately rather than improvised after a major incident.

- *Design redundancy to meet defined availability and recovery objectives (for example, clustered services and multi-zone deployment).*
- *Maintain a risk-based vulnerability and patch management process that includes testing and verification in representative environments before production rollout, with clear rollback procedures and evidence of execution.*

4.5.1 Knowing the recovery objectives

Recovery architecture must begin with clear objectives. RTO defines the maximum tolerable duration of downtime, while RPO defines the maximum tolerable amount of data loss measured in time.

These measures should be defined for each IBS or critical service and then used to choose an economically realistic recovery pattern. Without that, recovery architecture easily drifts into either over-engineering or false confidence.

4.5.2 Basic recovery-oriented patterns

Resilient architectures prevent a single failure from cascading into a total system collapse. The approaches to decoupling systems were described above - the techniques that can be applied to existing systems are Bulkheading, Circuit Breakers and Sidecars.

4.5.3 Architectural approaches that support data recovery

Resilience is often won or lost at the data layer, and this is particularly true in terms of recovery. Approaches which could be considered.

- Asynchronous Replication: Maintain a "warm" standby database in a different geographic region.
- Event Sourcing: Instead of just storing the current state, store the sequence of events. If the database is corrupted, you can "replay" the events to reconstruct the state perfectly.
- Immutable Backups: Use "Write Once, Read Many" (WORM) storage to protect against ransomware; even if an attacker gains admin access, they cannot delete or encrypt your backups.

4.5.4 Recovery approaches at the deployment and infrastructure layer

The architecture of the infrastructure must be as dynamic as the software it supports. Recovery depends as much on infrastructure discipline as on application design. There are a number of approaches which help with recovery. They include:

- Statelessness: Design application nodes so they can be replaced automatically. If a node fails, the platform should terminate and recreate it without loss of user session data, relying on externalised state where appropriate.
- Infrastructure as Code (IaC): Treat environments as reproducible code artefacts. In a provider or regional outage, the organisation should be able to redeploy the required stack into an alternative region using controlled, tested automation.
- Chaos engineering: Use controlled fault-injection exercises to validate that systems fail in the contained ways the architecture assumes and that automated recovery mechanisms work as designed. See 5.3.3

4.5.5 Deployment patterns for recovery

The appropriate recovery pattern depends on the RTO, RPO, and budget available . Backup and restore is the lowest-cost option, with infrastructure rebuilt and data restored after failure, but its RTO is typically measured in hours or days. Other modes of working after service outage include:

- Pilot light keeps a minimal version of the environment, often the database layer, always running and synchronised so that application capacity can be scaled up during a disaster.
- Warm standby keeps a scaled-down but functioning environment online, reducing RTO further because traffic only needs to be redirected.
- Multi-site active-active splits traffic across two or more fully functional regions so that recovery is effectively immediate, though at very high cost.

4.5.6 Automated Recovery (Failover) techniques

Here are examples of other techniques that are both simple to add and are widely in use.

- Automated failover (DNS & load balancing)
  Recovery should not depend on ad-hoc manual intervention. Use health checks linked to global server load balancing (GSLB). If the primary site fails to respond, traffic is automatically re-routed to the recovery site in line with defined failover criteria. See Chapter 5.3.7 and 5.4.1.
- Database Clustering
  To maintain high availability for the database it is deployed over a server and storage farm and manages use and consistency over all nodes. Oracle RAC is a “share everywhere” database whilst Microsoft SQLServer operates in an Active-Passive mode.

4.5.7 Data integrity & point-in-time recovery (PITR)

In a ransomware or corruption event, a "live" backup is useless because it will just be the same as the corrupted data. Architect for PITR, allowing you to "roll back the clock" to a specific second before the corruption occurred. Two techniques are:

- Worm (Write Once Read Many)
  A data storage principle where information, once recorded, cannot be altered or deleted. It ensures data immutability and tamper-proofing.
- Infrastructure as code (IaC)
  To recover an environment, you must be able to recreate it exactly. Store your networking, security groups, and server configurations in code (e.g., Terraform[17] or CloudFormation). This ensures the recovery site is an identical twin of the production site, avoiding "configuration drift" errors.

[17] Terraform see https://en.wikipedia.org/wiki/Terraform_(software)

4.6 Use of AI in planning for availability

AI presents several opportunities for availability improvement as outlined in this section. For further details see Annex - AI Support for Enterprise Architecture.

4.6.1 Inputs to architecture design

Stronger scenario planning

- AI mines internal and external data (incidents, threats, providers, environment) to propose "extreme but plausible" disruption scenarios instead of a few manually imagined ones.
- It helps quantify each scenario's likelihood and impact, so resilience planning can prioritise the combinations that actually matter (e.g. cyber + vendor outage + surge demand) see Chapter 3.6.

Deep simulation of complex behaviour

- ML-enhanced system-dynamics or agent-based models let you simulate how a complex sociotechnical system will behave under stress, including cascades and feedback loops you would not see analytically.
- Synthetic data and "digital twins" of operations allow safe stress tests of rare events, so you can test options (buffering, routing, throttling, failover) before touching production.

4.6.2 Using operational data

Continuous risk sensing and early warning

- Predictive analytics over telemetry, incidents, vendors, and external signals can act as an early-warning system, surfacing patterns that historically precede material disruption. This must include any suppliers, see Chapter 3.6.
- These models feed resilience planning with dynamic risk views, so your scenarios and playbooks update as the system and its environment evolve.

4.6.3 Prioritising resilience investments

These allow us to determine where to locate resilience investments

- AI can use operational data to estimate how changes (extra redundancy, refactoring a hub system, new playbook) would shift key metrics like time to detect/respond/recover, helping you compare options and pick the highest leverage interventions.

- It supports portfolio level planning by clustering systems and processes by criticality and fragility, guiding which parts of the architecture to harden first.

These allow us to monitor the benefits

- AI tools can track remediation actions from exercises and reviews, send reminders, and maintain a live view of completion status against resilience roadmaps.
- They generate resilience “scorecards” and trend metrics from repeated simulations and real incidents, evidencing to boards and regulators how the complex system’s resilience is maturing over time.

4.6.4 Documenting and maintaining systems maps

- For legacy systems, AI can provide a map that augments current inhouse understanding of dependencies
- For ongoing risk management, AI can track changes to the system and affected services.

4.7. Governance

4.7.1 Cross functional working

Planning for availability is the responsibility of the CTO family, in consultation with business services. Programme management and enterprise architects make proposals for implementation methods – technical or operational.

4.7.2 Failure modes

The EA team works with service delivery teams to anticipate failure modes, enable the definition of monitoring and detection and validate recovery approaches. This ensures objectives and priorities are aligned across technology, operations, and the business.

Documented evidence of failure modes must be available to the CIO for evidence to the Board. Assurance should include demonstrable detection and response capability: monitoring coverage for critical paths, defined alert thresholds, and tested incident response procedures. This is defined in ISO/IEC 27001 on operational planning and performance evaluation, and should be supported by logs, alert signals, and post-incident reviews.

4.7.3 Reporting and compliance

Under the NIS Framework, compliance is expressed in terms of

- lost user hours
- loss of data integrity
- loss or damage to health or life
- financial loss to a major user.

The CTO family will base reporting on the SLO data for incident management, and from Business Continuity for major incidents – see Chapter 6.

4.8 What the Board should expect to see

At the beginning of this chapter, we identified some questions that the CEO or NED's might ask. Here are the deliverables which provide the answers.

Are we confident that, when (not if) an incident occurs, we can restore IBS and/or critical services within agreed tolerances and continue operating?

- ISMS Artefacts (ISO/IEC 27001): scope, risk assessment method, risk register, Statement of Applicability, and audit and management-review actions.
- Accurate Configuration Baseline: current CMDB and/or automated discovery: inventory, dependency relationships, and drift reports, validated continuously.

- Validated Recovery Architecture: RTO/RPO-matched recovery patterns, IaC templates, PITR and immutable-backup design, proven through game

- SRE-to-Stakeholder Contract: agreed SLOs/SLIs, RTO/RPO mapping, and escalation paths linking telemetry to IBS Impact Tolerances.

Do we have a map showing IBS and/or critical services and their supporting systems and infrastructure?

- Infrastructure & ADM Visualisation: logical and physical topology including microservices, with Critical Path per IBS, a SPOF register, and a recovery sequencing map.

Which IBS or critical services are at risk of not meeting Impact Tolerances?

- SLO & Error-Budget Framework: per-IBS SLOs, multi-burn-rate alerting, and error budgets linked to Impact Tolerances.

What more can we do to compartmentalise in case of cyber-attack?

- FMEA & Deficiency Log: service-focused failure analysis, an architectural decision record, and an accepted-risk register.

How are we using AI to improve availability?

- There are no standard approaches to reporting on this.

Engineering resilience for IT systems: Chapter 5 – Service Delivery

5.1 Purpose of this Chapter

This chapter is about what happens when the system is live and in service. Its mandate is precise: to ensure that the act of running, updating, and defending a production service does not itself become the source of failure.

Up to 50% of system failures are caused by updates; new software versions, patches, configuration changes, and supply chain components introduced into a live environment. A further 10% are attributed to external disruptions such as cyber-attack[1]. The remaining failures arise from design weaknesses and operational stress, where performance load is beyond provisioned capacity, cascading dependencies, and bad data propagating through logic. This chapter addresses all three.

Chapter 4 - *Planning for Improved Availability* established the architectural and organisational foundation; the configuration baseline, dependency maps, SRE function, SLOs, and the FMEA-based vulnerability assessment. That planning work is the prerequisite for everything in this chapter. Chapter 6 - *Response and Recovery* takes over when prevention has failed; incident detection, diagnosis, escalation, and post-incident analysis. This chapter occupies the space between those two; the active, day-to-day disciplines that keep the system safe while it is running.

This chapter follows a delivery lifecycle:

- Regulatory Compliance and IT Service Delivery: alignment of regulations identified in Chapter 1 with the IT service delivery model
- Safe Upgrades and Supply Chain Management: controlling the introduction of change into the live environment
- Testing for Resilience: validating that the system can absorb both planned change and unplanned stress
- Active Prevention: defending the live system against saturation, failure propagation, and bad data
- Degraded Mode Operating Procedures: defining what operators do when prevention is insufficient but full recovery has not yet begun
- Key Takeaways: role-based accountabilities and checklists

The chapter aims to ensure the delivery teams can support the board to answer the following Strategic governance questions for board-level accountability:

- Have we declared our Important Business Services (IBS) and their Impact Tolerances: and are these reviewed at least annually?

[1] Nine Banks, https://www.bcs.org/media/rxdmjr5h/availability-6-report-nine-banks-data-roundtable-220425.pdf

- Can we demonstrate - not merely assert - that recovery capability has been tested within the last 12 months?
- Do our third-party contracts include Software Bill of Materials (SBOM) disclosure, staged rollout requirements, and rollback obligations? (contracting detail: Chapter 3.5)
- Are our Degraded Mode Operating Procedures (DMOP) current, offline-accessible, and rehearsed?
- Does our Board have a designated member accountable for operational resilience?
- Do we treat AI that calls external services or acts autonomously as a board-level resilience and supply-chain risk, and have we assured ourselves it is governed accordingly?

5.1.1 Regulatory Compliance and IT Service Delivery

The 2026 regulatory environment has materially shifted the context for service delivery. DORA was enforced from January 2025, with 2026 marking the transition from paperwork compliance to demonstrated proof of resilience. DORA applies directly to EU financial entities and their critical ICT third-party providers. It is referenced throughout this chapter as a leading-practice benchmark for ICT change management, testing rigour, and Register-of-Information discipline.

Non-financial UK organisations adopting these controls position themselves favourably under the UK Cyber Security and Resilience Bill (CSRB) and the FCA/PRA Unified Framework, which apply equivalent expectations. The UK CSRB extends NIS obligations to MSPs, data centres, and large load controllers, introduces 24-hour initial / 72-hour full incident reporting, and establishes penalties up to £17 million or 4% of global turnover. It is expected to pass into law in late 2026 or early 2027.

The following table summarises the regulatory frameworks imposing service delivery obligations relevant to this chapter.

Framework	Scope	Key Service Delivery Obligation	Reporting
UK CSRB (2026)[2]	UK Critical National Infrastructure (CNI), Managed Service Providers (MSPs), Data Centres	Security and resilience duties; supply chain risk management; NCSC CAF	24h initial / 72h full
EU DORA (Jan 2025)[3]	EU Financial Sector, Critical ICT	ICT risk governance; TLPT; Register of Information	4h classification; 1 month final
EU NIS2[4] (Oct 2024)	EU 18 Critical Sectors	Risk management; executive accountability; supply chain security	24h warning / 72h final
FCA/PRA Unified[5] (2026)	UK Financial Services	Impact Tolerance; tested recovery; unified incident portal	Impact Tolerance breach
ISO 22301[6]	Cross-sector	BCMS: BIA, continuity strategies, exercised plans	Per organisational policy
NIST SP 800-84[7]	Cross-sector (guidance)	Exercise and test programme design	Per organisational policy

These frameworks require tested, evidenced operational resilience at the service delivery level.

[2] https://www.gov.uk/government/collections/cyber-security-and-resilience-bill
[3] https://www.regulation-dora.eu/
[4] European Union (2022). Directive (EU) 2022/2555 — NIS2 Directive. Official Journal of the European Union, L 333/80.
[5] Financial Conduct Authority and Prudential Regulation Authority (2026). Unified Cyber and Operational Resilience Framework. London: Bank of England.
[6] https://www.iso.org/standard/75106.html
[7] National Institute of Standards and Technology (2006). NIST SP 800-84: Guide to Test, Training, and Exercise Programs. Gaithersburg, MD: NIST.

Engineering resilience for IT systems: Chapter 5 – Service Delivery

5.1.2 Topics in this chapter

This chapter covers the following topics
1: Safe Upgrades and Supply Chain Management (Section 5.2)
2: Testing for Resilience (Section 5.3)
3: Active Prevention (Section 5.4)
4: Degraded Mode Operating Procedures (Section 5.5)

The wheel below illustrates the lifecycle.

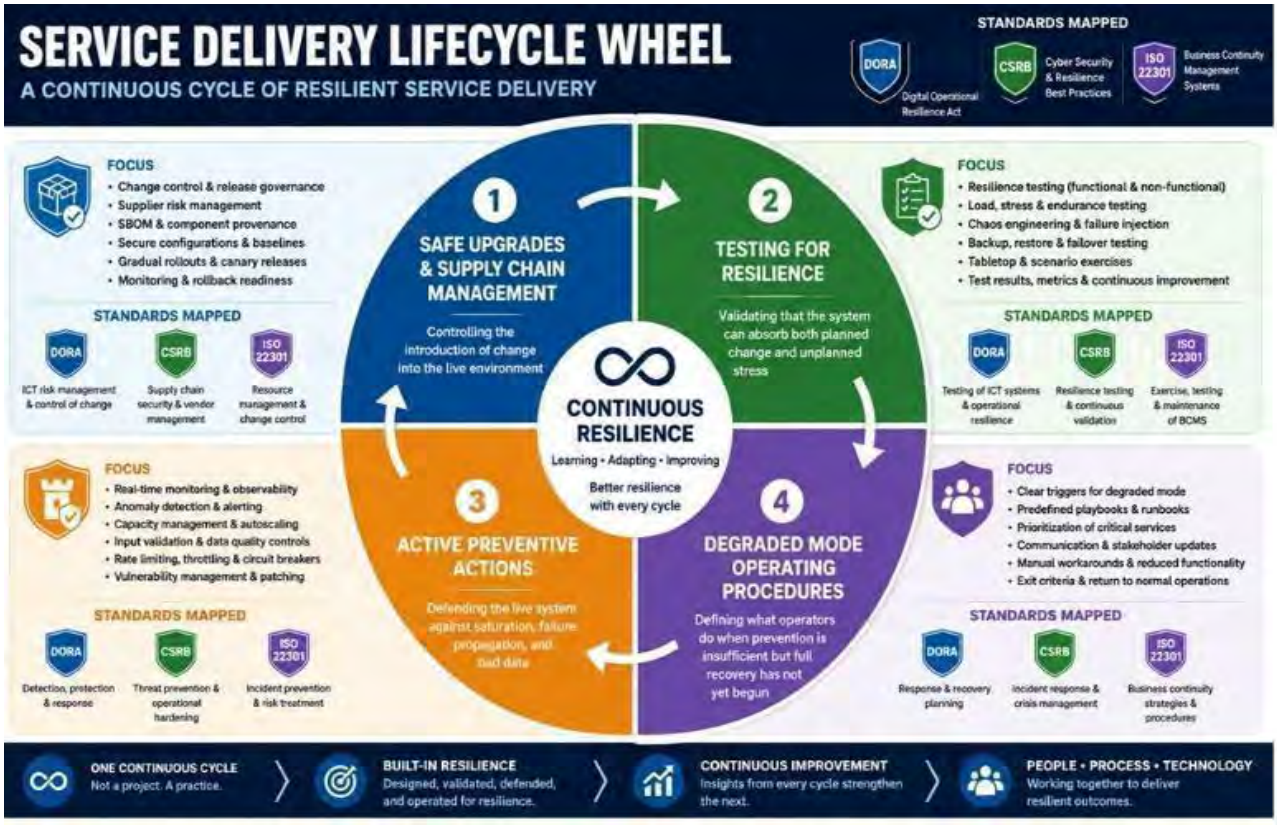


Diagram: source: https://trustedtech-consulting.com/resilience.html

Throughout the chapter, AI is treated not as a separate topic but as a capability embedded within each section, consistent with the approach taken in Chapters 4 and 6.

5.2 Safe Upgrades and Supply Chain Management

Updates can cause failures. This is not a theoretical risk; it is the documented primary cause of outages in production IT systems. A report from the Treasury Select Committee found that in nine major UK banks, over two years, nearly half of all outages were due to the introduction of new software versions into existing systems. The CrowdStrike incident of July 2024 demonstrated the same vulnerability at global scale. Managing the introduction of change is therefore central to the operational discipline of this chapter.

Upgrades of software and hardware are a central part of the management of service delivery. Supplier provided upgrades are initially considered in Chapter 3.4.4. Here we are concerned with deployment of these upgrades which involves effective change management and asset control.

5.2.1 The Update Risk

Upgrades arrive in two categories. Optional or advisory upgrades, driven by business needs or new requirements and are under the organisation's control; they can be scheduled, batched, and validated before deployment. Mandatory or supplier-driven upgrades, for example security patches, are applied remotely and are not under the organisation's direct control. Both carry risk; only the first is fully manageable.

For optional upgrades, the following strategies reduce blast radius:

- Batching: group updates and implement during a pre-advised maintenance window, reducing user-visible disruption
- Canary deployment: route a small percentage of live traffic (typically 1–5%) to the updated version; hold and observe before widening the rollout
- Blue/Green deployment: maintain two identical environments; deploy to the idle environment and validate before switching traffic
- Feature Flag On/off release control: create a configuration flag in the release to toggle the release on and off for controlled change management.
- Staged deployment: where possible deploy the release in stages with low risk business impact areas first
- Digital twin testing: validate upgrades against a synthetic replica of the production environment before touching live systems

For supplier-driven upgrades, the contract is the only control the organisation holds; notification requirements, maintenance windows, and rollback obligations must be defined contractually, with gaps escalated to the Board as a supply chain risk (see Chapter 3.4.2). Possible in-service defences include

isolation of the failed system and failover, streamlined recovery processes, and data protection through backup.

5.2.2 Supply Chain Volatility and Software Bill of Materials (SBOM) Management

The modern software supply chain — APIs, open-source libraries, CI/CD pipeline components, and third-party SaaS dependencies — changes daily, and each change is a potential injection point for failure. Supply chain and vendor management Chapter 3.5) owns the Software Bill of Materials (SBOM) discipline, defined in ISO/IEC 19770-2[8]

For financial services regulated firms, DORA Article 9 mandates ICT change management[9] [10] controls — changes to production systems must be tested, validated, and subject to rollback — with FCA/PRA SS1/21 applying equivalent expectations to the broader UK-regulated population.

Two topics first discussed in Chapter 3 are important for service delivery:

- The SBOM remains the main evidentiary artefact
- AI-generated code and autonomous AI agents.

Checklist: SBOM Audit (per ISO/IEC 19770-2)
The SBOM Audit checklist (per ISO/IEC 19770-2) is maintained in Chapter 3.5). Apply it in full at every deployment gate in this chapter: no third-party update enters production without validation in a mirrored environment, and rollback is defined for all automated update mechanisms before first deployment.

AI Impact: Vibe-Coded Applications, Agentic Processes, and the Extended Supply Chain

1. AI-Generated and Vibe-Coded Applications

AI-assisted development tools (e.g. Cursor, Claude Code, GitHub Copilot, Base44, Bolt) produce code whose dependencies, libraries, and architectural patterns were selected by the AI model, not the organisation's architecture team, and it enters the operational environment without passing the SBOM pipeline or staged deployment gates. The shadow-AI adoption evidence and

[8] International Organization for Standardization (2015). ISO/IEC 19770-2:2015 — IT Asset Management: Software Identification Tag. Geneva: ISO.

[9] European Union (2022). Regulation (EU) 2022/2554 — Digital Operational Resilience Act (DORA). Official Journal of the European Union, L 333/1.

[10] SAP and Oxford Economics, "The Value of AI in the UK: Growth, People & Data," SAP News Center, February 26, 2026.news.sap

its consequences for IBS classification are consolidated in Chapter 3.6) [11][12][13]

The risk that an AI capability presents scales with its integration boundary. An AI tool operating locally, using the organisation's own data and within the firewall, presents limited resilience risk. The moment it integrates with external services, reaches across data boundaries, or calls third-party endpoints, it becomes an unverified supply-chain channel and the controls in this section apply in full.

The governance rule is established in Chapter 3.6: AI-generated code is not exempt from supply chain governance, it is a new supply chain channel, subject to the same SBOM inventory, staged deployment, validation, and rollback requirements as vendor software, with every AI-generated dependency treated as an unverified third-party component until validated.

2. Agentic AI Processes as Service Components

Autonomous AI agents, systems that call APIs, query databases, make decisions, and trigger actions across multiple steps without human prompts are architecturally equivalent to a new service in the infrastructure. They must be treated under the same disciplines as any other service. Three specific risks apply:

- Agentic drift: the agent's behaviour changes when the provider updates the underlying foundation model, without the organisation's knowledge or consent, a supplier-driven update with no staging gate, precisely the risk category described in Section 3.4.4.
- Cascading action: an agent with tool-use permissions can propagate a bad decision through multiple downstream systems before human review occurs. Circuit breakers and compartmentalisation (Section 4.3.2) must extend to agentic action boundaries.
- Non-determinism: the same input may produce different outputs, making conventional ART and regression testing unreliable. Testing strategies for agentic components must include statistical validation over multiple runs, not single-pass functional testing.

Every agentic process should have a non-agentic fallback path defined in the DMOP (Chapter 5.5), and agentic drift should be monitored as a supplier-driven update event.

As change is known to cause most production failures, and because AI can generate change at a volume and velocity no manual process anticipated, AI-

[11] Matt Hertel, "62% of businesses have AI agents deployed," LinkedIn, April 10, 2026, citing Microsoft Cyber Pulse Report 2026,.linkedin

[12] Microsoft UK (2026). UK Cyber Pulse Report: AI Agent Deployment. March 2026.

[13] IBM Security (2025). Cost of a Data Breach Report 2025. Armonk, NY: IBM Corporation.

generated code and agentic actions must flow through the same ITIL change, problem and incident management processes as any other change. AOI changes should be considered as a parallel track.

3. Regulatory Exposure — Already in Scope

EU AI Act (August 2026)[14]: Under Article 3(3), anyone who builds an AI system, or has one built and places it on the market under their own name — is classified as a provider, with full conformity obligations (detail: Chapter 3.6). A business user who wraps an API in a custom UI and deploys it internally has become a provider. Penalties reach €35 million or 7% of global turnover for prohibited practices; €15 million or 3% for high-risk compliance failures. Any UK business operating in or selling to the EU market faces extraterritorial compliance requirements.

UK (Active 2026): The ICO's first report on agentic AI[15] The Data (Use and Access) Act 2025 reformed automated decision-making provisions under UK GDPR from February 2026 agentic processes making decisions about individuals are already subject to legal obligations. The anticipated UK AI (Regulation) Bill, expected in the second half of 2026, will likely introduce risk-based classification. On 28 April 2026[16], the UK Tech Minister announced plans to set deployment standards for the most capable AI models.

Governance advice: An AI Asset Register - defined and owned in Chapter 3.5 as a living, risk-tiered inventory of AI-generated code, AI-assisted applications, and agentic processes with named ownership — could be maintained alongside the SBOM. An approved-AI-tool allowlist could be created as a companion control to the AI Asset Register: tools not on the allowlist will not be permitted to process organisational data or integrate with production services until approved, and use of unapproved tools is treated as a shadow-AI exposure for monitoring and escalation.

5.2.3 Deployment Strategies for Reduced Risk

Staged deployment is the primary method for controlling risk from update-induced failures. The principle is consistent: reduce the blast radius of a bad

[14] https://cy.ico.org.uk/about-the-ico/media-centre/news-and-blogs/2026/01/ai-ll-get-that/

[15] UK Parliament, "Artificial Intelligence (Regulation) Bill [HL]," Bill 76 (as introduced), March 4, 2025,

[16] Bird & Bird (2026). UK AI Regulation: Government Announces Plans to Set Standards for AI Deployment. 28 April 2026.

deployment by limiting its initial scope, validating behaviour at each gate, and maintaining a fast rollback path.

The key architectural requirement is that rollback must be planned, automated or near-automated. Deployment pipelines should ensure that reverting to the previous known-to-be-good state is a single, validated, low-friction operation.

Example: CrowdStrike Falcon Update — July 2024

What happened: In July 2024, CrowdStrike triggered a global outage impacting up to 8.5 million Windows endpoints across critical sectors. Devices experienced system crashes (Blue Screen of Death) and boot failures, rendering them unusable without manual intervention. The failure impacted devices instantly due to a simultaneous global deployment via the Falcon platform, creating a widespread availability incident.

Key failures: No staging gate for content updates; no automated rollback; no degraded-mode operating procedure for systems dependent on the endpoint agent.

Lessons for this chapter: Mandatory staging gates apply to all update types, including content and signature updates, not just version releases. Rollback capability must be planned, automated, tested, and assigned to a named role. The vendor-contract dimension of this incident is analysed in Section 3.4.4.

Example: Cloudflare Bot Management Outage[17] — November 2025

What happened: A database permissions change caused the Bot Management feature file to double in size, exceeding a hardcoded limit. Traffic routing crashed globally, affecting 2.4 billion users across ChatGPT, X, Spotify, and Shopify for six hours.

Key failures: No staging gate for configuration propagation; hardcoded memory limit with no graceful fallback; circular service dependencies (Turnstile required for Dashboard login, but Turnstile was down).

Resolution: Code Orange: Fail Small[18], company-wide programme ensuring configuration changes are staged and size-bounded before global propagation. Supply chain lens: Section 3.4.4).

Lesson: Configuration updates should be subject to the same staging discipline as code deployments.

[17] Cloudflare (2025). Cloudflare Outage on November 18, 2025. Cloudflare Blog, 21 November.

[18] Cloudflare (2025). Code Orange: Fail Small. Cloudflare Blog, 30 December.

5.3 Testing for Resilience

Most testing focuses on functional correctness: does the system do what the specification requires? This is necessary but insufficient. A system that passes every functional test can still fail catastrophically under load, recover from failure incorrectly, or fail silently in ways that are not visible until the damage is done.

Resilience testing addresses a different set of questions: what happens when this component fails? Can the system absorb the failure? Does it degrade gracefully or collapse? Does it recover correctly and completely? These questions cannot be answered by functional test suites. They require a deliberate, adversarial testing posture, and they require that testing to be continuous, not a one-time pre-launch exercise.

All staged deployment decisions from Section 4.2 should be validated through the testing disciplines described here before go-live. Testing is the gate, not the afterthought.

5.3.1 Beyond Functional Testing: Non-Functional Requirements

Non-functional requirements (NFRs) define how a system behaves under conditions rather than what it does. They include:

- Performance: response times under normal and peak load
- Reliability and availability: mean time between failures (MTBF); behaviour under sustained stress
- Scalability: capacity to handle growth without architectural change
- Recoverability: time and integrity of restoration after failure
- Testability: the degree to which the system can be instrumented and observed during testing

NFRs are routinely omitted from acceptance testing because they were not included in the original specification. This is a governance failure, not a technical one. NFR specifications should state both the BAU reliability target MTBF and Mean Time To Repair (MTTR) and the Maximum Tolerable Downtime (MTD) Tolerance for each IBS.

As a general advisory consideration resilience and recoverability should be a named, explicit non-functional requirement in every service design, for Financial Services (FS) systems this is consistent with FCA/PRA Operational Resilience expectations and more broadly the BCI Good Practice Guidelines[19] on BCM implementation.

[19] Business Continuity Institute (2023). Good Practice Guidelines (GPG). Caversham: BCI.

5.3.2 Testing the Change: Regression Testing for Resilience Properties

Without rigorous testing of the change release itself the discipline of staged deployment (Section 5.2.3) is meaningless. DORA Article 25 mandates "testing and validation" before deployment.

The table below maps each NFR to its characteristic test type providing single vocabulary when scoping regression coverage for a change.

Non-Functional Requirement	Characteristic Test Type	Purpose
Performance	Load and stress testing	Validate response times and throughput under normal, peak, and breaking-point conditions.
Reliability / Availability	Chaos engineering and Automated Recovery Testing (ART)	Surface latent failure modes and confirm self-healing behaviour holds under induced fault.
Scalability	Stress, Performance and Capacity testing	Confirm the system absorbs growth horizontally and vertically without architectural change.
Recoverability	ART and restore validation	Measure actual RTO and RPO against committed targets, including backup integrity proof.
Testability	Instrumentation review	Verify the system exposes the data collection, logging and audit needed to observe, diagnose, and validate the four NFRs above.

Whilst a change can be functionally correct it can still break availability, introduce latency, or degrade recovery capability. This subsection, Testing the change, focuses on the question: does this update preserve the resilience and non-functional properties established from Chapter 4.5 architectural recovery patterns.

Checklist: Testing strategies for safe change introduction:

Regression Testing for Resilience Properties: Execute baseline NFR tests (Section 4.3.1) against the change in isolation. Does performance remain within SLO? Does the availability calculation remain accurate? Are error rates stable? Validate that no hidden dependencies or side effects have been introduced.

Canary Validation: Route 1–5% of live traffic to the changed version; hold and observe for anomalies (latency percentiles, error rate, saturation trends) before expanding scope. Metrics gates determine whether to proceed to the next stage or rollback. Note: Canary release is also referenced earlier in section 4.2.1 where the approach to limit traffic for a software release reduces the blast radius of the change.

Blue/Green Verification: Maintain two identical production environments; deploy the change to the idle environment, run full NFR test suite, validate data consistency, then switch traffic. If failure occurs, switch back instantly.

Feature-Flag Testing: Confirm the flag toggles correctly, that enabling the feature introduces no side effects, and that disabling it reverts the system to pre-change state. Functional and regression testing conducted for both flag settings, pre-change state and post-change state. This is particularly critical for AI-generated or AI-assisted code (Section 4.2.2 AI Impact).

Automated Rollback Testing: Validate that reverting to the previous version is non-destructive and automated. A change without a tested rollback path is not deployable. Note: It is important as part of deployment planning to understand the rollback timings at different stage gates during the rollout.

Contract and API Tests: For microservices architectures, validate that service-to-service contracts remain satisfied. A change in one service can silently break a downstream dependency, contract tests catch this before production impact. This is particularly important considering the complexity of current architectures where external services can form part of the IBS (see section 4.3.7).

Example: CrowdStrike - July 2024

This example revisits the CrowdStrike incident through the lens of regression testing for resilience properties, complementing the deployment-strategy analysis in Section 4.2.3.

What happened: (see Section 4.3.2)

Key failures: A defective content update interacting with the Microsoft Windows OS caused OS-level crashes. The update was classified as low-risk data "content" rather than configuration, bypassing full regression and boot-path testing, precisely the NFR validation gap this section addresses. There was no staged rollout (canary or ring deployment) and no effective kill switch, meaning the fault could not be contained or rapidly reversed once deployed.

Lesson: Regression testing should cover resilience properties, not functional correctness alone. High-privilege changes should be governed as systemic risk, with boot-path, fault-injection, and OS-interaction layer testing built into acceptance criteria (consider this mandatory under DORA Article 25 for financial services). Resilience needs to be proven through testing, not assumed through classification.

5.3.3 Chaos Engineering

Chaos Engineering is the practice of intentionally introducing controlled disruptions into a live or near-live system to observe how it responds. The purpose is not to cause damage; it is to surface failure modes that exist but have not yet manifested in production.

The discipline originates from Netflix's Chaos Monkey programme, which randomly terminated production instances to validate self-healing mechanisms. In 2026, the approach has matured into a structured engineering practice: hypotheses are formed about expected system behaviour, experiments are designed to test those hypotheses, and findings are fed directly into the architectural backlog.

Chaos Engineering should be scoped to Important Business Services (IBS) first. Note: Important Business Services (IBS) and Impact Tolerances are defined in Section 1.4.2 and the Glossary, this chapter uses the terms operationally without redefining them.

Experiments should begin in a staging environment and progress to production only once baseline behaviour is understood and rollback procedures are confirmed operational.

5.3.4 Automated Recovery Testing (ART)

Automated Recovery Testing: intentionally induces specific failures to measure recovery speed and integrity against defined Recovery Time Objectives (RTOs). Where Chaos Engineering explores unknown failure modes, ART validates known recovery procedures, confirming that the playbooks work, that the automation triggers correctly, and that the system returns to a verified known-good state within its committed Impact Tolerance. Results of ART should be readily available to the incident management team discussed in Chapter 6.

Backup Restore Testing: A backup that has never been restored is not a backup; it is an untested assumption. Backup restore testing is a mandatory component of ART, not optional.

For each IBS, execute quarterly restore validation:

- Restore from backup to an alternate environment or alternate hardware (not production)
- Confirm segregation of backups from core enterprise network/storage
- Verify data integrity: row counts, checksums, referential integrity, business logic invariants
- Measure actual RTO/RPO during the restore operation against committed targets
- Confirm the option to rollback if data corruption is detected during validation
- Document restoration time, data completeness, and any anomalies encountered

ART should be run regularly against all IBS-supporting systems, at minimum annually, and after any material architectural change. No change should enter production without documented evidence of passing ART resilience regression tests.

Example: GitLab - January 2017
What happened: GitLab, at the time a rapidly growing DevOps platform, experienced a major production incident resulting in approximately six hours of user data being permanently lost. A routine maintenance task on the production database went wrong when an engineer accidentally issued a destructive command that deleted live data from the primary PostgreSQL database.

Key failures: Five separate backup mechanisms were in place; PostgreSQL (Structured Query Language) replication, LVM (Logical Volume Manager) snapshots, WAL (Write-Ahead Logging) archiving, backup.gitlab.com, and NFS (Network File System) snapshots, none functioned when needed. The processes required to recover systems were slow, manual, and unrehearsed.

Lesson: It was the absence of rehearsed and reliable recovery procedures, identifiable through regular restore testing, that could have prevented this. Backup quantity provides no assurance without restore validation. Every mechanism in the list above would have appeared compliant on a controls audit, however this is only useful when proven under recovery conditions.

Advice: ART Results and Recovery Confidence Scoring: *The Board may wish to consider ART results reported as evidence of tested recovery capability, not merely planned capability. Report quarterly to the Board, with escalation triggered if any IBS scores below 0.85 Recovery Confidence or if ART execution fails.*

Recovery Confidence Scores are a composite measure of recovery speed against RTO, data integrity against RPO, and completeness of human response.

- RTC= (Recoveries achieved within RTO/ Total recoveries tested) ×100
- Provide a percentage of ART / tests /incidents achieved with RTO.
- The Board can set an escalation point e.g. if any IBS scores below 0.85, or if ART execution fails.

This reframes resilience from a compliance checkbox to a strategic risk indicator directly tied to Impact Tolerance.

5.3.5 Wargaming and Failures as a Service

Wargaming is defined as simulated challenge exercises involving both technical and leadership teams. It serves two purposes. First, it validates the human and process elements of recovery: who makes decisions, on what authority, using what communication channels. Second, it surfaces assumptions that have never been tested.

Failures as a Service (FaaS) extends this approach by allowing failure scenarios to be triggered on demand in a controlled environment. Larger organisations have introduced digital twin strategies to provide this capability: this is a synthetic replica of the production environment on which failures can be induced, observed, and resolved without risk to live services.

Example: Southwest Airlines[20] — December 2022

What happened: A winter storm triggered cascading failures in crew-scheduling software. Southwest cancelled 72% of flights over a week while competitors operated in single digits. Voice phone calls as a crew-scheduling fallback were overwhelmed by a 9-hour wait time.

Key failures: Decades of IT underinvestment; no tested Degraded Mode Operating Procedures (DMOP) for crew-scheduling failure at scale; manual backup that could not absorb the volume.

Lesson: Wargaming should be considered to simulate the saturation of manual workarounds, not just the initial switchover.

[20] Southwest Airlines (2023). Review of December 2022 Operational Disruption. U.S. Department of Transportation Records.

5.3.6 Observability 2.0 and Precision Testing

Traditional monitoring (Monitoring 1.0) focuses on predefined metrics and static thresholds of known-knowns, monitoring what we already know may break.

It relies on three pillars: metrics, logs, and traces.

- Metrics provide numerical time-series data, such as CPU usage, queue depth, latency, and error rates.
- Logs capture discrete events, including errors, access records, and system actions.
- Traces show the end-to-end path of a single request across multiple services.

Accurate time synchronisation across infrastructure and applications is important for incident investigation, distributed tracing, and forensic timelines. Establish a consistent approach (for example, managed Network Time Protocol (NTP) services with appropriate redundancy) and verify synchronisation as part of operational monitoring.

Observability 2.0 preserves raw, detailed data for ad-hoc hypothesis testing of unknown-unknowns. Integrating Observability 2.0 with Chaos Engineering provides detailed data for targeted approaches often referred to as Precision Disruptions: AI agents analyse both the raw data and the dependency paths and inject failure at exact points of systemic fragility. This meets DORA TLPT requirements (Article 26)[21] and NCSC CAF 4.0 Principle B5[22] on Recovery Sequencing.

Capability	Monitoring 1.0	Observability 2.0
Data Focus	Predefined metrics (known-knowns)	Detailed and extensive data capture (unknown-unknowns)
Posture	Alerting/Reactive threshold alerts	Exploring/ Proactive hypothesis testing
Analysis	Siloed tools, manual correlation	Unified visibility, AI-enhanced correlation

[21] European Union (2022). Regulation (EU) 2022/2554 Article 26 — TLPT Requirements (Threat-Led Penetration Testing). Official Journal of the European Union.

[22] National Cyber Security Centre (2025). Cyber Assessment Framework (CAF) 4.0. London: NCSC.

Resilience Goal	High availability (uptime)	Anti-fragility (recovery and learning)
Testing	Validates known failure modes	Discovers unknown dependency paths

5.3.7 Third-Party Dependency and Failover Testing

Dependency Testing: Given the prevalence of SaaS and third-party integrations (e.g. Stripe, Autho, CDN (Content Delivery Network) providers, KYC (know your Customer) vendors), resilience testing should include vendor degradation scenarios. The vendor-management dimension — including NFR validation and the feed into Service Level Management — is governed in Section 3.4.4).

- Implement synthetic transaction testing against vendor APIs;
- execute contract tests to validate integration assumptions;
- inject vendor slowness or outage conditions to validate timeout and circuit-breaker logic.

Results validate that posted RTO/RPOs remain achievable under defined vendor-outage durations and feed Service Level Management and SLA negotiation (Section 3.4.4).

Failover Testing: To distinguish full-scale DR exercises from ART, Failover testing design scope should begin from the core infrastructure through to frontline service operations of the enterprise. This would include non-intrusive/non-service impacting activities such as physical inspection, logic and configuration checking, to intrusive/ service-impacting; site, datacentre, key service centre and IBS shutdowns.

Annual recovery testing is a regulatory requirement (DORA Article 24(6), ISO 22301 Clause 8.5, FCA Operational Resilience); the particular means and extent of testing has some discretion. For other sectors the board should consider to execute annually a tested failover. Depending on the architecture, testing services across to alternate data centre or regions, with actual operations teams executing the procedure at realistic constraints e.g. 3am, with degraded alerting and without guaranteed escalation paths, this evidences resilience. rather than a scripted, daytime exercise with full support that tests the script, not the system.

The scope would be up to full live operations to include network failover validation (BGP failover, DNS cutover, WAN circuit restoration) through to non-IBS, IBS and partner network services.

This is distinct from wargames; it is a live procedural validation of major business impacting service failures. This includes operational staff, site shutdowns and staff movements and business contingency measures. This is a major undertaking and requires months of planning for an effective comprehensive test.

Checklist: Resilience Testing (Per IBS)

- Define NFR acceptance criteria for every IBS: performance, availability, scalability, recoverability
- Run Chaos Engineering experiments in staging before introducing to production scope
- Execute ART at least annually and after any material architectural change.
- Conduct at minimum one wargame per year covering each IBS and its supporting teams.
- Ensure IBS operational teams have well understood documented procedures ready in case of an event.
- Validate rollback procedures as part of every ART exercise — not separately
- Report ART results to the Board as evidence of tested recovery capability
- Maintain a digital twin or isolated sandbox environment for FaaS exercises

AI in Practice: AI in Resilience Testing

AI-driven chaos simulation: AI agents can perform intelligent chaos engineering — identifying the specific weak points in a complex microservices architecture and suggesting targeted disruption experiments, rather than applying random failures. This focuses testing effort on the highest-risk dependency paths.

Digital twin acceleration: ML-enhanced system-dynamics models simulate how a complex socio-technical system will behave under stress, including cascades and feedback loops that would not be visible in analytical assessment. Synthetic data and digital twins allow safe stress-tests of rare events — including concurrent failure scenarios such as cyberattack combined with vendor outage and demand surge.

Automated Root Cause Analysis during ART: AI correlates alerts and telemetry across the stack to pinpoint the origin of an induced failure in seconds, reducing the time engineers spend on diagnosis and enabling faster iteration across test scenarios.

5.4 Active Prevention

Testing validates the system as designed. Active prevention defends the live system day-to-day against capacity saturation, failure propagation, and bad data reaching logic that cannot handle it. These are not reactive controls. Following on from Chapter 3, these are engineering decisions made at design time and operational policies defined before the stress event occurs.

5.4.1 Day-to-Day Monitoring

Prevention depends on sight. Capacity headroom, load shedding thresholds, compartment boundaries and validation guardrails are all set against evidence that something is drifting. Monitoring is how that evidence arrives. Section 4.2.4 defines the observability architecture, and Section 6.2.2 covers how signals are read once an incident is running. This section covers what sits between them, the discipline of running a live service with continuous sight of its condition.

Monitoring is a continuous regulatory control, not a project. For financial entities in scope of DORA, Articles 9 and 10 require continuous monitoring and control of ICT systems, and mechanisms to promptly detect anomalous activity against defined alert thresholds. NIST CSF 2.0 places Continuous Monitoring (DE.CM) within the Detect function. NCSC CAF Objective C covers security monitoring through C1, and proactive discovery through C2 Threat Hunting. For UK regulated firms, monitoring is what identifies that an important business service can remain within its impact tolerance. For entities in scope of NIS2, Article 23 requires you send an early warning within 24 hours of becoming aware of a significant incident. Slow detection eats into that window.

Coverage should be designed against the dependency map (Section 4.2.4), layer by layer, not assembled tool by tool. Gaps between the layers are where undetected degradation lives, because each adjacent team assumes the other is watching.

In the table below, the right-hand column identifies where each layer is recognised in framework guidance. It does not prescribe the implementation.

Monitoring layer	What it detects	Recognised in
Physical and environmental	Power, cooling, fire suppression, comms room conditions	NIST CSF DE.CM-02
Device and endpoint	Agent health and coverage, patch state, unmanaged devices	Supported by NIST CSF DE.CM-09
Network and connectivity	Path availability, latency and loss; for example, BGP changes, DNS resolution, circuit health	NIST CSF DE.CM-01
Platform and infrastructure	Compute, storage, memory and queue saturation; capacity headroom	ITIL capacity management
Application	Golden Signals; dependency call health; circuit-breaker state	DORA Art. 10; internal SLI/SLO standards
Data and integrity	Validation failure rates, replication lag, reconciliation breaks, canary record hits (Section 4.4.5)	NCSC CAF C1 Security Monitoring
User experience	Journey completion, session errors, perceived response time by channel	Supports measurement against impact tolerance (FCA/PRA SS1/21)
Business transaction	Throughput against expected volume; backlog against impact tolerance	ISO 22301 performance evaluation principles
Supplier and third party	Vendor API health, status-page state, SLA breach, sub-processor availability	NIST CSF DE.CM-06

Monitoring Coverage: Produce a coverage statement for each IBS. It records which layers are instrumented, which signals map to that service's impact tolerance, and which layers are knowingly not covered. Review it monthly and after any material architectural change. A declared gap is a governed risk. An undeclared gap is an audit finding.

Active and Passive Monitoring: Passive monitoring observes real traffic; logs, metrics, traces and real-user monitoring (RUM), and it is authoritative about user impact. Active monitoring generates traffic; synthetic transactions, health probes and heartbeats, and it is the only signal available where there are no users. That covers a low-volume service overnight, a failover target carrying no load, and a recovery site untouched since the last test. Those are the conditions under which recovery capability matters most. Both are required. A warm standby that no synthetic transaction has exercised is not a capability; it is an assumption.

Probe Design: Probes should assert on content, not on the response code. A service can respond quickly and return HTTP 200 while the payload is stale or corrupt (Section 6.2.1), and a status-only check will report that as healthy. Use synthetic test accounts and non-production data, so that no customer data appears in a probe assertion.

Warnings and Alerts: ITIL 4 distinguishes informational, warning and exception events. Treat warnings as leading indicators that still permit preventive action, and alerts as actionable notifications where impact has started or is imminent. Most resilience value sits in the warning tier, and most estates under-invest in it. Static thresholds produce useful alerts. Trend, saturation and error-budget-burn conditions produce earlier ones. Multi-burn-rate alerting against per-IBS service level objectives (Section 4.4.1) ties both back to the impact tolerance the objective was derived from.

Alert Discipline: Every alert needs a named owner and a linked runbook. An alert with neither is not a control; it is noise, and it should be deleted. Every alert should state user-facing impact rather than technical symptom: "payments backlog 20 minutes against a 60-minute tolerance", not "queue depth 40,000". Review alert volume weekly against incident count. Volume rising without a corresponding rise in incidents indicates threshold drift, not a deteriorating service. Time-box every suppression and expire it automatically. A permanently suppressed alert is an undeclared coverage gap.

Advice: Alert fatigue is a resilience risk, not an operational inconvenience. An engineer the system has trained to dismiss a recurring alert will dismiss the one that matters.

Monitoring Independence: A monitoring stack that shares critical failure domains with the service it observes may be unavailable during the incident. Use push and pull monitoring data together. Push reporting survives some polling-path failures, and pull checks detect monitoring data that has stopped arriving. Host alerting and paging outside the failure domain they watch, retain out-of-band management for infrastructure, and maintain one notification channel that depends on neither the corporate network nor the primary identity provider. The Cloudflare outage examined in Section 5.2.3 shows the failure mode; engineers

could not log in to fix the outage because authentication was part of it. Any tool that requires the failing service to work is not a control.

Devices and Connectivity: Application monitoring is well served by commercial platforms. The layers beneath it are where avoidable outages start. Monitor endpoint agent health and coverage rather than presence, because an agent that has stopped reporting still shows as installed. Treat the agent itself as a supplier-controlled change vector (see section 5.2.1). Instrument path health end to end, including failover paths in normal running. An unmonitored redundant circuit is not redundancy; it is an unverified assumption. Monitor certificate, credential and licence expiry as warning-tier signals, internal certificates included. These are among the most predictable outages in enterprise IT, and among the most common.

Supplier Signals: Vendor governance is discussed in Section 3.4.4. During service delivery, the responsibility is to ensure that signals from suppliers reach the same alerting platform as internal monitoring data, at the same latency. Poll vendor status pages and API health endpoints automatically, rather than reading them once an incident has begun, and run the vendor synthetic tests (Section 5.3.7) continuously against live critical-path dependencies. For ICT services supporting critical or important functions, DORA requires contractual provisions for ongoing performance monitoring and rights of access, inspection and audit. Where a provider will not expose system health monitoring data, escalate it (see Section 3.4.2) as a resilience finding. Several nominally independent suppliers relying on the same cloud region can create a material common-cause risk, and should be assessed as one dependency.

Cadence and Reporting: Detection runs continuously. The organisation consumes it on a rhythm. Open warnings and active suppressions pass at shift handover. Capacity, failed probes and the expiry horizon are reviewed daily.

Checklist: Day-to-Day Monitoring

- Maintain a coverage statement per IBS recording instrumented layers, mapped signals and declared gaps
- Run continuous active checks against every IBS, failover target and critical third-party dependency
- Assert probes on content, not response code; use synthetic accounts and non-production data
- Implement multi-burn-rate alerting against per-IBS SLOs, derived from impact tolerance
- Give every alert a named owner, a linked runbook and a user-facing impact statement
- Review alert volume against incident count weekly; time-box and auto-expire every suppression

- Host alerting and paging outside the failure domain they monitor; verify one out-of-band channel
- Monitor endpoint agent health and coverage, and path health including failover paths in normal running
- Monitor certificate, credential and licence expiry with escalating warning periods
- Poll vendor health endpoints automatically; secure contractual monitoring rights; assess common-cause supplier dependencies

Alert volume and threshold drift are monitored weekly, coverage monthly. Availability, impact tolerance headroom and detection latency are reported to the Board quarterly. Detection latency belongs at Board level because it determines how much of a regulatory reporting window remains once an incident is declared. The clock starts on awareness. Late detection does not delay the deadline; it shortens the response. After every incident, revisit thresholds and coverage: was the signal present, was it seen, was it actionable.

5.4.2 Managing the Margin of Error

One of the most common causes of self-inflicted outages is resource provisioning at levels that leave no margin for variability in demand. A system running at 90% CPU utilisation under normal load will fail under a 10% demand spike. This is not a failure of the infrastructure; it is a failure of planning.

ITSM (IT Service Management) and ITOC (IT Operations Centre) principles recommend provisioning at no more than 70–80% of peak theoretical capacity. The remaining headroom is the operational buffer that absorbs demand spikes, maintenance overhead, and the processing cost of monitoring itself. Provisioning decisions should be explicitly tied to each IBS Impact Tolerance threshold.

Managing the Margin of Error: Resource provisioning should include a safety buffer defined against each IBS Impact Tolerance threshold. Capacity should be able to absorb demand spikes without triggering degradation. Review provisioning levels quarterly and after any material change to traffic patterns, architectural changes, changes in service scope or changes interfacing technologies. Traffic patterns may not scale in a one-to-one ratio, growth in one area can lead to exponential growth in another, traffic modelling is required to ensure capacity, another area where AI can support engineering teams in impact discovery.

Example: Error budget[23] for a known Service Level Objective (SLO)
For a service with a 99.9% availability, with an SLO (see 4.4.2) measured over a 30-day window the error budget can be calculated as:

Error budget = (100% - availability) * time-period
= (100−99.9) × 30days = 0.1% × 43,200 minutes ≈ 43.2 minutes

Tighter SLOs shrink the error budget dramatically:
- 99.95% delivers an error budget of roughly 21.6 minutes per month,
- 99.99% delivers an error budget of just 4.3 minutes.

Burn rate measures how quickly the budget is being consumed relative to the SLO window. A burn rate of 1 consumes the entire budget exactly at the end of the period; a burn rate of 10 consumes it in one-tenth of the time. Modern practice uses multi-window, multi-burn-rate alerting rather than static error-rate thresholds:

Interpreting Burn Rates
- Burn Rate = 1.0: Perfectly on target. You will use up your exact error budget at the end of the window (e.g., at the end of 30 days).
- Burn Rate > 1.0: You are burning through your budget too fast. If sustained, you will breach your SLO before the time window ends.
- Burn Rate < 1.0: You are running below the allowed error rate. The service is highly reliable, which might indicate room to move faster with deployments.

This approach catches acute incidents quickly while still surfacing slow degradations that a single threshold would miss.

Increasingly, regulated organisations link internal error budgets to externally committed impact tolerances, making the budget a direct translation layer between engineering telemetry and board-level resilience reporting[24].

5.4.3 Load Shedding Implementation

Systems will typically be designed to automatically drop non-essential background tasks during periods of high saturation. Load shedding decisions are which workloads, at what threshold, in what sequence, these can be pre-defined

[23] See https://oneuptime.com/blog/post/2025-09-03-what-are-error-budgets/view

[24] The site reliability workbook: Practical ways to implement SRE. O'Reilly Media. Beyer, B., Murphy, N. R., Rensin, D. K., Kawahara, K., & Thorne, S. (Eds.). (2018).

in the DMOP (see Section 4.5). Operators should generally execute a pre-authorised plan, not make real-time triage decisions under pressure.

AI in Practice: AI in Active Prevention

Predictive telemetry and early warning: AI-driven analytics applied to live telemetry, incident history, and external threat intelligence can surface patterns that historically precede material disruption enabling preventive action before rather than during a failure. This extends the Golden Signals monitoring approach (Latency, Traffic, Errors, Saturation) with predictive rather than reactive alerting.

Anomaly detection in data flows: ML models establish baseline behaviour for data validation patterns. Deviations; unusual lookup table query failures, format check rejection rates rising above baseline are surfaced as early warning signals, identifying data-layer issues before they reach logic-based shutdown conditions.

The monitoring AI is itself a monitored dependency: model drift reduces detection quality silently. Maintain a non-AI fallback alerting path, version and log material model changes, and monitor false negatives where they can be measured. Where such a system is classified as high risk under the EU AI Act, deployers must monitor its operation in accordance with the provider's instructions (Art. 26(5)), and providers must maintain post-market monitoring (Art. 72). Classification turns on intended purpose, not on use within a regulated service.

5.5 Degraded Mode Operating Procedures (DMOP)

Degraded Mode Operating Procedures define what the organisation does when prevention has not worked, but full recovery has not yet begun. They occupy a specific and important operational space, the period between a failure occurring and Chapter 6's incident management process taking full control.

The key distinction that gives DMOPs their value is that they are pre-authorised. The decisions they contain; which services to drop, which manual workarounds to activate, which stakeholder communications to issue, these decisions were made in advance, at a time when there was no pressure, no ambiguity, and no time constraint. When an operator invokes a DMOP at 2 AM, their job is execution, not judgment. In 24/7 Ambulance Operations the chain of command has clear instruction, there are well rehearsed fallback systems both manual and electronic to ensure should an IBS fail DMOP transition incident management and recovery is smooth, minimizing impact to emergency response.

ISO 22301[25] clause 8.4 (Business Continuity Procedures) and the BCI Good Practice Guidelines both require documented and rehearsed downtime procedures. The DMOP is the operational expression of that requirement at the service level — specific to each IBS, accessible offline, and assigned to a named role.

The contrast between Marks & Spencer and the Co-operative Group in April 2025[26] demonstrates these disciplines in practice. Both were targeted by the same threat actor using identical social engineering. The Co-op detected the breach within minutes and maintained customer services throughout. M&S did not detect the intrusion for two days; online sales were suspended for 46 days with an estimated £300 million financial impact. The difference was not technology spend; M&S had tripled its cyber headcount. The difference was rehearsed operational readiness: war-gamed incident response, practised manual fallbacks, and a culture that assumed breach as inevitable.

5.5.1 DMOP Structure and Content

A DMOP for each Important Business Service should contain:

- Trigger conditions: the specific thresholds or events that activate the DMOP measurable criteria, not subjective judgements
- Authorisation: the named role empowered to invoke the DMOP without seeking further approval

[25] International Organization for Standardization (2019). ISO 22301:2019 — Security and Resilience: Business Continuity Management Systems. Geneva: ISO.
[26] Marks and Spencer plc (2025). Cyber Update. London: M&S Corporate.

- Load shedding sequence: the pre-defined order in which non-essential workloads are dropped, with explicit criteria for each step
- Manual workaround procedures: paper-based or alternative-system procedures that allow core business functions to continue at reduced capacity
- Communication templates: pre-drafted stakeholder messages for internal and external audiences, specific to the degraded mode scenario
- Escalation threshold: the point at which the DMOP transfers control to the Chapter 6 incident management process and the Major Incident Management (MIM) team
- Offline accessibility: the DMOP should be physically accessible without dependency on the systems that are failing, these are often kept stored on laminated cards, offline tablets, or printed binders maintained at operational posts

5.5.2 DMOP Rehearsal

A DMOP that has never been rehearsed is a document, not a procedure.

Rehearsal requirements:

- Include DMOP invocation in all ART exercises and wargames (Section 5.3.3)
- Test offline accessibility: confirm that the DMOP can be located and used without access to the systems it covers
- Test communication templates: confirm that stakeholder contact lists are current
- Record rehearsal outcomes and update the DMOP accordingly, treat it as a living document, not a compliance artefact
- Report back to Board with outcomes, lessons learned and recovery metrics

5.5.3 Cognitive Aids and the First 60 Seconds

Advice: Under extreme stress, operators default to fight-or-flight mode. The DMOP should include cognitive aids: laminated checklists for the first 60 seconds:

- STOP. *Do not make any further changes to the system.*
- Assess, *Could this be a cyber incident (see below)?*
- RECORD. *Write down exactly what changed and the time.*
- ALERT. *Notify the designated incident owner via the pre-defined contact method.*
- DO NOT IMPROVISE. *Follow the DMOP steps in order.*

If there is any possibility of the incident being cyber related, preserve evidence and do not restore or re-image until a cyber triage decision has been made. Treating a cyber event as a routine outage can destroy forensic data and re-introduce the threat, Note: the analysis will

fall within the Cybersecurity discipline, for the purposes of reliability engineering support will be given to restore services once cyber investigation has determined the best return to service approach.

Example: UK Government Digital Service — Rehearsed DMOPs
For GOV.UK Pay, GDS teams rehearse manual fallback procedures[27] quarterly. If automated payment reconciliation fails, a trained finance team executes a secure spreadsheet-based process with dual authorisation. The principle: a DMOP is a living, practised capability.

Example: British Airways IT Outage — May[28] 2017
An engineer disconnected a power supply at a data centre. Upon reconnection, an uncontrolled server recovery cascaded through the facility. Approximately 75,000 passengers were stranded. The estimated cost exceeded £80 million.
Key failures: No DMOP prescribing controlled circuit-by-circuit restoration; no mandatory wait-and-check gates; unavailability of trained staff for complex recovery.
Lesson: Power restoration is a degraded mode operation requiring a prescribed, rehearsed, gate-controlled procedure.

5.5.4 Scalable DMOPs

Example: Southwest Airlines Meltdown — December 2022
Now looking through the lens of prevention case study from section 5.3.5
What happened: (See section 5.3.5)
Key failures: No tested DMOP for crew-scheduling failure at scale, manual workaround that could not absorb the volume had to be used, there was no saturation scenario in rehearsal to test capacity and delay.
Lesson: Wargaming should simulate the saturation of manual workarounds, not just the initial switchover.

The Southwest Airlines meltdown demonstrated that a DMOP functioning at normal scale can collapse when disruption exceeds the manual workaround

[27] UK Government Digital Service (2024). Operational Resilience and Continuous Improvement — GOV.UK Pay Case Notes. London: Cabinet Office.

[28] British Airways (2017). IT Systems Failure — 27 May 2017. London: International Airlines Group.

capacity. A DMOP needs to be designed as a Scalable System of Survival. Rehearsal should include the saturation scenario.

Recovery timelines may extend across multiple days. A power failure, for example, can be sustained on generator supply, but then generator fuel capacity needs to be considered within the DMOP plan, including resupply arrangements and fuel supplier dependencies.

5.5.5 Communication Exercises for DMOP

Communication templates are necessary but insufficient for DMOP exercises. During wargames, facilitators should inject realistic external pressure: a journalist tweet, a board member call, a regulator inquiry. The M&S April 2025 incident demonstrated the consequence: employees resorted to WhatsApp on personal devices when official channels were disrupted.

Advice: DMOP Governance: *Every IBS should have a current, rehearsed DMOP. DMOPs should be offline-accessible, assigned to a named role, and reviewed following any material change to the IBS architecture or impact tolerance. This is acceptable for most sectors under BS22301. Significant public services or suppliers identified under the UK Cyber Security and Resilience Bill not having a DMOP would be difficult to defend. For Financial Services this is more stringent: under DORA inclusion of a DMOP in the annual Digital Recovery exercise is mandatory.*

5.5.6 Use of AI in DMOP

> AI in Practice: AI in Degraded Mode Operations
>
> AI-assisted DMOP activation monitoring: AI systems monitoring live telemetry can detect when trigger thresholds for DMOP activation are being approached, providing advance warning to operators rather than a binary threshold breach. This allows pre-emptive preparation rather than reactive invocation.
>
> Resilient AI components: Where AI models or LLMs are embedded within an IBS, the DMOP needs to include a non-AI fallback path. AI model drift, where a model's outputs degrade without visible error can constitute an invisible failure within a critical service. The DMOP should specify the trigger for switching to the non-AI fallback and the procedure for doing so.

5.6 What the Board should expect to see

The questions posed at the beginning of this chapter can be answered by a summary report that will rely on further deliverables produced by the service delivery team (and listed below)

- Have we declared our Important Business Services (IBS) and their Impact Tolerances — and are these reviewed at least annually?
- Can we demonstrate — not merely assert — that recovery capability has been tested within the last 12 months?
- Do our third-party contracts include Software Bill of Materials (SBOM) disclosure, staged rollout requirements, and rollback obligations? (contracting detail: Chapter 3)
- Are our Degraded Mode Operating Procedures (DMOP) current, offline-accessible, and rehearsed?
- Does our Board have a designated member accountable for operational resilience?
- Do we treat AI that calls external services or acts autonomously as a board-level resilience and supply-chain risk, and have we assured ourselves it is governed accordingly?

Depending on the regulatory environment of the organisation, it may also be necessary to produce the following

- (Financial Services) DORA Register of Information: complete structured inventory of all ICT third-party service arrangements, maintained continuously for supervisory submission (see Chapter 3.3)

The summary report will rely on the following deliverables produced by the service deliver team:

- SBOM Registry: current software bill of materials for all production systems, updated on each deployment and reviewed quarterly (for register ownership see Chapter 3.5; consumed at every deployment gate in this chapter)
- ART Test Results: documented results of Automated Recovery Testing exercises against each IBS, reported to the Board
- DMOP Set: a current, rehearsed, offline-accessible DMOP for each IBS, assigned to a named role and reviewed after every material change
- Data Validation Standards: per-IBS specification of validation rules as acceptance testing criteria
- Deployment Runbooks: documented staged deployment procedures for each IBS-supporting system, including automated rollback steps

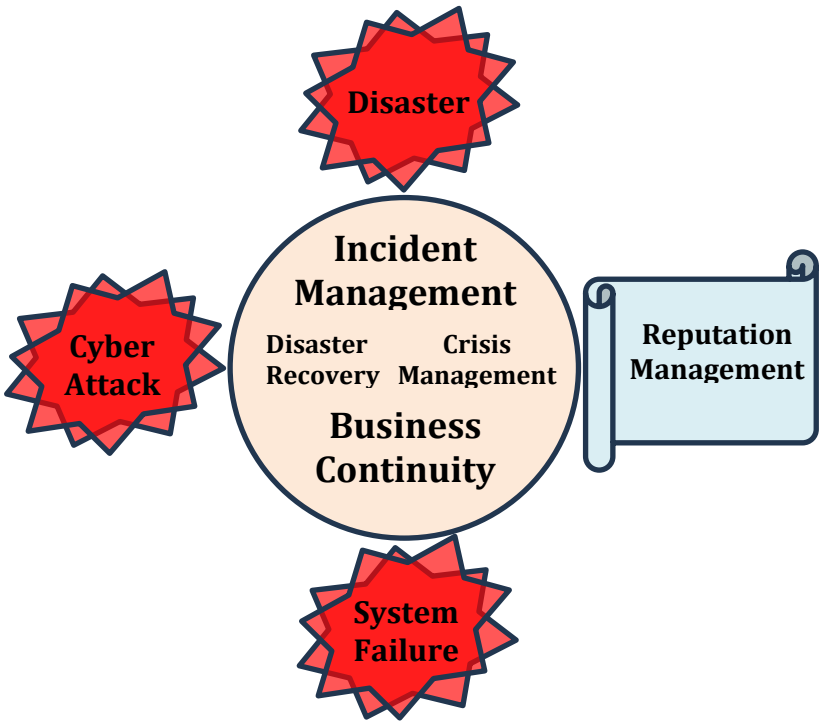
- SRE Resilience Scorecards: recurring measurement of availability KPIs, near-miss trends, and RTO/RPO test results reported to the Board quarterly
- CSRB Incident Reporting Readiness Pack: pre-prepared templates and tooling for 24-hour initial and 72-hour detailed incident reports
- Resilience Scorecard (Enhanced): Impact Tolerance Headroom and Recovery Confidence Scores reported quarterly to the Board

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

6.1 Purpose of this chapter

Systems will break. The resulting service outages are referred to internally as incidents. This chapter is about minimising the impact of service outages by restoring the business service. It covers business continuity, disaster recovery, incident and crisis management, and emphasises the importance of post incident retro analysis.

Not all incidents have a (potential) large impact on the customers and/or the users. Those affecting Important Business Services or over the threshold set by the regulator for lost user hours or harm to users clearly take priority. These are handled using the major incident management process.



The approach taken in this Chapter is influenced by the architecture and characteristics of the overall system envisaged. For instance, a system which only supplies a real-time picture of something is best served by a "just reboot everything" approach. Whereas a system holding historical information (or customer data) needs to be recovered properly to avoid loss of data. Further, the processes defined here must be balanced against legal, regulatory and business requirements.

In large organisations there will be several functions (silos) with teams dealing with different aspects of response and recovery from incidents. In smaller

organisations several functions may be assigned to a single person. This chapter covers the *functions* of Business Continuity Manager, Risk Manager, (major) Incident Manager, Crisis Manager, Disaster Recovery Manager, as they collaborate to restore services and minimise damage to reputation.

6.1.1 Key messages

Resilience is an organisational discipline, before it is a technical one. Systems will fail, and how the organisation absorbs the failure depends on governance, culture, skills, and the human responses prepared in advance as well as technical factors.

This chapter covers

- Detection and classification of IT related failures
- Incident Management
- Major Incident Management
- Post Analysis and reporting

Throughout, the chapter aims to prepare for the day the unthinkable happens.

For CEOs and Non-Executive Directors

Strategic governance questions for board-level accountability:

- Can we restore our Important Business Services (IBS) within agreed Impact Tolerances when, not if, an incident occurs, and have we proven it within the last 12 months?
- Do we hold a single authoritative inventory of all applications, with IBS clearly flagged, to give the Board oversight during an outage? (See Chapter 4)
- Is decision authority for major incidents pre-assigned, including who may invoke a “kill switch” (often managed as part of service delivery initially see Chapter 5), and under what circumstances so recovery is not delayed waiting for sign-off?
- Do we understand our duty-of-care and regulatory reporting obligations (FCA, NIS thresholds) for customer-impacting outages?
- Are we balancing investment in recovery capability against security spend, recognising that the ability to recover quickly is essential because no security is perfect?
- Does the Board receive post-incident reports for material incidents, with tracked action items and named owners?

6.1.2 Definitions

We use the definitions:

- Incident management is a process used to respond to and address unplanned events that can affect service quality or service operations[1].
- Incidents can in general be foreseen and if detected and managed will not normally turn into a crisis. IT incidents can propagate or cascade if not anticipated – see Chapter 5.
- Disasters are external sources of disruption. The sources have been historically events such as floods and accidents; increasingly, they involve failures of IT infrastructure. An IT disaster recovery plan is a portfolio of policies, tools, and processes used to recover or continue operations of critical IT infrastructure, software, and systems after a natural or human-made disaster[2].
- A crisis disrupts the whole organisation and causes reputational damage. Crisis management is the process organizations use to prepare for, respond to, and recover from disruptive and unexpected events (like data breaches, supply chain disruptions, or natural disasters) that threaten to harm the organization, its people, operations, reputation, assets, or stakeholders[3].
- Major Incident Management refers to the team and processes within the broader Incident Management framework, designed to handle high-impact disruptions to IT services that can severely affect business operations. They have close links to Disaster Recovery and Crisis Management.
- Business continuity ensures organizations can maintain their critical business functions during - and after - an incident has occurred[4].

[1] https://www.ibm.com/think/topics/incident-management

[2] https://www.ibm.com/think/topics/disaster-recovery

[3] https://www.pwc.com/ua/en/services/forensic/crisis-management.html

[4] https://www.thebci.org/thought-leadership/what-is-business-continuity.html

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

6.2 Detection and classification

6.2.1 Incident Detection

In Chapter 5 we discussed AIOps systems which provide monitoring of system performance and analysis of the resulting log files, as well as taking actions to reduce the number and impact of outages.

Most systems operated by organisations are distributed, in which case the definition of outages is more complex than simple service loss. A hard outage is the traditional definition of downtime: the service is entirely unreachable due to network partitions, DNS routing failures, collapsed infrastructure, or a catastrophic software crash. While this is the most disruptive in the short term, it is the easiest to detect. Health checks fail immediately and automated failover mechanisms can be engaged (see Chapter 5).

More complex cases include:

- High Latency: "Slow is Down"
  High latency occurs when a system successfully processes a request, but takes significantly longer than the baseline to do so. In an interconnected architecture, latency is highly contagious. Delays in internal service response can trigger timeouts, causing automated retry mechanisms and a self-inflicted Denial of Service to users.
- Elevated Error Rates: The Degraded State
  An elevated error rate means the service is reachable and responding quickly, but some responses are failures. This can be due to partial infrastructure failures, such as a faulty node, or when an application cannot reach an essential service like a database. The user experience becomes erratic and unpredictable. When processes are deviating from the usual activity this may be an indication of the beginning of an incident that would necessitate recovery.
- Corrupted or Untrustworthy Results: The Silent Killer
  The service is reachable, responds quickly, and returns a success code—but the data is wrong. It could be that the payload is stale, incorrect, or corrupted. This can happen due to database corruption, split-brain network scenarios, or logic bugs introduced during deployment. The system appears healthy to standard monitoring tools, the corrupted data can silently propagate, polluting downstream systems. Recovering from untrustworthy results requires complex forensic analysis, data purging, and state reconstruction, leading to massive, hidden operational costs.

6.2.2 Monitoring, alerts and recovery

With these caveats in mind – for instance that data on response times may not be a clear signal, that data corruption may not be visible - monitoring of status, performance and "service" should ideally be applied to every component of the system and in layers up to the full end to end system.

- Status and performance and capacity of individual components
- Status of any backup, standby systems
- Relative capacity of load shared components
- Service monitoring of components (eg how fast an API is performing)
- User perceived service monitoring (both status and performance)

This structure supports detection of more complex outages within the system.

For monitoring of outages in the supply chain, there are commercial services such as DownDetector[5] which reports signals from affected users.

Some systems (e.g. Dynatrace[6]) provide real time monitoring that implements a sequence:

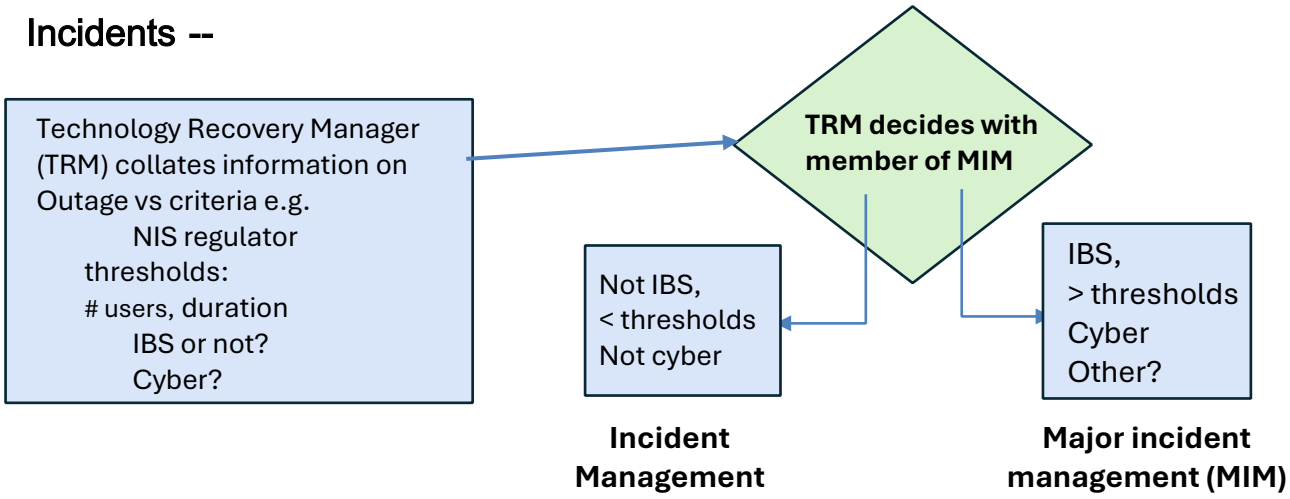
- Autonomous (received) events
- Proactive Monitoring (status, performance, capacity events)
- Event filtering
- Alerting (to support teams)
- Automated diagnostics
- Root cause analysis
- Automated recovery
- Advice and support for manual recovery

Real time alerts and recovery
In addition to error logs, ongoing active real time alerts can assist in determining the recovery route. Monitoring of traffic, after identification of where the issue may lie, often helps address the questions:

- When did the issue first begin? Often the incident is due to a prior buildup of the issue so estimating or identifying the start time is key to establishing a cause
- When did the system last run cleanly?

[5] https://downdetector.co.uk/
[6] https://www.dynatrace.com

6.2.3 Classification by impact of the incident



Examining the criteria set up by the regulator under NIS and/or the identification of IBS, a Technology Recovery Manager (TRM) from the RUN function together with a member of Major Incident Management (MIM) allocates the incident for resolution.

The configuration map as described in Chapter 4 is a key part of all remedial action.

6.2.4 Record keeping and analysis

Data Logs
We discussed the importance of system data logs in Chapter 5. Capturing errors with full documentation is a requirement for speedy recovery. The architecture must make provision for these error logs and glossaries of interpreted error codes, see Chapter 5.3.6.

6.2.5 Comparison of Incident and Major Incident[7]

The correct characterisation of an incident is important as major incidents need to be treated differently from incidents. The Table compares the two. However, in the heat of the moment, identification may not be easy.

[7] https://kepner-tregoe.com/blogs/incident-management-vs-major-incident-management/#:~:text=A%20standard%20incident%20usually%20affects,the%20business%20as%20a%20whole.

Checklist: major incidents and incidents

Impact and frequency

- A standard incident usually affects only a few users, allowing for response and resolution times that are typically longer to help keep operational costs low.
- Major incidents, on the other hand, are rare and have significant repercussions for the business as a whole.

Skills and roles involved

- Service desk personnel with limited training and technical expertise handle most incidents. Complex issues are escalated to second- or third-tier support teams with more specialized knowledge.
- Major incidents focus on engaging the individuals who can resolve the disruption the fastest, thus minimizing extended business impact. Typically, these resources are highly skilled (and correspondingly high-cost) subject matter experts.

Processes

- Recent years have seen a shift in incident management processes toward self-service, automation and asynchronous support interactions (e.g. email-based interactions with global call centre teams).
- Major incident processes must be optimized in the opposite direction, prioritizing solution effectiveness and speed of resolution over resource cost and automation.

Communication

- In standard incident scenarios, there may be no need for communication of the incident or its resolution.
- Major incidents are different in that active and broad communication with stakeholders is essential for managing expectations and assuring stakeholders that the situation is under control. In many major incidents, the perception created by communication plays a more significant role in shaping the overall impact than the technical problem and its associated symptoms. Effective communication during a major incident needs to address four distinct groups of stakeholders:
  - The affected user community whose activities are directly impacted by the incident
  - Stakeholders who are either indirectly, or potentially affected, whose trust is crucial to managing the incident
  - Internal teams and subject matter experts involved in diagnosing and resolving incidents (this may include vendors)
  - Support and IT Management

6.3 Incident Management

Incident Management is of course part of Service Delivery, which is the topic of Chapter 5. It is covered here because of the similarities to and links to Major Incident Management, and the extra focus we provide in this chapter on the learnings from the post-incident review.

6.3.1 Team

RUN group[8]
The "run" (RUN) group performs all the operations and provides support to keep the technology environment operating and meeting service-level expectations. It usually includes:

- a service desk to receive user incidents and requests and resolve as many of them as possible; those that the service desk can't resolve are passed on to the appropriate service group
- a level-one group with staff trained on many technologies to address simple tickets for most technology domains
- level-two and level-three groups to address more complex tickets that require some degree of technical specialization and to oversee the operational health of each technology area
- a field-service organization to resolve all tickets that require "physical touch" of an asset in a business location (for example, installing a new local-area network or changing a power supply on a local server)
- a facility organization to oversee and operate the data centres and resolve all tickets that require physical touch inside a data centre"

Key role - Technical Recovery Manager (TRM)
This person is often the first point of call on an incident. The TRM's main role is to lead a team of SMEs (subject matter experts) and individuals who maintain the system daily. A TRM must know when changes in core services have been agreed and any changes in recovery requirements. TRM's will often have responsibility for specific services or business areas.

They are responsible for maintaining

- up to date application recovery documents
- recovery playbooks which detail historic incidents and how recovery from these was achieved.

[8] https://www.quora.com/What-does-a-run-team-do-in-the-technology-department-of-companies

Key role - Subject Matter Experts (SME)
They have expertise in the software used to create the application, infrastructure or system and can assist in recovering an asset that has failed or is about to fail. Job titles include Software Engineers, DevOps, SREs, architects, Test/Quality Engineer (QE). Users of the system may also be included.

On call rotas
For key roles there is a rota system in place. Most large organisations have 24 by 7 staff who follow the sun so can monitor round the clock during the normal working day. Rotas should take account of holidays and have dedicated phone numbers.

6.3.2 Processes and Systems for managing an Incident

Assigning Incident support number: the RUN or daily maintenance support will review the alert and assign a support number (typically by using a ticket assignment subsystem) and make a log entry to document the initial investigation. They will update the incident log with details and the steps taken to resolve the incident.

Incident Assessment: A member of the RUN group will assess the priority of the incident and determine if further action is required. An experienced TRM will be able to assess if there's a need to include a MIM team member or a TRM from a different RUN Group, if the system or service is an IBS, and if the actual or potential number of lost user hours exceeds the regulator's guidelines.

The Exceptional Case: Need to Press the 'Kill Switch': In some cases, it is necessary to shut down system operations to limit the level of the damage. The existence of a 'kill switch' mentioned in Chapter 5 as one of the possible actions to be taken during monitoring becomes the responsibility of incident management if it is invoked.

Example: Knight Capital Group
Knight was a leading U.S. global financial services firm that dominated equity market making and electronic trading, handling at its peak over 3.3 billion trades and more than $21 billion in value daily. On 1st August 2013, an error in the deployment of a software update to its high-speed electronic trading systems caused the systems to create erroneous stock market trades, When the operations staff realised the problem, they had no way to stop the systems, which created four million trades in 45 minutes, leading to the loss of nearly half a billion dollars and the company's bankruptcy [9].

Incident capturing system

Real time monitoring may cause an alert to be triggered to the incident management system which holds all incidents; this will assist in the classification of the incident or may in some cases initiate automated recovery processes.

Checklist: classifying an outage

- Is it a full outage or partial?
- What business process is being impacted?
- How does that impact our customers/users?
- Is there a monetary impact?
- Is there a specific time for recovery required e.g. deadline for close of stock market for which trades required to go through to complete end of day processing?
- Can end of day processing of outstanding data be completed if the issue happens late in the day and there's still a backlog? – This implies a need to quickly assess the backlog if there's been a build-up, if reporting hasn't been carried out quickly enough.
- Is there any FCA (Financial Conduct Authority) or other regulatory reporting required?

Severity ratings

It is useful to have a classification system for the severity of incidents. Examples include Low/Medium/High/Critical or Bronze/Silver/Gold. In general, the higher the level the more severe the impact and the highest level might be used to designate incidents with major impact. These ratings for internal use may correlate with IBS or regulator set thresholds.

Recovery Playbooks

A recovery playbook documents past incidents and how recovery was achieved. The playbook will document links to IBS and indication of the customer impact to be expected from similar future incidents including what customer communication was undertaken. Contacts that were used in recovery and suggestions ideas about desirable code or system changes are also desirable features of playbooks. Recovery playbooks should be produced from a periodic review of logs from 'low level' incidents and invariably should be produced as part of the Retro process for all major incidents.

6.3.3 Communication channels

Recovery from an Incident may involve multiple activities. Everyone involved needs to understand their role and who are the key stakeholders. This is facilitated by co-location of involved staff.

Alternative agreed communications channels are essential: e.g. Teams, Google etc. Individuals need to know how to cascade information or information requests to other relevant personnel. This is a key consideration for the wider senior management team if the ordinary communication systems are not available.

6.3 Major Incident Management

6.3.4 Major Incident Management

Example: Major Incident

A contractor doing maintenance work at a British Airways data center inadvertently switched off the power supply, knocking out the airline's computer systems. Upon reconnection, an uncontrolled server recovery cascaded through the facility. Approximately 75,000 passengers were stranded. The estimated cost exceeded £80 million.

Key failures: No DMOP prescribing controlled circuit-by-circuit restoration; no mandatory wait-and-check gates; unavailability of trained staff for complex recovery.

Lesson: Power restoration is a degraded mode operation requiring a prescribed, rehearsed, gate-controlled procedure.[9]

Stages of a Major Incident

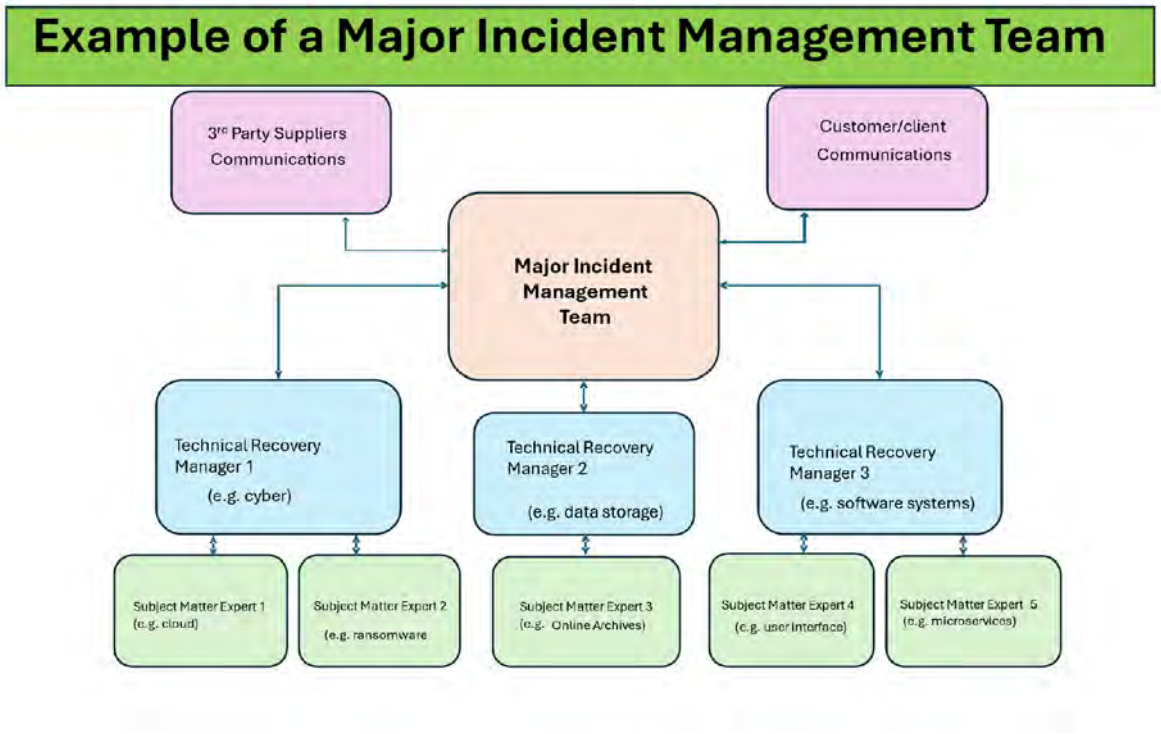
Major incidents are treated in 4 main stages within an organisation, namely:

- Identification - we discussed the identification of a major incident above, and the role of the MIM Team.
- Containment - we discussed methods for containment in Chapter 5.
- Resolution - The MIM Team are responsible for this. Since it is to be hoped that Major Incidents are rare, it is essential to have in place
    - People with clear roles. A key feature of the roles is assignment of authority to take decisions or know who is empowered to take key decisions within the desired 'time to recover.'
    - These individuals rely on systems that are accessible and hold details for recovery steps and all aspects of the system, application or infrastructure that is failing or has failed.
    - The authority to take decisions without senior management signoff and to take a lead role in co-ordination between multiple business areas to ensure that recovery happens swiftly.
    - Organisations will differ in how overall responsibilities for coordination, redirection and guidance are distributed. It is important, however, to have clear delineation of these responsibilities.

[9] https://www.forrester.com/blogs/17-06-15-lessons_learned_from_the_recent_british_airways_outage/

- Learning[10] - Most outages will not require or benefit from extensive analysis and problem solving. In all cases, analysis and problem solving will be involved in the aftermath of an outage to identify root causes and make necessary changes to architecture or processes, see section 6.5.3.

Key role - Major incident management (MIM) team



This team has close relationships with

- Technical Recovery Managers (as above)
- Suppliers
- Customer Service and Marketing
- Programme Managers and Enterprise Architects
- Site Reliability Engineers

This team operates a control-room-like structure and, in preparation for major outages, they run simulations at regular intervals to ensure that the team stays abreast of key trends and that the processes are still appropriate.

6.4.2 Incident diagnosis

Diagnosis of incidents requires:

[10] https://www.manageengine.com/products/service-desk/it-incident-management/major-incident-management.html

- Access to systems to instigate investigation, and access to call outs for support, are a key factor where speed of recovery is essential.
- Ability to gather the correct human resources and keep everyone focussed on the issue at hand. At every step, ask for updates from teams, ensure each person's role is clear and ensure that the right person has been included and will be available. For example, is the person who interacts with 3rd parties (such as the supplier manager) available outside of working hours?

- Access to the system to gather data related to failure: the types of data which can help are in the Checklist. Authorisation to access these may be a hurdle in practice.

Checklist – data to inform diagnosis

- Accurate and up to date map of system interdependencies (see Chapters 3.4.3, 3.4.4 and 4.2.2)
- Real time monitoring of how the system normally runs with changes in traffic flows (See Chapter 5.4.1)
- Error logs (see Chapter 5.4.1)
- Data on changes and new implementations that have taken place (see Chapter 5.4.1)
- Records of previous incident recovery (see section 6.5.3)

To reduce the time taken in the initial diagnosis stage, AI will increasingly be able to correlate the current information describing the current issue and search through previous incident records to identify possible causes and propose remediation steps.

Cyber security

The Cyber Security team will be able to review initial incident data in case the issue is on a known list of cyber-attacks. If it is, they will lead the recovery and restoration or be able to eliminate the threat. Decisions for cyber security incidents on restarts or decision to shut down any system or hardware device should be on the advice of the security team.

They will have established readiness processes to

- protect the organisation's reputation,
- know the best response to criminal demands for payments,
- engage the appropriate legal advice
- reporting to ensure visibility to the boardroom.

6.4.3 Culture and diagnosis

The team needs to operate within a blameless ethos where recovery is the foremost concern. This requires a culture of respect and understanding in recovery even though it is often a stressful situation, even if procedures are well-rehearsed.

The team must be able to openly discuss any shortcomings or gaps in current support or knowledge. It is the responsibility of the MIM to ensure that team members and those they rely on feel they are in a safe space. This allows those involved to highlight any sensitive issues that may assist with recovery. Keeping all the key members in communication while discouraging side conversations can prevent information being hidden from the wider group.

When a potential root cause is identified this should be accepted as valuable information (rather than a start at assigning blame). The aim is to understand how the situation came about and how to support remediation. The norm should be an acceptance that no question or line of investigation should be excluded. Allowing evidence to lead the areas for investigation will ensure that all routes to resolution are covered.

When discussing issues, it should be the task at hand and not team dynamics or interpersonal issues that is the focus of the discussion. Depersonalisation of any comments or issues will foster a more effective recovery team.

6.4.4 Mitigating the Impact of the Incident

Workarounds and partial resumption may be introduced before full recovery of the service is possible.

Should the solution require a roll back of recently deployed change, or introduction of a workaround, this may take place during the working day or, if this would cause wider impact, then it may have to wait until a more suitable time. This may need agreement with the organisation's key stakeholders so that they are aware that changes will take place. And if the roll back discontinues a new feature, reminders will be needed concerning the use of the old process.

Example: MVP after failure of payments system
When the shipping giant Maersk was hit by the NotPetya ransomware virus in 2017, almost all their software systems including those on-board ships were destroyed. But in accordance with normal procedures, most ships in transit had printed out hard copy documents of the customs and harbour documentation they needed for their next port. So, those ships could continue

their current voyages until they were safely docked in port. Other shipping continued with manually prepared paperwork taped to containers.

Checklist: Workarounds

- When there is a swift identification of a complete outage and a recovery strategy is not immediately identified, it may be desirable to implement the use of an MVP. This will be a reduced version of a key system that can carry out core activities in the event of failure of the main system (see 5.5).
- Ensuring there are means for human override where this could make sense in an emergency.
- Core System Only Mode: Ability to disconnect housekeeping systems, such as financial recording and auditing, where they are stopping vital system delivery (see 5.5).

6.4.5 Recovery after an Incident

Role of a diagnosis of the cause of the outage
Recovery will often need to be initiated before a root cause is determined (see 6.6.2).

Checklist: frequent immediate diagnoses

- Hardware or device failure
- Legacy software malfunction.
- Operating system or 3rd pert software or service error
- Failed updates/changes
- Malware or ransomware infections
- User or operator error
- Data verification error
- Abnormal loads.

In each case, if the incident resembles a previous outage, a Recovery playbook can provide the resolution steps and actions to be carried out. In some cases, this may be automated, e.g. for server or service restarts. Intermittent issues can benefit from QA support to see if the issue can be replicated, to get a better understanding of the fix required. This is particularly relevant for code upgrades or migrations where the issue shows up mainly in the Production environment. Differences in system features (e.g. firewalls) in the Pre-Production environment may differ from Production.

A Playbook also includes:

- understanding business impact and what needs to be provided for staff who may need to interact with customers
- what comms needs to be published for external customers – e.g. pre-prepared holding pages and/or pre-formatted websites for entering update information.

If the Playbook does not provide a recovery actions, you may need

- remote access to systems,
- use of admin passwords,
- contact details for 3rd party technical experts (eg SaaS components)
- contact details for 3rd party support teams.
- Access controls for corporate IT systems eg for ID authentication and multi-factor authentication (MFA).

All of these may run up against hurdles if the organisation has not practiced recovery. In Chapter 3 we defined the use of SLAs and the need to ensure explicit 3rd Party support provisions. Without this, accessing support in case of an outage can be a source of substantial delay.

Advice: Practicing recovery
Running practice simulations for major recoveries at regular intervals is one of the most effective ways to identify areas for improvement and for gap analysis. This ensures that major incidents have been considered and the team's ability to work effectively is aligned to improvements in incident response and recovery strategy.

To ensure systems can be restored from scratch you will need to practice doing so, regularly. Exercise of backup recovery systems could be:

- *Trial restores, in which a separate system is restored from backup to an isolated recovery environment.*
- *Trials of 'warm standby' systems where the backup is available on a running machine and can be taken live.*
- *Including database restores into both the software developers' automated testing and the Quality Assurance team's validation of the software.*

It is recommended that Disaster Recovery exercises (DR) be practiced at least annually. It is advisable to undertaking a DR for any new components such as servers, connectivity to a new end point/URL or a significant increase in traffic.

The existence of a full backup of the system data is not enough to prove that restoration will be possible. Potential issues include

- configuration data missing;
- new hardware incompatible with backups of the software;
- backups themselves corrupted.

Example: British Library
The British Library is the United Kingdom's main central library, retaining a copy of every book published in the UK and much else. Its computer software systems were destroyed by a targeted ransomware attack in October 2023.

Despite an effective crisis management plan and intact offline backups, two years later many of its external-facing services were still unavailable. The library's outdated legacy systems could not be reinstated as before, and the backups proved incompatible with new, more secure, systems.

The British Library's management concluded that "Given that no security is perfect, the ability to quickly recover is essential when (not if) an attack is successful. Investment in security needs to be balanced against investment in back-up and recovery capabilities."

Advice: Rectifying data after recovery
Sometimes after the systems are again operational there may be areas that require attention, such as rectifying data that is corrupted or erroneous. This needs eg:

- *Access to write-only backups: Incorruptible archives and backups, preventing attackers from destroying past backups and preventing restoration.*
- *System allows operators to enter 'back dated' transactions after a period of manual operation.*

Internal communications
Regular updates of the incident log are particularly important for prolonged outages to give visibility to processes undertaken and their results. A preferred layout for providing timeline updates to those focusing on the recovery and the sequence of events includes the specific time that an action or activity has been decided.

Example… Timeline updates among the team
Incident Update <Date>
Incident Reference: INCXXXXXXX
System Impacted: XXXXXX
Start Time: 09.32
Recovery Time: 10:42

09.32 – Alert was provided indicating that there was an issue in system x
09.37 – Notification provided from the support team that they were investigating the alert
09.45 – Support team contacted TRM to confirm that the alert required recovery as per the Recovery Playbook
09.51 – TRM set up a comms channel to notify stakeholders of the incident and that recovery was underway
09.56 – TRM received confirmation of potential business impact as call increase is being reported…

External communication and Crisis management
Users are often updated via messages at the point that they are using the service, and customer help desks need to be fully informed of the likely duration.

The business team affected by the outage will focus on press/social media by monitoring customer comments and responding as appropriate to manage expectations for recovery or offering support alternatives.

A major incident can be declared a crisis: then the Crisis Management Team take over the responsibility of managing the situation with stakeholders, while the leader of the recovery team will still be tasked with actual recovery.

6.4.6 Incident Closure

Checklist: Closure of the incident should include the recording of:

- Business impact – including total staff or system impacted, cost or value of impact
- Outage time – this should include the start time and the total time that the system was unavailable
- Actual time of recovery – this is sometimes to the minute which helps determine if the recovery is within SLA
- Opening of a 'Retro' process to capture the suspicions, learning, and findings from recovery process (diagnosis, repair determination, and recovery implementation)

- A Recovery playbook entry should be created for all incidents in systems and services important to the organisation.

The incident ticket can be closed with an initial indication of the cause if the failure was introduced by a change activity that can also be noted. Often there are multiple incidents all relating to the same cause: if a lead or parent incident can be identified all related incidents are then closed.

6.4.7 The role of Business Continuity

In general, Business Continuity will not be concerned with the incidents that can be dealt with, without unacceptably disrupting critical business services. Their role is at the intersections with Major Incident Management.

The four pillars of a Business Continuity Program are[11]

- Assessment including hazard identification and risk evaluation. The risks increasingly include IT related topics. The business impact analysis and related recovery time and point objectives for an IBS may be delivered by the BC Team.
- Preparedness involves ongoing training and activities that simulate the events of crisis and the actions needed to recover. This often includes oversight of data backup strategy for use in recovery.
- The goals of response teams and how they are created, the emergency operations centre (EOC), and response procedures. This includes oversight of when Degraded mode operation is feasible (see Chapter 5) and the ability to resume critical business functions once the IT systems are operational.
- In order to recover from a crisis, the business should keep good records of the business response and damage sustained, as well as come up with a corrective action plan to mitigate the effects of a similar crisis in the future. The report that does this is discussed under Post Incident Retro, below.

In circumstances when it is certain or probable that the recovery will not be completed within the agreed timelines, it is important that Business Continuity is immediately informed so that they can activate the necessary Business Continuity Plans.

[11] https://www.sciencedirect.com/science/chapter/edited-volume/abs/pii/B9780124116481000039#:~:text=The%20four%20pillars%20of%20a%20BCP%20are%20assessment%2C%20preparedness%2C%20response,hazard%20identification%20and%20risk%20evaluation

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

6.5 Post Incident Retro

6.5.1 Why this is important

This process should ensure that preventative measures can be put in place either to remediate the cause permanently or to provide a workaround until a permanent fix can be deployed.

What is needed

- Build the knowledge base: at the moment this is mainly done through personal experience of the support SMEs.
- Identify event patterns and their pointers to the root problem. AI can quickly learn how to recognise problems and what best actions to take.
- Link these to immediate actions that can be taken manually or automatically. This process needs to include all key stakeholders

Use of the Retro

The focus of the Retro process is on ensuring that the discovery of the root cause is obtained in a blameless atmosphere. Openness is key to flushing out all tasks to ensure that the incident doesn't occur again. The process leads to a report which focuses on actions. And, if it is established that there will not be a permanent fix for some time, then awareness of the risk of this re-occurring is communicated and recorded in a Playbook.

The different audiences may require subtly different information from the Retro output. Section 6.5.3 below covers the internal and external (eg regulatory) requirements.

6.5.2 Agenda – list of areas for capture

Speed of engagement

- Did the call out or support group response meet internal SLAs?
- Was the response appropriate given the initial information available?

Diagnosis and Remediation Proposal Process

- Were all initial stage assessments carried out in accordance with prior planning? What additional assessments were required?
- What alternatives were considered for remediation and the reasoning that led to the method chosen.
- How could these processes can be speeded up? eg including managing order and timing of team, stakeholder and 3rd party engagement.
- Were key stakeholders identified and included for comms?

- Was process followed to ensure all pre-checks for the system were carried out prior to a change being implemented?
- Were instructions clear and appropriate?
- Did the investigation follow logical steps?

Engagement with other teams

- Were there any delays or issues with engagement for teams that were required for support in the investigation or recovery?
- Had teams or SMEs that were required joined the recovery communications channel (e.g. MS Teams chat) within a given time frame?

Identifying the Root Cause (RC)
Once recovery is accomplished, the team should identify the root cause.

Checklist: techniques to identify the Root Cause

- Traditionally, 5 why questions are used to establish why the issue arose as opposed to what was carried out to fix the issue. This is an iterative interrogative technique used to explore the cause-and-effect relationships underlying a particular problem. The primary goal of the technique is to determine the root cause of a defect or problem by repeating the question "why?" five times, each time directing the current "why" to the answer of the previous "why".
- The Fishbone Analysis Process, also known as Ishikawa or cause-and-effect analysis, is a visual method used to identify and analyse the root causes of a problem or an effect. It is named after its creator, Kaoru Ishikawa, a Japanese quality control expert. The technique gets its name from the shape of the diagram, which resembles the skeleton of a fish, with the 'head' representing the problem or effect and the 'bones' branching off to indicate potential causes. As part of Dr Ishikawa's Fishbone analysis, the use of 4 P's (People, Place, Procedure, Policies) and the 4 S's (Surroundings, Suppliers, Systems, Skills) are often cited as useful as potential 'bones.'
- Fresh Look at RCA from the US Office of Personnel Management - provides some useful methods to take a fresh approach to identifying root cause.[12]

Were there any issues that contributed to the incident?

[12] See https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://www.opm.gov/policy-data-oversight/human-capital-management/closing-skills-gaps/root-cause-analysis.pdf&ved=2ahUKEwimq87p55OUAxWDWUEAHZLWLasQFnoECBkQAQ&usg=AOvVaw29NM-gxCgC96G7dD90N

Checklist: contributing factors
Factors often helpful in preventing the Root Cause from recurring:

- Technology – old infrastructure
- Design not fully understood
- 3rd Party issues
- Process not in place, not clear or no longer fit for purpose
- Documentation lacking and whether incomplete or out of date.

Larger organisations often have lists of predefined root causes to allow for collation of statistics on RCs. This enables identification of repeat patterns, and or other weaknesses to be addressed.

Impact Analysis

- Has the full impact has been identified and captured? This should include where possible actual costs to the business so that the cost of prolonged outage can be calculated against the cost of development or justification for updating or upgrading to a new service. Costing implications are discussed further in Chapter 2.

Escalation

- Were any requests for escalation were smooth and successful?
- If issues were encountered, how could team members help to avoid delays in future incidents involving escalation?

Monitoring & Alerting

- Is the current monitoring and alerting appropriate?
- Are updates or actions are required? Eg a specific check may be set up in the monitoring tool to alert if the specific issue is seen again; even better is a threshold that when crossed can prompt an alert to an increasing likelihood of failure. If highlighted early enough via a specific alert this can prevent actual failure from occurring.

Risk mitigation

- Has the Retro process captured any trends that will raise future risks or times when compliance might not be met, so that mitigation can be considered and planned for later implementation?

Improvements

Updates to documentation in a ‘recovery playbook’ should include support matrix contacts. Recovery playbooks often require review to consider the larger implications of the resolution: for example, desirable future code changes to refine what may have been implemented as a temporary fix.

6.5.3 Post-Incident Report: Content and Communication

The Post Incident Retro Process above involves learning and it produces knowledge. The Post-Incident Report (sometimes called a post-mortem, incident review, or root cause analysis document) is how that knowledge is preserved, shared, and converted into action.

A well-structured report serves three audiences simultaneously:

- the team that owns the system,
- other engineering teams who may face similar failure modes,
- customers, regulators and executives (where disclosure is appropriate) who need assurance that the organisation has a handle on recurrence.

A post-incident report should answer five questions in order: *what happened*, *who was affected*, *why did it happen*, *how did we respond*, and *what will we do about it*.

The sections of a Post-Incident Report below map to those questions. They have become a broadly accepted industry template.

#	Section	Purpose
1	Summary	In one to three paragraphs, explain the incident background, impact, root cause, mitigation procedure, and further action to prevent similar incidents.
2	Metrics and graphs	Quantitative evidence of impact — availability, error rates, latency, affected users, financial loss. Each chart should be labelled and timestamped.
3	Customer/user impact	Who was affected, how many, how long, and in what way. It is also equally important to mention explicitly what was *not* affected.
4	Incident response analysis	How the event was detected and how mitigation was achieved.
5	Post-incident analysis	Whether the failure mode was foreseeable, for example: existing risks, prior reviews, change-triggered causes, and whether the condition is programmatically auditable.
6	Timeline	Chronological events with timestamp and time zone, starting from the earliest contributing event that led to this incident.
7	Root-cause analysis (e.g. 5 Whys)	The causal chain continued until the cause is *actionable*.
8	Lessons learned	Numbered and succinct takeaways from this incident.

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

#	Section	Purpose
9	Action items	Tracked commitments with owners and due dates (see Action items — the "what we're doing about it").

Action items — the "what we're doing about it"
Action items are the only section of a post-incident report that changes the future. They deserve disproportionate attention.

Checklist: Questions that drive good actions:

1. If this happens again, could we detect it faster?
2. Could we reduce the blast radius — fewer users affected, smaller region, shorter window?
3. Could we minimise the customer-visible effect during the failure?
4. What contributing factors could recur, and how are we preventing them?
5. Where customers or downstream services have been the source of the issue could architectural changes be employed to avoid this class of event; is guidance for these changes documented and accessible?

Each action item is enhanced by the use of:

- A verb-led title ("Update the deployment runbook to...", "Add automated testing for...")
- A committed completion date (or a "Complete" status)
- A priority level

Suggested priority level definitions:

Priority	Meaning	Expected time frame
High	Addresses the root cause of customer-impacting issue, or an error that extended outage duration. Pre-empts other work; feature slips are acceptable.	Immediate
Medium	Operational or process improvement derived from lessons learned.	Within a couple of months
Low	Preventative maintenance, hardening for future scale, or best-practice alignment.	Within the year

Hygiene factors for the Report

- Discipline around dates matters. Aim for most completion dates within 90 days; longer horizons should be exceptions tied to a short-term mitigation already in place. Stretch-goal dates that predictably slip erode the credibility of the process more than honest longer estimates.

- Most reports settle into two to five action items. A list of fifteen is usually a signal that the items are written at the wrong level of abstraction, or that several reports have been conflated into one.
- Post-incident communication: Communication during and after an incident is a distinct discipline from the report itself, but the two are closely linked. A short set of principles applies:
    - Separate internal and external narratives, but keep them consistent. External messages necessarily omit detail; However, they must not contradict the internal record.
    - Distinguish "what we'll do" from "what customers might consider". External reports often include recommended customer actions — architectural best practices, monitoring, redundancy options. Frame these as *recommendations*, not as an implication that the customer caused their own impact.
    - Publish on a defined timeline. Many organisations commit to a service level for report publication — for example, within 14 days for high-severity incidents, or 24–48 hours for short, externally-visible events. Whatever the commitment, it should be consistent and tracked.

Checklist: before publishing a post-incident report

- Summary was written after the rest of the report
- Timeline begins at the earliest contributing event, not begins from the time of incident was detected
- Impact is quantified (users, duration, financial, regulatory)
- What was not impacted is explicitly stated
- 5 Whys has reached an actionable cause; "human error" is not the terminal answer
- Lessons learned are phrased so that another team can apply them
- Every action item has a date and priority
- Internal service names, code names, and personal data have been removed
- Recommended customer or downstream-team actions, if any, are framed as guidance rather than blame

6.6 The Role of AI in Response, Recovery and Learning

AI capabilities are evolving so rapidly that its actual applications lag behind those that can be imagined. We focus on applications that have been introduced and demonstrated some value and the capabilities needed to integrate these applications into operational practice:

6.6.1. Faster detection and triage

- AI-driven monitoring can analyse huge volumes of logs, metrics and traces in real time, spotting anomalies and incidents much earlier than threshold rules alone. New capabilities are needed to interpret this analysis and rule out common AI errors such as hallucinations and misplaced emphasis. The AI application may also need adjusting to reduce the number of false positive alerts.
- Incident-response assistants can automatically group related alerts, assess probable severity and impact, and route or escalate to the right teams with context included. Similar interpretive and cautionary capabilities are required.

6.6.2. Better diagnosis and decision support

- AI models can correlate current signals with past incidents and propose likely root causes, affected components, and blast radius, cutting time spent in initial diagnosis. A principal risk here is misidentification based on superficial correlation, requiring capabilities to critically evaluate AI findings.
- Generative AI can search runbooks, knowledge bases, and historical tickets to suggest concrete next steps, queries, or configuration checks tailored to the live incident. In this application, a particular AI implementation mimics other team members and will gain or lose trust depending on its effectiveness.
- AI platforms can capture the knowledge of people's roles and capabilities as well as functions, suppliers, etc. which may reduce the barriers to managing outages.

6.6.3 Automated and self-healing recovery

- Self-healing mechanisms use AI to trigger automated runbooks: restarting services, draining and replacing bad nodes, rolling back recent changes, or failing over to standby systems when confidence is high. Such mechanisms require thorough testing by team members familiar with possible AI errors.
- Event-driven orchestration with AI decision engines can choose between fully automated, semi-automated (with human approval), or advisory-only

actions based on risk and confidence thresholds. Similar interpretive and critical skills are needed as well as extensive testing to implement such AI subsystems.

6.6.4 Disaster recovery and continuity

- In larger disruptions, AI can manage backup and replication schedules, pick optimal recovery points, and orchestrate orderly failover of applications and data to alternate sites or clouds. Capabilities to determine the level of supervision are needed for this to be a safe application.
- During recovery, AI can dynamically prioritise which services to restore first based on business criticality and interdependencies, improving effective recovery time objective (RTO) for key services. Capabilities to determine the level of supervision are needed for this to be a safe application.

.

6.6.5 Learning to recover better next time

- After incidents, AI can mine post-incident data (chats, tickets, telemetry) to identify recurring failure patterns and weak runbooks, then suggest improvements to architecture and playbooks. These suggestions need to be critically assessed by team members knowledgeable about possible AI errors.
- Over time, this turns recovery into a learning loop, improving automated remediation coverage and reducing manual effort and downtime with each incident. This is of course an ideal that depends on the critical examination of performance in testing and operation.

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

6.7 What the Board should expect to see

Systems will fail; the incident is the moment resilience is proven or lost. This chapter is about minimising impact, detecting major failures early, classifying them correctly, recovering the business service, and learning so it does not recur.

The disciplines required are rapid detection, clear decision authority, rehearsed recovery playbooks, and blameless post-incident analysis to determine whether an outage stays an incident or becomes a problem or potential crisis.

The questions from the beginning of the chapter are now repeated, along with the key deliverables we have described, which provide assurance that they can be answered.

Can we restore our Important Business Services (IBS) within agreed Impact Tolerances when, not if, an incident occurs, and have we proven it within the last 12 months?

> Recovery Playbooks: per-system documented recovery steps and historic-incident resolutions, with automated restarts where it is safe.
> Incident & Major Incident Records: classified, time-stamped incident logs capturing business impact, outage duration, and actual recovery time.
> DR Exercise Schedule & Reports: at least annual DR tests (and for material new components), with gap analysis and remediation actions.

Do we hold a single authoritative inventory of all applications, with IBS clearly flagged, to give the Board oversight during an outage? (See Chapter 4)

> Produced as a deliverable in Chapter 4

Is decision authority for major incidents pre-assigned, including who may invoke a “kill switch” (often managed as part of service delivery initially see Chapter 5), and under what circumstances so recovery is not delayed waiting for sign-off?

> Pre-assigned roles for incident management and escalation of incidents from daily operations to incident management team.
> Kill-Switch Authority Matrix: pre-agreed terms of reference for who may disable key systems, when, and under whose authority.

Engineering resilience for IT systems:
Chapter 6 – Incidents – response, recovery and learning

Do we understand our duty-of-care and regulatory reporting obligations (FCA, NIS thresholds) for customer-impacting outages

> Root-Cause Register: agreed root causes (5 Whys / Fishbone) and contributing factors, collated to surface recurring patterns.

Are we balancing investment in recovery capability against security spend, recognising that the ability to recover quickly is essential because no security is perfect?

> Post-Incident Reports: the nine-section template (summary through action items) with tracked owners, priorities, and dates.

In addition, deliverables of immediate value for the recovery team include

> Communications Pack: pre-prepared holding pages, customer update templates, and defined out-of-band communication channels.
> Third-Party Recovery Schedule: provider availability, escalation paths, technical contacts, and SLA recovery obligations.

Engineering Resilience for IT Systems
Annex – AI Support for Enterprise Architecture

1. Purpose and scope

This Annex documents the use of AI in supporting the enhancement of Enterprise Architecture.

Enterprise Architecture (EA) links business objectives to technology decisions and helps ensure that resilience investment supports the organisation's outcomes and risk appetite.

The focus here is on the systems that underpin IBSs: services whose failure would materially impact customers, regulators, or the wider business. These services are typically assembled from many components and dependencies, so resilience must be engineered and evidenced across the end-to-end service chain.

2. Key Messages

Resilience is designed, not retrofitted. This Annex is addressed primarily to Enterprise Architects and Programme Managers: the people who decide, before any incident occurs, whether recovery will be possible, affordable, and fast enough. It sets out how to prioritise investment by business impact, how to build and maintain the architectural views that make failure visible, and how to select the patterns that contain damage and speed recovery.

- Failure is a planning assumption. Architect services so they fail in contained, recoverable ways and can be restored within agreed objectives.
- Prioritise by business impact, not technical complexity. IBSs and their impact tolerances determine where resilience investment is justified; not every service needs the same availability.
- You cannot recover what you have not mapped. Accurate, current configuration, dependency, and architectural views are the foundation of both anticipation and recovery; treat them as living artefacts under change control.
- Make objectives measurable, then prove them. RTO, RPO, and SLO targets, with error budgets, replace aspiration with evidence; a recovery capability that has not been validated through testing is an assumption, not a capability.
- Plan for the estate you have, not the one you wish you had. Inherited and legacy systems are the harder problem; apply lifecycle strategies, compensating controls, and explicit retirement decision points.
- Treat AI as both a planning tool and a planned component. AI strengthens scenario design and early warning, but AI embedded within services carries its own failure modes and needs the same governance as any other software (see Chapter 1, Section 1.7, and Chapter 4).

Additional governance questions include:

- Are the services capable of operating in degraded mode, and what are the implications of that degradation? (See Chapter 4, Section 4.5, on Degraded Mode Operating Procedures.)
- Are reliability and recovery objectives owned across IT development, IT operations, and business executives?
- How are resilience investment decisions made, and who determines the trade-off between availability and cost? (See Chapter 2, Section 2.4.)
- Confirm that redundancy, backup/restore capability, and recovery processes are implemented and tested to meet defined objectives.

3. AI to Support the Enterprise Architecture

AI can support the architect to:

- *Design to the non-functional requirements*, deliver guardrails around reliability, scalability, observability and security, enabling testing of the nonfunctional requirements at the design phase, and understanding the design choices
- *Design for failure* , using AI assistants to enforce resilience patterns (timeouts, retries with backoff, circuit breakers, bulkheads, graceful degradation) in design reviews, IaC, and code templates. In addition, AI can generate alternative "degraded" flows (smaller models, cached answers, read-only modes) so services can continue in a reduced but safe mode under stress
- *Design modular, observable architectures* supporting the design of logically separate data, model, and application layers with clear interfaces and standardised APIs, so faults can be isolated and contained. Additionally, AI can apply observability requirements, using AI to ensure that observability is embedded, enhancing the use of logs, traces, and metrics for anomaly detection and root-cause hints; designing the architecture so every dependency is traceable and measurable.
- *Design to governance, security and change resilience frameworks*; embedding AI-driven policy checks (least privilege, encryption, data residency) into CI/CD so every change is evaluated for its impact on resilience and cyber-posture, as well as using AI tooling to map services and data flows to business services and impact tolerances, ensuring architecture decisions support overall operational resilience objectives.

4. AI to Improve existing complex operational systems.

AI can improve resilience in existing complex operational systems by helping you understand what you really have, then to find the weak points, and so change the system in a safer, more incremental, data-driven way. This is dependent on the observability of the systems and the data available.

Few complex operational systems today exist in isolation. The dependencies may be cross organizational and/or extend into long and complex supply chains. These structures may limit the data available.

The contribution that AI can make can be categorized:

Making the invisible architecture visible

1. Code and dependency mining: AI can scan large legacy codebases, interfaces, and config to build maps of services, data flows, and dependencies, revealing hidden coupling and single points of failure you can't see from documentation alone. In addition, an expert system[1] guided by human experience can be used to set standards.
2. Hotspot and technical-debt detection: AI can highlight modules with high change/failure concentration, complex dependency fan-in/fan-out, and outdated technology, giving you a prioritized list of resilience risks to address first.
3. Learning from incidents and operations
    a. Incident analytics: NLP over tickets, incident logs, and post-mortems can surface recurring root causes, brittle integration points, and "near misses" that architecturally weaken resilience.
    b. Operational risk signals: ML on operational data (alerts, capacity, vendor performance) can identify leading indicators of disruption and inform which parts of the architecture need buffering, decoupling, or redundancy.

Scenario testing and complex-system simulation

1. AI-enhanced scenario planning: AI tools can generate "extreme but plausible" disruption scenarios (key provider down, data corruption, cascading job failures) and simulate business and technical impact.
2. Control and optimization of complex systems: For systems like grids, logistics, or trading platforms, AI control agents can propose parameter changes, routing strategies, or throttling policies that keep the system

[1] https://www.techtarget.com/searchenterpriseai/definition/expert-system

stable under stress. The sheer scale of the number of suppliers involved in these complex systems means that help from AI in keeping track of assets, versions, etc can generate alerts to prioritise interventions, see for instance [2].

Safer, incremental modernization for resilience

1. Refactoring and strangler patterns: Generative AI can help design and implement strangler-fig refactors around fragile legacy cores, suggesting service boundaries and target interfaces that reduce blast radius.
2. Automated tests and impact analysis: AI-generated tests plus impact analysis on dependency graphs allow you to change or replace components with higher confidence, reducing the risk that resilience improvements themselves cause outages.
3. Data Management and governance: AI-driven visualization tools for data mapping and analysis are in wide use, and there are many free tools[3]. AI can be used to support robust data governance, ensuring data accuracy, integrity, and availability.

Building an adaptive, learning system

1. Continuous resilience assessment: AI services can act as a "resilience advisor", continuously scanning telemetry, changes, and incidents to update a live risk view and recommend architectural or configuration adjustments.
2. Feedback into governance: Insights from AI analytics feed architecture and operational-resilience forums, so design standards, risk appetite, and modernization roadmaps evolve with how the system actually behaves in production.

[2] https://www.ey.com/en_us/insights/consulting/ai-enhances-it-asset-management-automation-and-security

[3] https://www.reddit.com/r/datavisualization/comments/1fmxgmg/what_is_the_best_free_ai_tool_for_data/

5. AI to Improve Operational Resilience

Areas where AI can provide support to reduce the number of operational failures,

Coding

Sometimes, legacy modules are incompletely documented and there is limited experience and expertise on those systems remaining within the organisation. AI can support the re-coding of the system, regenerating documentation, analysing the code, inputs and outputs, functionality and regenerating code. This is an area which is evolving rapidly[4] with huge focus across the industry on not just cyber security, but identification of code vulnerabilities and code quality.

Planning

AI supports planning for resilience in complex existing systems by giving you better foresight, richer simulations, and continuous evidence about where to invest change effort.

Stronger scenario planning

1. AI mines internal and external data (incidents, threats, providers, environment) to propose "extreme but plausible" disruption scenarios instead of a few manually imagined ones.
2. It helps quantify each scenario's likelihood and impact, so resilience planning can prioritise the combinations that actually matter (e.g. cyber + vendor outage + surge demand)

Deep simulation of complex behaviour

1. ML (see Glossary) enhanced system-dynamics or agent-based models let you simulate how a complex socio-technical system will behave under stress, including cascades and feedback loops you would not see analytically.
2. Synthetic data and "digital twins" of operations allow safe stress-tests of rare events, so you can test options (buffering, routing, throttling, failover) before touching production.

Continuous risk sensing and early warning

1. Predictive analytics over telemetry, incidents, vendors, and external signals can act as an early-warning system, surfacing patterns that historically precede material disruption.

[4] https://www.anthropic.com/glasswing

2. These models feed resilience planning with dynamic risk views, so your scenarios and playbooks update as the system and its environment evolve.

Prioritising resilience investments

1. AI can estimate how changes (extra redundancy, refactoring a hub system, new playbook) would shift key metrics like time to detect/respond/recover, helping you compare options and pick the highest-leverage interventions.
2. It supports portfolio-level planning by clustering systems and processes by criticality and fragility, guiding which parts of the architecture to harden first.

Tracking execution and demonstrating progress

1. AI tools can track remediation actions from exercises and reviews, send reminders, and maintain a live view of completion status against resilience roadmaps.
2. They generate resilience "scorecards" and trend metrics from repeated simulations and real incidents, evidencing to boards and regulators how the complex system's resilience is maturing over time.

6. AI in Testing

The use of AI to provide testing is becoming widespread and broad, covering many of the associated processes.

Quality Assurance (QA)

The big consultancies leverage AI to transform Quality Assurance for system resilience.

1. Deloitte deploys AI agents to autonomously generate executable test scripts[5]
2. McKinsey uses agentic AI to automate the testing lifecycle and eliminate technical debt.
3. PwC shifts to dynamic, intelligence-driven validation systems, allowing AI to interpret real-time system changes[6].

Inputs to the testing process.

1. Inputs into testing include specifications, metrics on software quality, feedback from customers, Customer Support, Project Managers, the Product Team and others.
2. AI can help analyse inputs; for example, tools such as ChatGPT[7] can be asked to find inconsistencies in specifications.

Automation of testing

1. Running automated tests in Continuous Integration on every code change creates confidence to release software changes.
2. AI can help teams automate tests eg Tools such as Cursor[8] and Copilot[9].

Growing Expertise

1. AI can be used as a "learning assistant" to help engineers resolve errors in automated tests, for example, an automation engineer can ask ChatGPT to explain a coding error.

---

[5] Deloitte QA

[6] https://www.deloitte.com/nz/en/services/consulting/perspectives/ai-assisted-software-engineering.html; https://www.mckinsey.com/capabilities/quantumblack/our-insights/ai-for-it-modernization-faster-cheaper-and-better; https://www.pwc.com/us/en/industries/health-industries/library/computer-system-validation.html

[7] https://www.chatgpt.com

[8] https://cursor.com/en

[9] https://copilot.microsoft.com

Risk based automation of testing.

1. AI can be employed to manage the balance between manual and automated regression testing. Prioritising automation for frequently changed systems can reduce change-related incidents.
2. AI agents can adjust test scripts based on previous results, focusing on high risk areas.

Managing Test Data and Environments

1. AI can be used to generate test data, for instance analysing historical user behaviour to test edge cases and hidden bugs.
2. It can mitigate bias in generated data.
3. It can also be used to document changes to existing datasets or data models, and oversee the integrity of existing data and processes with long running data accumulation, particularly in identifying anomalous events.
4. Specialised testing can use AI-driven visual testing tools to drive user experience testing. AI Bots can simulate real user interactions, uncovering interactions.

Managing the testing programme

1. AI-enabled platforms can automate repetitive tasks and provide insight into potential issues.
2. They can improve collaboration between QA and development teams as part of CI[10] Pipelines.

Feedback from the testing process

1. AI can be used to review the effectiveness of testing processes by using AI to analyse post-deployment metrics. The testing process provides feedback to engineers, Project Managers, the Product Team, Customer Support and others. The feedback can consist of bug reports, questions, performance metrics, predictive analytics, and comments.
2. Generative AI tools can review feedback before it is published.

[10] https://aws.amazon.com/devops/continuous-integration/#:~:text=Continuous%20integration%20refers%20to%20the,for%20a%20release%20to%20production .

7. AI for problem anticipation and mitigation

AI supports both problem anticipation and recovery from partial or total failure of complex systems by speeding up detection, improving diagnosis and increasingly automating much of the remediation. Given the close alignment of anticipation and remediation in IT operations, Artificial Intelligence for IT Operations (AIOps) is becoming both a practice and a set of tools and processes to increase availability and systems reliability.

AIOps utilises AI and machine learning to analyse the vast telemetry from modern IT estates, correlating alerts, detecting anomalies, identifying root causes, and increasingly automating remediation across complex, hybrid environments.

The use of AI to anticipate and remediate system reliability issues can be seen in areas such as:

Resource Optimisation:

AI can dynamically allocate resources (such as compute, storage, or network bandwidth) based on real-time demand and predicted needs, ensuring optimal performance and minimizing bottlenecks during peak loads or disruptive events.

Operational signalling.

1. AI-driven systems can continuously monitor operations and detect real-time anomalies or deviations from standard patterns.
2. Machine learning models can rapidly process vast amounts of data to highlight emerging risks. For example, Amazon uses AI to forecast demand spikes and supply bottlenecks, ensuring availability while reducing excess inventory by 15 to 20 percent.[11]

Risk and control frameworks

1. AI can compare risks and controls in the organisation with industry-/sector-wide data to identify what common risks and controls appear to be missing from the organisation's own risk and control registers.
2. AI can also provide analysis of controls and recommendations for control improvements[12].

[11] https://neontri.com/blog/ai-demand-forecasting/

[12] Examples are available at Diligent - Clarify risk. Elevate governance.

Anomaly detection:

1. Experiments indicate that RNN[13]s (see Annex Glossary) have great potential for finding anomalies in networks. They make use of the sequential dependencies existing in network traffic data, for instance--observing anomalies that would otherwise have been overlooked and addressing false positives[14]. They are effective with both normal network traffic and malicious intrusions, and are a key component in modern cyber defence. They enable real-time monitoring, anomaly detection, and rapid response, far quicker than manual systems[15].
2. In practice it may be that AI generates options for human decision making.

Predictive Analytics for Risk Mitigation

AI can identify patterns within large data sets, finding patterns and anticipating events much more accurately than people can. By leveraging historical data, real-time information, and advanced algorithms, AI can forecast potential disruptions - such as hardware failures, cyber threats, or environmental hazards—before they escalate. This allows organizations to act pre-emptively.

[13] See Annex Glossary

[14] https://ieeexplore.ieee.org/document/10467790

[15] https://www.paloaltonetworks.co.uk/cyberpedia/ai-risks-and-benefits-in-cybersecurity

Engineering Resilience for IT Systems

Annex: Glossary

This Glossary builds on that originally developed by Tom Gilb and later updated with terms, abbreviations and acronyms used in the industry.

200 OK - When your application sends an HTTP request to a server, the 200 code confirms the server received the request, processed it without errors, and returned the requested resource.

Active-Passive mode - one primary active node handles all requests, and one or more secondary nodes remain idle (passive) until a failover is needed.

ADM - activity network diagram, uses arrows to represent the data flow to deliver a service.

Agentic AI - the framework within which AI agents operate, from the algorithms to the architecture, linking them together in an ecosystem.

AI agents may be triggered autonomously for instance to undertake stress testing or recovery actions[1].

AIOps - Artificial intelligence for IT operations, or AIOps, is the application of artificial intelligence (AI) capabilities—such as natural language processing and machine learning models—to automate, streamline and optimize IT service management and operational workflows.

Alert Fatigue – the desensitisation of operations staff caused by excessive or poorly calibrated monitoring alarms. Alert fatigue increases the risk that genuine precursor events are overlooked. Mitigated by threshold calibration aligned to the four Golden Signals (see below).

ALB - Arm's-length bodies: A commonly used term in the UK covering a wide range of public bodies, including non-ministerial departments, non-departmental public bodies, executive agencies and other bodies, such as public corporations.

ANSI - American National Standards Institute

API – Application Programme Interface – for instance to implement data integrity checking.

Application architecture – applied to resilience, software designed to contain (localise the effect of) failures.

[1] https://www.zendesk.co.uk/blog/agentic-ai-in-itsm/#Use%20cases%20of%20agentic%20AI%20in%20ITSM

Annex: Glossary

Architecture – see Application Architecture, Business Architecture, Data Architecture, Enterprise Architecture, Technology Architecture

ART – Automated Recovery Testing: intentionally inducing failures in a controlled environment to assess how well a system rebounds, measuring recovery speed and downtime against declared Impact Tolerance thresholds.

Augmentation of humans by AI is characteristic of early successful applications of AI[2] for instance in healthcare, where analysis of images is mapped to human decision making. Such systems were called *Expert systems*[3] and are now often referred to as Workforce AI.

Authenticity – a non-functional requirement that the information or data presented is genuine and trustworthy.

Automated Recovery Testing – using tools simulate complex disaster scenarios, such as ransomware attacks or data center outages. They allow organizations to test their recovery plans under challenging conditions.

Auto-scaling - a method used in cloud computing that dynamically adjusts the computational resource in a server farm - typically measured by the number of active servers - automatically based on the load on the farm.

Availability refers to the measure of how consistently a system is operational and accessible. High availability has low downtime and continuous operation. Availability measures the impact of service outages on users: it is an external metric of the system.

B2B – Business to Business

B2C – Business to consumer

Backoff - A backoff algorithm resolves contention issues among stations trying to transmit data simultaneously by keeping stations waiting for random time slots before sending data, with the wait time increasing exponentially to prevent collisions and optimize network performance.

BCMT - Business Continuity Management Team: an operational group (often synonymous with the Emergency Management Team) consisting of senior

[2] https://clanx.ai/glossary/human-ai-colaboration#:~:text=%E2%80%8D-,What%20is%20an%20example%20of%20Human%2DAI%20collaboration%3F,medical%20images%20and%20patient%20records .

[3] https://www.freshworks.com/freshservice/resources/cio-guide-to-modern-itsm-report/

managers from all functional areas. Their role is to manage the execution of continuity and recovery plans.

BCI – the Business Continuity Institute[4]

BCM – Business Continuity Management: the holistic management discipline that identifies threats to an organisation, the impacts to business operations those threats might cause, and provides a framework for building organisational resilience and effective response. Aligns with ISO 22301 and the BCI Good Practice Guidelines. See also BCI, BCMT, IBS.

BCS – the Chartered Institute for IT[5].

BGP – Border Gateway Protocol: the routing protocol used to exchange reachability information between autonomous systems on the public internet. BGP misconfiguration is a recurring cause of large-scale outages (see Meta 2021, Facebook DNS/BGP withdrawal) and is a recognised single point of failure in network resilience design.

Blameless Post-Mortem – a structured post-incident review that shifts focus from individual blame to systemic causation, enabling organisations to identify and address weaknesses without fear of retribution. A prerequisite for effective near-miss reporting and continuous improvement.

Blast radius - the full extent of damage that a single failure could cause. What's the worst that could happen? The metaphor is useful for understanding potential risks and the need to balance these with strategic decisions for protecting against real threats.

Blue/Green - Blue/green deployment is a change management strategy for releasing software code. Also known as A/B deployment, it uses two identical hardware environments -- one active (blue) and one idle (green) -- to support safe, low-disruption releases. Blue/green deployments are often used for consumer-facing applications and applications requiring high availability.

Board - The Board's key purpose “is to ensure the company's prosperity by collectively directing the company's affairs, while meeting the appropriate interests of its shareholders and relevant stakeholders”.

BoE – Bank of England

BSI - British Standards Institute

[4] https://www.bci.org
[5] https://www.bcs.org

Annex: Glossary

Bulkheading - much like a ship's hull, partitioning the system so that a failure in one module (e.g., the search engine) is contained and cannot sink the rest of the ship (e.g., the checkout process).

Business analysis – analysis and implementation of information systems in line with the needs of the business.

Business Architecture – applied to resilience, allowing an organisation to maintain its market share through disruptions.

Business Continuity - the capability of the organisation to continue delivery of products or services at acceptable pre-defined levels following a disruptive incident.

Bulkheads - application design that's tolerant of failure. In a bulkhead architecture, also known as a cell-based architecture, elements of an application are isolated into pools so that if one fails, the other elements continue to function.

CAF – Cyber Assessment Framework: the NCSC framework used by UK regulators and Operators of Essential Services (OES) to assess cyber resilience under the NIS Regulations. Structured around four objectives — managing security risk, protecting against cyber-attack, detecting cyber security events, and minimising the impact of incidents.

Canary Deployment – a software release strategy that allows the gradual and controlled rollout of new features or updates to a subset of users or services before full production deployment. Used to limit the blast radius of a defective update and provide an early signal of release failure.

CAP theorem – that a distributed system can deliver only two of three desired characteristics: consistency, availability and partition tolerance (the 'C,' 'A' and 'P' in CAP).

CDIO – increasingly the roles of CIO and CDO are combined in a CDIO.

CDN - A content delivery network (CDN) is a group of geographically distributed servers that speed up the delivery of web content by bringing it closer to where users are.

CDO – Chief Digital Officer, often responsible for digital change.

CEO – Chief Executive Officer

Annex: Glossary

Chaos Engineering - the practice of intentionally injecting faults into a system to test its resilience. The goal is to identify potential failure points and correct them before they cause an actual outage or other disruption.

CI/CD - continuous integration/continuous delivery or deployment. Continuous integration refers to automatically integrating code changes into a shared source code repository. Continuous delivery and/or deployment is a 2 part process for the integration, testing, and automatic deployment of code changes.

CIO – Chief Information Officer: usually tasked with keeping the lights on.

Circuit breakers - These can prevent a caller service from retrying a call to another service when the call has previously caused repeated timeouts or failures.

CISO – Chief Information Security Officer

CISQ – Consortium for Information & Software Quality[6] "Since software resilience is defined as sustainable behaviour, a full measure of resilience requires that static analysis of the system's internal structure be coupled with dynamic analysis of system behaviour. Static analysis determines a resilience element's properly designed existence in the code, while dynamic analysis evaluates its ability to sustain resilient performance."

Cloud - the application of Internet capacity to flexibly allocate storage, processing memory and processing power to execute applications and to retain data with reliable backup provisions. Cloud applications often involve significant machine to machine interactions in addition to user-machine interactions. Cloud based services configure virtual machines to accomplish computational tasks.

CMC – Crisis Management Committee: the senior decision-making body convened during a major incident to direct the organisation's strategic response. Sits above the BCMT and Senior Incident Manager in the escalation chain and owns external communications, regulatory notification, and continuity of authority.

CMDB - configuration management database is an ITIL term for a database used by an organization to store information about hardware and software assets.

CNI - Critical National Infrastructure, infrastructure considered essential by governments for the functioning of a society and economy and deserving of special protection for national security.

[6] https://www.it-cisq.org/

Annex: Glossary

Concentration Risk – the operational resilience risk arising from excessive dependency on a single cloud provider, technology vendor, or third-party service. A single provider failure can propagate across the entire organisation. Mitigated through multi-region and provider diversity strategies.

Confidentiality of data - the privacy of information, including authorizations to view, share, and use it. So, protecting data against unintentional, unlawful, or unauthorized access, disclosure, or theft.

Complex system - composed of more than one elementary and/or complex component. Complex systems can exhibit unpredictable failure characteristics. Most software systems are complex.

COTS - Commercial Off The Shelf Software

CRA - EU's Cyber Resilience Act[7].

A crisis disrupts the whole organisation and causes reputational damage.

Crisis management is the process organizations use to prepare for, respond to, and recover from disruptive and unexpected events (like data breaches, supply chain disruptions, or natural disasters) that threaten to harm the organization, its people, operations, reputation, assets, or stakeholders[8].

Critical Path – the minimum set of infrastructure components, applications, and dependencies required for delivery of a named Important Business Service (IBS). Identifying the critical path is the prerequisite for effective monitoring, failure prevention, and recovery planning.

CSF - the NIST Cybersecurity Framework (CSF) 2.0 provides guidance to industry, government agencies, and other organizations to manage cybersecurity risks.

CSRB – Cyber Safety Review Board: a US public-private body, established under Executive Order 14028, that reviews significant cyber incidents and publishes lessons learned. Notable reviews include Log4j and the Microsoft Exchange Online intrusion. Provides a model for structured, blameless post-incident review at national scale.

CTO - Chief Technology Officer, is the leader of a company's technical and technological department. The CIO concentrates on the information systems to

[7] https://digital-strategy.ec.europa.eu/en/policies/cyber-resilience-act

[8] https://www.pwc.com/ua/en/services/forensic/crisis-management.html

increase their efficiency, the CTO works on the technology strategy to optimize the product or service. The CISO is also often part of the CTO family.

CVE - Common Vulnerabilities and Exposures are a set of security threats that are included in a reference system that outlines publicly known risks. The CVE threat list is maintained by the MITRE Corporation, a nonprofit organization that runs federal government-sponsored research and development centres. https://www.cve.org/

Cyber Security and Resilience Bill (Network and Information Systems) Bill proposes new laws to improve UK cyber defences and protect our essential public services.

Data architecture – when applied to resilience, the ability of data structures to support changes in type and volume of demand.

Data centre - A large group of networked computers typically used by organizations for the remote storage, processing, or distribution of large amounts of data. Data centre performance data is collected and published by Uptime Institute.

Digitalisation - the process of using digital technology – such as computers and the internet. This restructures many domains of social life around digital communication and media infrastructures. Gartner's glossary states "Digitalisation is the use of digital technologies to change a business model and provide new revenue and value-producing opportunities. It is the process of moving to a digital business." Digitalised systems have four main components – hardware which is visible, software which is intangible, data, and networks (telecoms, internet) which are constructed of a mixture of hardware and software. Hardware failures are decreasing in importance whereas software failures are increasing in importance. Data is increasingly seen as a source of strategic advantage.

Defect - A shortcoming, imperfection or lack. See also Risk and Failure[9].

Defence in depth - layers security and resilience controls so that one failed barrier does not expose the core service, whether that barrier is a network access control layer, a firewall, or key management.

Defend as One – UK Government policy to share cyber data and capabilities.

[9] https://blog.cm-dm.com/post/2012/09/14/How-to-differenciate-Bugs%2C-Software-Risks-and-Software-Failures-Part-2

Annex: Glossary

Deployment pipelines - automated processes that govern how software, data products, or machine learning models move from development to production.

Design for recovery shifts attention from Mean Time Between Failures to Mean Time to Repair, favouring approaches such as immutable infrastructure and automated rebuilds.

DevOps - a set of practices that combines software development and IT operations to deliver software solutions.

Disasters are external sources of disruption.

Disaster Recovery exercises (DR) are practice simulations of disasters.

Digital Twin – a virtual replica of a production system or operational environment used for safe stress-testing of upgrades, recovery procedures, and rare failure scenarios without affecting live services. Used in Automated Recovery Testing (ART) and Chaos Engineering programmes.

DLQs – dead-letter queues - a special type of message queue that temporarily stores messages that a software system cannot process due to errors.

DMOP - Degraded Mode Operating Procedures; developing and rehearsing "downtime procedures" (paper-based or alternative) for when primary and secondary systems fail.

DNS - DNS converts human-readable domain names into numerical computer IP addresses.

DORA – Digital Operations Resilience Act: an EU Act designed to strengthen the IT security of financial systems.

DoS – Denial of Service, either due to a crash or by flooding of services with heavy traffic.

DownDetector – a service which captures real time information on outages[10].

DR – Disaster Recovery

Dynatrace – an AIOps platform.

EBA - European Banking Authority

[10] https://downdetector.co.uk/

Annex: Glossary

EIOPA - European Insurance and Occupational Pensions Authority

Emergency Management Team – often also responsible for resilience, that is avoiding or ameliorating risks as well as responding to emergencies.

Enterprise Architecture – Connection between Business architecture and an organization's capability to achieve its current and future objectives.

Error Budget – the allowable quantity of unreliability defined by a Service Level Objective. When the error budget is exhausted, engineering priorities shift from feature delivery to reliability restoration. Used within Site Reliability Engineering governance frameworks.

ESMA - the European Securities and Markets Authority

Event sourcing - the state of a system is determined by a sequence of events. Instead of storing just the current state of the data, event sourcing involves storing all the events that led to the current state.

EverGreen IT - usually refers to 'a technology ecosystem that is continuously changing and evolving, never becoming out-of-date or obsolete'.

Failure - an undesirable and unacceptable system state. For example, a state of failure can result from a defect causing issues such as safety problems, operator discomfort, customer discomfort, but not all are critical to a system's continued survival. Total failure is defined by catastrophe levels.

FaaS - Failures as a Service – a four step process

- Form a hypothesis — Ask yourself, "What could go wrong?"
- Plan your experiment — Determine how you can recreate that problem in a safe way that won't impact users
- Minimize the blast radius — Start with the smallest experiment that teaches you something
- Run the experiment — Make sure to carefully observe the result.

FCA - Financial Conduct Authority (UK)

Fishbone analysis - also referred to as a cause-and-effect diagram or an Ishikawa diagram. It takes the shape of a fish skeleton. The possible causes are listed on the left side. These make up the "bones" of the fish. On the right side, place the effect or problem you are investigating—the "head." This structure provides a quick way to visualize the various causes associated with the effect.

Annex: Glossary

FMEA - Failure Mode and Effects Analysis: a structured methodology for identifying potential failure modes in products, processes, or systems and assessing their causes and consequences.

FS Process – used to refer to the Bank of England's Prudential Regulatory Authority's requirements for financial services organisations:

- Identify important business services (IBS);
- Set impact tolerances for these services which define how much disruption can be absorbed before intolerable harm is inflicted on the users of the services;
- Undertake regular testing against severe but plausible (which goes beyond probabilistic assessment) operationally disruptive scenarios to identify vulnerabilities;
- Take mitigating action so that services can remain within tolerance

GDPR - Data protection legislation controls how your personal information is used by organisations, including businesses and government departments. In the UK, data protection is governed by the UK General Data Protection Regulation.

GDS – Government Digital Service: the UK Cabinet Office unit responsible for digital standards across central government, including the Service Standard, the Technology Code of Practice, and the GOV.UK platform. Sets baseline expectations for resilience, accessibility, and security in public-sector digital services.

GenAI - Generative AI involves AI tools that can generate new content based on specific prompts or instructions. Their use in the creative industries is causing major disruption and challenge to intellectual property regimes[11]. In IT Service Management, GenAI is used to summarise manuals, and recommend actions in the case of an outage.

Golden Signals - four key metrics — latency, traffic, errors, and saturation — for monitoring IT systems' health and performance.

Graceful degradation allows a system under stress to preserve core functions while disabling non-essential features.

GSLB - global server load balancing - method of distributing internet and web application traffic across multiple servers in different geographical locations.

Guardrails - designed to protect against threats such as:

- adversarial inputs that manipulate AI behaviour to produce restricted or unsafe outputs.

[11] https://www.weforum.org/stories/2025/01/the-impact-of-genai-on-the-creative-industries/

- Sensitive information exposure.
- Outputs that include personally identifiable information, proprietary data or sensitive information such as healthcare records.

Hardware - Computer hardware is the name for the physical components that a digital system requires to function.

Health check - monitors the condition and status of backend servers that communicate with a load balancer.

HLD - High Level Design defines the overall system architecture

Honey traps - decoy servers or systems that are deployed next to systems your organization uses for production. Honeypots are designed to look like attractive targets, and they allow IT teams to monitor the system's security responses and to redirect any malign attacker away from their intended target.

HTTP - Hypertext Transfer Protocol is an application layer protocol in the Internet protocol suite for distributed, collaborative, hypermedia information systems.

KYC – Know Your Customer: the regulatory requirement on financial institutions and other regulated entities to verify the identity of their clients and assess risk of illicit activity. A core component of Anti-Money Laundering (AML) regimes. Outages affecting KYC systems can halt customer onboarding and trigger regulatory reporting obligations.

IAC - Infrastructure as Code: Treat environments as reproducible code artefacts. In a provider or regional outage, the organisation should be able to redeploy the required stack into an alternative region using controlled, tested automation.

IBS – Important Business Service

ICO - Information Commissioner's Office (UK)

I(C)T – Information (and Communications) Technology: the combined infrastructure, applications, networks, and services used to process and communicate information. Used in EU regulation (notably DORA) as the umbrella term covering everything from end-user devices to cloud platforms.

IEC - International Electrotechnical Commission

Impact - the estimated or actual numeric effect of an event, in service resilience work the event is a failure of an IT system, usually of software.

Impact tolerances in operational resilience are used to describe the extent to which a failure of a service is acceptable to the organisation in terms of disruption or damage to users: both numbers affected and duration of outage. Defined within the Bank of England's FCA/PRA Operational Resilience framework as the maximum level of disruption that an organisation can absorb before intolerable harm is inflicted on users of an Important Business Service. It is expressed in time, and tested against severe but plausible scenarios.

Important Business Service – an important concept in the Bank of England's FS Process.

Incident management is a process used to respond to and address unplanned events that can affect service quality or service operations.

Integrity - the accuracy, completeness, consistency, and validity of an organization's data.

Infosec - the processes and tools designed and deployed to protect sensitive business information from modification, disruption, destruction, and inspection.

Internet - The computer network providing a variety of information and communication facilities, consisting of interconnected networks using standardized communication protocols.

Internet of Things – used to describe networks with physical devices, vehicles, appliances, and other physical objects that are embedded with sensors, software, and network connectivity, allowing them to collect and share data.

IRCA - the International Register of Certificated Auditors. IRCA is the world's original and largest international certification body for auditors of management systems and is a division of the Chartered Quality Institute (CQI).

ISMS – Information Security Management System - a centrally managed framework that enables you to manage, monitor, review and improve your information security practices in one place.

ISO - International Standards Organisation

ITIL - (also known as Information Technology Infrastructure Library) is a framework with a set of practices (previously processes) for IT activities such as IT service management (ITSM) and IT asset management (ITAM) that focus on aligning IT services with the needs of the business. There is no formal independent third-party compliance assessment available to demonstrate ITIL compliance in an organization. Certification in ITIL is only available to

individuals and not organizations. Since 2021, the ITIL trademark has been owned by PeopleCert[12].

ITOC – IT Operations Centre: the operational team and physical or virtual facility responsible for day-to-day monitoring, event management, and incident response across the IT estate. Often integrated with the SOC (Security Operations Centre) and NOC (Network Operations Centre) into a unified observability function.

ITSM – IT Service Management: the discipline of designing, delivering, managing, and improving the IT services an organisation provides to its users. ITIL is the dominant framework. ITSM tooling underpins incident, problem, change, and configuration management — the operational substrate on which DR and BCM depend.

ITSOC principles – IEEE Information Theory Society principles[13]

IT SOPs – Standard Operating Procedures; detailed, step-by-step guides that document key processes and procedures.

IT systems – an alternative term for digitalised systems.

Jitter - term used in computer networking to refer to packet delay variation — the variance in the rate at which packets arrive at a destination (the average of the squared difference from the mean packet arrival rate). "High jitter" indicates that packets are arriving at significantly varying rates, possibly due to network congestion, packet loss, and routing of packets in a stream through different paths. High jitter can significantly degrade the performance of real-time web applications, including voice/video streaming and online gaming.

Latency - the delay that happens between when a user takes an action on a network or web application and when it reaches its destination, which is measured in milliseconds.

Legacy software. A legacy system is an older (but not obsolete) part of the IT landscape that's still in use today.

LGD - Lead government department: a government department that has other public sector organisations within its purview. LGD is usually the department with primary policy responsibility for the risk and expertise for the area impacted by the emergency scenario.

[12] https://www.peoplecert.org/

[13] https://www.itsoc.org/itsoc-search?search_api_fulltext=principles

LLD - Low Level Design explains the detailed internal logic and component structure

LLMs – large language models. These are highly effective for automating text-based tasks, generating human-like responses, providing 24/7 customer support, creating content, analysing data, and powering intelligent assistants. Many cloud service providers use LLM models to identify and predict faults, as well as moving services and data around within the infrastructure. The well-known problems of LLMs such as bias, data privacy concerns and resource used for training and deployment[14], are less important in most service resilience applications.

Load Balancer - used to distribute capacity during peak traffic times, and to increase reliability of applications. Since they rely on pools of backend resources, maintaining visibility into the health of those servers is essential to effectively routing client traffic.

Load Shedding – a resilience technique that deliberately drops non-essential background tasks or lower-priority requests during periods of high saturation to protect critical path services. Load shedding decisions must be pre-defined in Degraded Mode Operating Procedures to prevent real-time triage under pressure.

Loose coupling prevents one system from depending on another's immediate success, often through asynchronous messaging. This may be through message queues, asynchronous streaming or transient data stores.

Major Incident Management refers to the team and processes within the broader Incident Management framework, designed to handle high-impact disruptions to IT services that can severely affect business operations. They have close links to Disaster Recovery and Crisis Management. (See MIM below)

Malware – software that is specifically designed to disrupt, damage, or gain - unauthorized access to a computer system.

Metric - any kind of numerically expressed system attribute. A metric is defined in terms of a specified scale of measure, and usually one or more numeric points on that scale.

Micro-segmentation – a network security technique that divides a network into small, isolated zones to contain failures and limit lateral movement of threats. A core component of Zero Trust Architecture and system compartmentalisation strategies.

[14] https://lumenalta.com/insights/7-surprisingly-powerful-large-language-model-applications

Microservices - organise an application into a collection of loosely coupled, fine-grained services that communicate through lightweight protocols.

Microsoft SQLServer - a proprietary relational database management system using Structured Query Language (SQL, often pronounced "sequel").

MIM – Major Incident Management: the ITIL process for restoring service as quickly as possible after a high-impact incident, while coordinating communications and minimising business disruption. The Major Incident Manager is the operational counterpart to the Senior Incident Manager in the Business Continuity hierarchy.

ML – machine learning – the use of AI to train models on datasets.

Modular design - subdivides a system into smaller parts called modules, which can be independently created, modified, replaced, or exchanged with other modules or between different systems.

MSP – Managed Service Provider: a third-party organisation that delivers and manages IT services, infrastructure, or applications on behalf of a client, typically holding privileged access to client systems.

MTBF - mean time between failures

MTTA - mean time to acknowledge

MTTD - the Mean Time to Detect is the average time it takes a team to discover a security threat or incident.

MTTF - mean time to failure.

MTTR - mean time to repair/recover: the average amount of time it takes for a particular service to recover from failure. It includes the repair time and testing time.

Multi-factor authentication (MFA) [15]is a layered approach to securing data and applications where a system requires a user to present a combination of two or more credentials to verify a user's identity for login. MFA increases security because even if one credential becomes compromised, unauthorized users will be unable to meet the second authentication requirement and will not be able to access the targeted physical space, computing device, network, or database.

[15] https://www.cisa.gov/resources-tools/resources/multi-factor-authentication-mfa

Multi-site active-active splits traffic across two or more fully functional regions so that recovery is effectively immediate, though at very high cost.

MVP - A minimum viable product is a version of a product or service with just enough features to be usable.

NATS – National Air Traffic Services

Natural Accident Theory[16] predicts that in complex and tightly coupled systems, like today's IT systems, an accident type known as the "normal accident", is inevitable.

NCSC – National Cyber Security Centre: the UK technical authority on cyber security, part of GCHQ. Publishes the Cyber Assessment Framework (CAF), the 10 Steps to Cyber Security, and incident response guidance. The primary authoritative source for UK CNI and OES cyber resilience expectations.

NED – Non-executive (usually Board) Director.

NIS2 – Network and Information Systems Directive 2: the EU directive (Directive (EU) 2022/2555) that replaces and broadens the original NIS Directive, expanding the scope of essential and important entities, tightening incident reporting timelines, and introducing direct accountability for management bodies. It applies to a wide range of critical infrastructure operators, such as energy, healthcare, financial services, transport, drinking water supply and digital infrastructure and telecommunications. The metrics defined are

- User hours lost
- data integrity,
- threat to life or health,
- financial damage.

NIST – NIST promotes U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology.

Near-Miss Register – a structured log of transient and intermittent faults that did not result in a full service outage but represent precursor signals for potential failure. Near-miss data feeds directly into the preventive action cycle and is a prerequisite for blameless post-mortem culture.

NEDA[17] - membership organisation for Non-Executive Directors: it supports competent NEDs from diverse backgrounds with the right skills, knowledge and mindset to ensure they have a positive impact in the boardroom and beyond.

[16] https://www.sciencedirect.com/science/article/pii/S0265964624000444

[17] https://www.nedaglobal.com/

NEN - Royal Netherlands Standardization Institute

NFR's – Non Functional requirements: they usually include

- security,
- reliability,
- maintainability,
- scalability,
- usability.

Nine Banks – in 2025, the Parliamentary Treasury Select Committee found that nine of the top UK banks had accumulated at least 33 days' worth of IT outages over the preceding two years.

NTP is synchronised time protocol for reliable forensic timelines in reporting, security and audit logs across a network.

Observability - the ability to understand and react to the state of an IT system or application. Like monitoring, observability relies on outputs, logs, and performance metrics.

OES – Operator of Essential Services: these include critical infrastructure (water, transport, energy) and other important services, such as healthcare and digital infrastructure.

Open-Source - a business model in which source code is made available under a license that permits organisations or individuals to use, modify, share, and improve the software. Claimed advantages: large numbers of skilled individuals may be more effective in discovering errors, and that the negotiation over software features is more inclusive than proprietary products. Claimed disadvantages; the frequent absence of technical support, particularly for end users, and the potential for originators to lose interest in maintaining and upgrading the software over time.

Operational resilience – an organisation's ability to respond to and overcome adverse circumstances during operation that might cause financial loss or disrupt services to users. The term is widely used in financial services regulation.

Oracle RAC - Oracle Real Application Clusters is a database clustering solution that enables multiple servers to operate as a single, unified system.

OS – Operating System: the foundational software layer that manages hardware resources and provides services to applications. OS-level vulnerabilities,

patching cadence, and end-of-life status are core inputs to resilience and SBOM assessments. Examples include Windows Server, Linux distributions, and z/OS.

OSI - The Open Systems Interconnection model describes seven layers that computer systems use to communicate over a network. The OSI model is divided into seven distinct layers, each with specific responsibilities, ranging from physical hardware connections to high-level application interactions.

PACELC - the PACELC Theorem is an extension of the CAP Theorem. One of the questions that CAP Theorem wasn't able to answer was "what happens when there is no Partition, what logical combination can a Distributed System have?". In addition to Consistency, Availability, and Partition Tolerance (see CAP Theorem) it also includes Latency as one of the desired properties of a Distributed System. The acronym PACELC stands for Partitioned, Availability, Consistency Else Latency, Consistency.
Platform - a general term for a web site designed to be accessible to large numbers of users. Sites designed for online sales may serve as platforms when they support multiple sellers or customers or both.

Pilot light - keeps a minimal version of the environment, often the database layer, always running and synchronised so that application capacity can be scaled up during a disaster.

PITR - point-in-time recovery - the ability to restore data or a system to a specific point in time. In the PITR process, the administrator can restore data to a selected timestamp after an error, effectively undoing any changes made after that point.

Playbook – a strategic guide that defines roles, responsibilities, and decision frameworks for responding to a category of event. In most mature IT organisations, playbooks govern; runbooks define execution.

Poll-based Monitoring – (also called Pull-based Monitoring) an approach in which the monitoring application actively requests health status from sub-systems at defined intervals. Provides a resilience advantage over push-based logging: health data remains available even when a sub-system is overwhelmed and cannot push logs. See also Push-based Logging.

POS - includes the hardware and software related to transactions, such as the cash drawer, credit card swipe bar, barcode scanners, receipt printers.

PRA - the Bank of England's Prudential Regulatory Authority

Push-based Logging – a monitoring approach in which sub-systems proactively send log data to a central server. If a sub-system is overwhelmed, log

transmission may be delayed or lost. Should be supplemented by poll-based monitoring as a failsafe layer.

QE - Quality Engineering is the practice of proactively integrating quality standards, automated testing, continuous validation, and observability.

Quorum Systems or Triple Modular Redundancy - a cluster of components must vote and reach a majority agreement before a transaction is committed or a new cluster leader is elected. If an enterprise operates a cluster of three nodes, at least two must agree to establish the truth.

RACI - stands for the four different roles in governance:

- Responsibility
- Accountability
- Consulted
- Informed.

RC - root cause: Root cause analysis helps organizations decipher the root cause of the problem, identify the appropriate corrective actions and develop a plan to prevent future occurrences. It aims to implement solutions to the underlying problem for more efficient operations overall[18].

RDSP – Registered Data Service Provider

Recovery playbooks detail historic incidents and how recovery from these was achieved.

Redundancy removes or reduces single points of failure by duplicating critical components across failure domains. If a database is a critical component, this requires an architecture that keeps data in sync, either synchronously or asynchronously. Critical components can exist in all layers of the architecture.

Resilience describes the system's ability to withstand and recover from disruptions. This includes failures and cyber-attacks. It focuses on the effectiveness of anticipation prior to a disruption and the speed and effectiveness of response and recovery after a disruption or outage. Resilience is usually measured by internal metrics.

Retro - A retrospective or 'retro' is a meeting held regularly with your team to discuss what happened over a period.

[18] https://www.ibm.com/think/topics/root-cause-analysis#:~:text=Root%20cause%20analysis%20helps%20organizations,for%20more%20efficient%20operations%20overall.

Annex: Glossary

Risk - expose (someone or something valued) to danger, harm, or loss. Risks are quantifiable when they are known and their probability of occurrence can be assessed.

Risk Management - a process that allows individual risk events and overall risk to be understood and managed proactively.

RNN - recurrent neural network: a deep neural network trained on sequential or time series data to create a machine learning (ML) model that can make sequential predictions or conclusions based on sequential inputs.

RPO – Recovery Point Objective: the maximum tolerable period of data loss measured in time. RPO defines the point to which systems must be restored following a disruption. Determined by data criticality and backup cadence. Defined per IBS in FS Process and tested alongside RTO.

RTO - Recovery Time Objective: the target amount of time in which a feature must be restored to avoid unacceptable disruption.

RUN group performs all the operations and provides support to keep the technology environment operating and meeting service-level expectations.

SaaS – software as a service - a cloud-based software delivery model where individuals or organisations subscribe to applications rather than purchasing and installing them locally

Saturation - refers to the point at which a system's resources, such as CPU or memory, are fully utilized and cannot handle any more tasks or data.

SBOM – software bill of materials: components are described in a standard format.

Secure and Resilient by Design Team: this team establishes common profiles and architectural patterns to ensure that new services, support, and standards are "secure by design".

Security Operations Centres provides the dedicated team, processes, and technology to defend against cyberattacks.

Service – the action of doing work for another. Service quality is normally measured through the perception of the user of the service.

Service outage – time for which a service is not available to users. The duration of the outage and the number of users affected combine to define “lost user hours”

Service resilience – the factor in operational resilience that is based on the ability of the digital systems to prevent or mitigate software risk, as measured on the user side.

SfIA – a widely adopted framework for defining the skills and competencies needed to deliver, manage and protect digital capabilities — including in critical areas such as AI, data science and cybersecurity[19].

Sidecars - offload resilience logic (retries, timeouts, monitoring) to a separate process (like a Service Mesh) so the main application code remains simple and focused.

SIEM – Security Information and Event Management: a platform that aggregates, correlates, and analyses log and event data from across the IT estate to detect security incidents and support investigation. A core component of detection capability under NIST CSF and the NCSC CAF.

SLA – Service Level Agreement, usually part of a contract with suppliers

SLO – Service Level Objective, usually an internal metric

SME - Subject Matter Experts. Often, the SME role contributes to cross-functional projects as needed, but it's not their full-time job. For example, a product development manager may be the organization's subject matter expert on artificial intelligence.

SoA – Statement of Applicability is a mandatory document used in frameworks like ISO 27001 and ISO 42001. It acts as a bridge between your risk assessment and your chosen security controls.

Split-brain - a state indicating data or availability inconsistencies originating from the maintenance of two separate data sets with overlap in scope. This can be because of two overlapping functions of servers in a network design, or a failure condition based on servers not communicating and synchronizing their data to each other.

SPOF – Single Point of Failure: a component, system, or dependency whose failure would cause the complete loss of a critical service or capability. SPOFs must be identified through FMEA and Application Dependency Mapping, and eliminated or mitigated through redundancy, failover, or architectural redesign.

[19] https://sfia-online.org/en

Annex: Glossary

Software - a set of instructions, data or programs used to operate computers and execute specific tasks. Software quality has historically been measured through four aspects – reliability, efficiency, security and maintainability – from the well-known CISQ software quality model.

Software failure - when a software system no longer complies with the specifications that were initially defined for it, which means that it does not behave as expected. Most software systems today are complex tightly coupled systems, liable to unpredictable failures.

Software resilience - the capability of a software system to recover from a software failure as measured on the supply side. Software resilience does not consider lost user hours or the other three metrics in the NIS framework.

Software risk - the probability that the software system will fail. Most software systems today are complex tightly coupled systems, liable to unpredictable failures.

SRE - Site Reliability Engineering: the practice of using software tools to automate IT infrastructure tasks such as system management and application monitoring.

SS1/21 – PRA Supervisory Statement 1/21: the Prudential Regulation Authority statement on operational resilience for banks, building societies, PRA-designated investment firms, and insurers. Requires firms to identify Important Business Services, set Impact Tolerances, map dependencies, and demonstrate the ability to remain within tolerance during severe but plausible scenarios. Aligned with the FCA's PS21/3 and the Bank of England's policy statement on operational resilience.

Stakeholder - any person, group or object, which has some direct or indirect interest in a system. Stakeholders of IT systems are both internal to the organisation and external. Service resilience emphasises the external stakeholders' view in terms of lost user hours, damage to data integrity, threat to life or health, and financial damage.

Statelessness - design of application nodes so they can be replaced automatically. If a node fails, the platform should terminate and recreate it without loss of user session data, relying on externalised state where appropriate.

Supply Chain - the core activities within the organisation to convert raw materials or component parts into finished products or services. In the context of IT systems, the software and hardware components from a range of different sources, used to deliver a service to users.

System - any useful subset of the universe that we choose to specify. It can be conceptual or real. A system can be described fundamentally by a set of attributes. The attributes are of the following types:

- function: 'what' the system does
- performance: 'how good' (quality, resource saving, workload capacity)
- resource: 'at what cost' (resource expenditure)
- design: 'by what means.'
- requirements
- dependencies
- risks
- priorities.

System Compartmentalisation – an architectural resilience technique that isolates components and services into discrete zones so that a failure in one zone cannot cascade across the entire organisation. The primary function of resilience architecture. See also Bulkheads and Micro-segmentation.

TCP/IP - The TCP/IP model defines how devices should transmit data between them and enables communication over networks and large distances.

Test/Quality Engineer (QE) - Quality Engineering (QE) is the practice of proactively integrating quality standards, automated testing, continuous validation, and observability into every phase of the software development lifecycle.

Technology architecture – applied to resilience, the highest Level four is inherent resilience by design: architected into the technology stack from the start through inherent redundancy and with active monitoring at the data level, which includes anomaly detection and mitigation.

Technology Recovery Manager (TRM) - manages the recovery process after system outages, incidents, or disasters to ensure minimal downtime and service disruptions.

Third-Party - an individual or an entity that is not directly involved in an agreement or a transaction, but may still have a role in it. For example, in a transaction between an organisation (first party) and their customer (second party), a third party can be the payment provider.

TIBER-EU – Threat Intelligence-Based Ethical Red Teaming for the European Union: the European Central Bank framework for controlled, intelligence-led red team testing of critical financial entities. Forms the methodological basis for Threat-Led Penetration Testing (TLPT) under DORA. See also TLPT, DORA.

TMR – see Quorum

TOGAF - The TOGAF Standard, a standard of The Open Group, is an Enterprise Architecture methodology and framework in wide use[20]. Individuals can achieve certification.

TLPT – Threat-Led Penetration Testing: advanced, intelligence-led red team testing mandated by DORA for significant financial entities. Conducted against live production systems using tactics, techniques, and procedures drawn from real threat actors, with the objective of validating detection and response capability. Methodologically based on TIBER-EU.

TRM - Technology Recovery Manager – part of the RUN Group.

UK Cyber Security and Resilience Bill – see Cyber security and Resilience Bill

Uptime Institute[21] - collates data on data centre and infrastructure outages.

Uncertainty - an expression of doubt about how an impact estimate, or measurement, of an attribute reflects reality. We are 'uncertain' as to whether the reality is better or worse with respect to an observed or estimated value of an attribute, and by how much it differs. Uncertainty prevents estimation of risk.

User hours lost – an important measure for service resilience within the NIS framework.

Value at Risk (VaR) - estimates how much a set of investments might lose (with a given probability), given normal market conditions, in a set time period,

Vibe code – term for software development assisted by AI. The software developer describes a project or task to a LLM which generates source code automatically. This may involve accepting AI-generated code without thorough review of the output, instead relying on results and follow-up prompts to guide changes.

Virus – a program that spreads by first infecting files or the system areas of a computer or network router's hard drive and then making copies of itself. Some viruses are harmless, others may damage data files, and some may destroy files. Viruses are primarily spread through email messages. Unlike worms, viruses often require some sort of user action (e.g., opening an email attachment or visiting a malicious web page) to spread.

[20] https://www.opengroup.org/togaf

[21] https://uptimeinstitute.com/

Vulnerability - in cybersecurity, a weakness that can be exploited by cybercriminals to gain unauthorized access to a computer system. We use the term more widely, to include known weaknesses in operational software.

WAN – Wide Area Network: a telecommunications network that extends over a large geographic area, typically connecting multiple sites of an organisation. WAN resilience — including diverse carriers, SD-WAN failover, and dual-path routing.

Warm standby - a scaled-down but functioning environment online, reducing RTO further because traffic only needs to be redirected.

Workaround - typically a temporary fix that implies that a genuine solution to the problem is needed.

WORM - Write Once, Read Many – a data storage device in which information, once written, cannot be modified.

Zero Trust Architecture – a security model based on the principle of "never trust, always verify", replacing perimeter-based security with identity-centric access controls and micro-segmentation. In the context of operational resilience, Zero Trust Architecture limits the blast radius of a breach or failure by preventing lateral movement across the organisation's infrastructure.

ZTNA – Zero Trust Network Access: an architectural approach that grants application-level access based on verified identity, device posture, and context rather than network location. Replaces traditional VPN-based perimeter access. A practical implementation pattern for Zero Trust Architecture (see entry) and a control that limits blast radius during a breach.

Engineering Resilience for IT Systems
Annex: Public Reports

This Annex has short summaries and links to some reports identifying causes and impacts of IT outages. They are listed in alphabetic order of the organisation affected rather than the body compiling the report.

AT&T nationwide mobile network outage (February 2024)

Source: https://www.xurrent.com/blog/it-outages

The outage blocked 92 million calls, including 25,000 emergency 911 calls. Mobile payments failed at countless businesses. Logistics providers lost real-time tracking, forcing manual coordination. First responders were left with dangerous communication gaps.

The root cause was that a network configuration change was pushed without sufficient staged testing or rollback safeguards. Once deployed, the faulty settings spread quickly, affecting interconnected systems nationwide.

The immediate priority was to identify and reverse the faulty configuration. AT&T restored service region by region, validating stability before bringing each network segment back online. Customer credits were issued, and an internal review was conducted to address gaps in change management.

AT&T agreed to a $950,000 settlement with the FCC, but the financial penalties tell only part of the story. The reputational cost for a carrier of this size, especially after disrupting emergency services will continue to shape customer trust for years.

AWS (February 2017)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

A simple debugging of the billing system for Amazon's Simple Storage Service (S3) took services down for approximately four hours. While attempting to remove servers for a subsystem, an S3 engineer mistyped a command that then removed a "larger set of servers ... than intended," as Amazon described it.

The removed servers supported two key subsystems in northern Virginia, and the disruption meant that they required a full restart. During this time, other AWS services that relied on S3 for storage, including Elastic Compute Cloud and Elastic Block Store, were down.

While S3 subsystems are designed in such a way to minimize customer disruption in the event they fail, the subsystems in this region had not been completely restarted for years. As such, the restart process took four hours and resulted in

an AWS outage that cost companies relying on the service millions of dollars in losses.

AWS (October 2025)

Source: https://www.bbc.co.uk/news/articles/cev1en9077ro

AWS services went down at about 7am on 21st October 2025, when a failure occurred at its data centre in northern Virginia. It took out major social media platforms like Snapchat and Reddit, banks like Lloyds and Halifax, and games like Roblox and Fortnite. Up to 11 million sites were affected. At around 23:00 BST, Amazon said all AWS services had "returned to normal operations."

The outage was a Domain Name System (DNS) error. DNS is supposed to act like a map, and when AWS lost it, platforms like Snapchat, Canva and HMRC were all still there but AWS couldn't see where they were. These errors are usually a maintenance issue or a server failure. Sometimes that's human error, someone misconfiguring something somewhere, or in extreme cases a cyber attack - although there's no evidence of this.

British Airways (May 2017)

Source: https://www.forrester.com/blogs/17-06-15-lessons_learned_from_the_recent_british_airways_outage/

A contractor doing maintenance work at a British Airways data center inadvertently switched off the power supply, knocking out the airline's computer systems. Upon reconnection, an uncontrolled server recovery cascaded through the facility. Approximately 75,000 passengers were stranded. The estimated cost exceeded £80 million.

Key failures: No DMOP prescribing controlled circuit-by-circuit restoration; no mandatory wait-and-check gates; unavailability of trained staff for complex recovery.

Lesson: Power restoration is a degraded mode operation requiring a prescribed, rehearsed, gate-controlled procedure.

British Library (October 2023)

Source: https://www.bl.uk/stories/blogs/posts/learning-lessons-from-the-cyber-attack

The British Library is the United Kingdom's main central library, retaining a copy of every book published in the UK and much else. Its computer software systems were destroyed by a targeted ransomware attack in October 2023. A major ransomware attack on 28 October encrypted or destroyed much of the server estate. Some 600GB of files, including personal data of Library users and staff, were taken. When it became clear that no ransom would be paid, this data was put up for auction and subsequently dumped on the dark web. As well as the exfiltration of data for ransom, the attackers' methods included the encryption of data and systems, and the destruction of some servers to inhibit system recovery and to cover their tracks.

The server destruction had the most damaging impact on the Library: while they had secure copies of all digital collections – both born-digital and digitised content, and the metadata that describes it – they lacked viable infrastructure on which to restore it. The re-build of infrastructure, on equipment approved and purchased before the attack, started in December 2023.

Major software systems could not be brought back in their pre-attack form, either because they were no longer supported by the vendor or because they will not function on the new secure infrastructure. This includes our main library services platform, which supports services ranging from cataloguing and ingest of non-print legal deposit (NPLD) material to collection access and inter-library loan. Other systems will require modification or migration to more recent software versions before they can be restored in the new infrastructure. Cloud-based systems, including finance and payroll, have functioned normally throughout the incident.

Despite an effective crisis management plan and intact offline backups, two years later many of its external-facing services were still unavailable. The library's outdated legacy systems could not be reinstated as before, and the backups proved incompatible with new, more secure, systems.

The British Library's management concluded that "Given that no security is perfect, the ability to quickly recover is essential when (not if) an attack is successful. Investment in security needs to be balanced against investment in back-up and recovery capabilities."

Cloudflare (November 2025)

Source: https://controld.com/blog/biggest-cloudflare-outages

On November 18th 2025, Cloudflare suffered a global outage that affected roughly one in five webpages globally at the height of the incident, and one-third of the world's 10,000 most popular websites, apps, and services. For several hours, users struggled to reach major services, including X (formerly Twitter), ChatGPT, Spotify, Canva, Zoom, Coinbase, popular retailers, and even outage-tracking sites like Downdetctor.

In a detailed post-mortem, Cloudflare explained that the root cause wasn't a cyberattack. Instead, it lay inside its Bot Management system. A change to database permissions on a ClickHouse cluster altered how a query behaved, causing it to return duplicate rows when generating a “feature file” used by a machine-learning model that scores bot traffic. That configuration file suddenly grew far larger than normal. Once this oversized feature file was pushed out across Cloudflare’s global network, it exceeded a hard limit in the bot module, causing the core proxy software that processes customer traffic to panic and crash, resulting in a flood of errors for sites that rely on Cloudflare.

Engineers eventually halted generation of the faulty file, injected a version known to be good, and restarted key services. Cloudflare has since described the episode as its worst outage since 2019 and has committed to hardening how internal configuration files are validated, adding more global kill switches, and ensuring that a single buggy update can’t take such a large portion of the internet offline again.

Resolution: Cloudflare's “Code Orange: Fail Small” programme[14] now stages and size-bounds configuration changes before global propagation.

Lesson: your vendor's configuration discipline is your availability. Ask for evidence of it at selection and review.
Lesson: Without a remote kill switch, remediation reverts to the slowest available recovery path. The Cloudflare incident illustrates the operational cost of this control's absence directly; a feature flag toggle would have isolated and disabled the offending component without a full redeployment. Every new feature should consider the extent of its blast radius and if therefore it requires a toggle-off-available without a release.

CrowdStrike (July 2024)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

The CrowdStrike outage is one of the biggest IT outages of all time.

Just after midnight Eastern Standard Time on July 19, 2024 CrowdStrike rolled out an update for its Falcon Sensors to Microsoft Windows hosts around the world. But the update contained a faulty configuration file that crashed the machines, causing a blue screen of death. CrowdStrike discovered the issue and rolled back the update a little over an hour later, but it was too late for any systems that had already checked in with CrowdStrike's cloud updating service. According to Microsoft's estimates, approximately 8.5 million devices were affected in multiple industries, including travel, finance and healthcare.

Because the problematic file prevented Windows from booting, the recommended fix was slow, requiring users to start up machines in Safe Mode and navigate to the directory where the file was stored so they could delete it. Microsoft and CrowdStrike eventually released instructions on how the problem could also be addressed by creating and using bootable USB drives. Regardless of the approach, the sheer scale of the outage made remediation a cumbersome process.

Though some systems were restored and brought back online within hours, it wasn't until 10 days later that CrowdStrike declared approximately 99% of Windows sensors were online. The impact of the outage could be seen as American Airlines, United Airlines and Delta Airlines were still experiencing problems days after the initial failure.

There was no staged rollout (canary or ring deployment) and no effective kill switch, meaning the fault could not be contained or rapidly reversed once deployed.

Lesson: mandatory staging gates and rollback obligations apply to all update types, including content and signature updates — and they must be contractual, because the deployment decision sits with the supplier.

Lesson: Regression testing should cover resilience properties, not functional correctness alone. Resilience needs to be proven through testing, not assumed through classification.

Dyn (October 2016)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Dyn's outage on Oct. 21, 2016, lasted approximately two hours. The outage was due to a distributed denial of service (DDoS). Dyn was a domain name system (DNS) provider responsible for taking human-friendly URLs, or domain names, and translating them into IP addresses.

The attack was carried out through a botnet -- a series of internet-connected machines infected with the same malware known as Mirai -- which all made coordinated and repeated calls to Dyn's servers, shutting them down completely. Many big-name sites and services such as CNN, Netflix, Twitter and Reddit relied on Dyn for DNS services throughout the U.S. and Europe, and they were all knocked out for the time that Dyn was down.

Equifax Data Breach

Source: Top 5 Instances When Legacy Systems Were Compromised: DBot Journal
In 2017, Equifax experienced one of the most devastating data breaches in history, exposing sensitive information of 147 million Americans. The breach occurred because Equifax had failed to patch a known vulnerability in an Apache Struts web application, which was part of their legacy system.

The breach led to nearly $700 million in fines and settlements, damaging Equifax's reputation and trustworthiness as a credit reporting agency.

What Could Have Prevented It

Had Equifax kept its legacy systems up-to-date with the latest patches, the breach might have been avoided. Regular vulnerability scanning and patch management are essential components of IT security.

FAA NOTAM system failure (January 2023)

Source: https://www.xurrent.com/blog/it-outages

The NOTAM system is a critical safety service, delivering real-time flight operation alerts to pilots and airlines. On January 11, 2023, the U.S. Federal Aviation Administration's Notice to Air Missions (NOTAM) system went offline after a contractor mistakenly deleted essential database files. While the trigger was human error during maintenance, the deeper issue was the lack of safeguards to prevent accidental deletion of critical production data, along with insufficient redundancy to fail over to an unaffected copy of the system.

The outage delayed more than 32,000 flights and cancelled over 400. Airlines absorbed millions in direct costs from refunds, rebookings, and overtime. Passengers faced significant travel disruption, and the event drew national attention to the fragility of critical government IT systems.

As a result, the FAA rebuilt the database from backups and gradually restored service. The agency committed to implementing stricter access controls and improving backup verification processes.

Fastly (June 2021)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Fastly is a content delivery network powering operations such as BBC, Shopify, Amazon, CNN, and the U.S. and U.K. governments, all of which were affected when Fastly went down on June 8, 2021.

Fastly released a software update in May 2021 that contained a bug, only triggered under specific circumstances. It took nearly a month before it became known. At approximately 5:47 a.m. EST on June 8, a customer submitted a configuration change that, while not problematic itself, created a scenario that triggered the bug. The scale of the outage was substantial, with Fastly reporting that 85% of their network returned errors.

Remediation was swift. In the blog post reporting on the incident, Fastly said within 49 minutes, 95% of their network had returned to normal, with a full bug fix rolling out hours later. Even though sites were only down for a few minutes, the visibility of the outage was high as news outlets couldn't report on the news and stores couldn't make sales.

Google (December 2020)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Even though the downtime was only 47 minutes, users around the world on the morning of Dec. 14, 2020, felt the effects of not being able to access any services that required a Google account, including Gmail, Google Drive and YouTube.

Google uses a quota system to manage and allocate storage for its authentication services and switched to a new system earlier that year. Unfortunately, parts of the old quota system were left in place during the cutover, causing it to incorrectly report on resource usage, according to a post from Google regarding the incident.

While there were some fail-safes in place, none of them accounted for this scenario. New authentication data could not be written, so the data quickly turned stale, resulting in errors on authentication lookups and crashing the system.

Healthcare ransomware incidents in US (2023–2024)

Source: https://www.xurrent.com/blog/it-outages

Between 2023 and 2024, ransomware attacks on healthcare providers targeted U.S hospitals, clinic networks, and specialized care facilities. These attacks encrypted critical systems, including electronic health records, medical imaging, and prescription systems, effectively halting many clinical operations until systems were restored.

Downtime in healthcare averaged 24 days per incident, far longer than in most other industries. Costs per minute ranged from $5,300 to $9,000 for medium-to-large hospitals. The CrowdStrike global outage alone resulted in $1.94 billion in healthcare losses, showing how dependent medical services are on uninterrupted IT access. Beyond the financial toll, the operational disruption forced hospitals to cancel procedures, delay diagnoses, and revert to paper-based workflows. For patient care, the stakes were measured not only in dollars but in safety outcomes.

In many cases, ransomware entered through phishing attacks, compromised remote access, or unpatched vulnerabilities. Once inside, lateral movement was often unchecked due to insufficient network segmentation, enabling the encryption of large swaths of systems at once.

In response, affected hospitals activated downtime protocols and shifted to manual processes for admissions, charting, and medication orders. National

health agencies coordinated with cybersecurity teams to contain and eradicate the malware. Restoring full operations often required rebuilding entire environments from clean backups and verifying that no malicious code remained.

Iberian blackout (April 2025)

Source: https://www.bbc.co.uk/news/articles/cvg0r3z3lvqo

On 28 April 2025, shortly after midday, the Iberian peninsula's electricity supply was shut down, causing widespread chaos, cutting internet and telephone connections and halting transport links. Some areas did not recover power for 16 hours.

Entso-e, the European Network of Transmission System Operators for Electricity, described it as "the most severe and unprecedented blackout that had occurred in Europe in the past 20 years, with a major impact on citizens and society as a whole" in Spain, Portugal and to a lesser extent, France. Its report said that there were "multiple interacting factors" which led to the blackout.

An uncontrolled, sudden increase in voltage in the system "on a day with multiple concurrent phenomena" led to instability and "cascading generation". However, voltage controls of local energy generators were not fully aligned with those required by the grid operator. In some cases, manual control of voltage meant that the system responded relatively slowly to changes in the network. Further, the Spanish grid's voltage range is greater than that of other countries, potentially leading to their disconnection in the case of voltage surges.

Jaguar Land Rover (August 2025)

Source: https://cybermonitoringcentre.com/2025/10/22/cyber-monitoring-centre-statement-on-the-jaguar-land-rovercyber-incident-october-2025/

In late August 2025, Jaguar Land Rover experienced a major cyber incident. The incident impacted JLR’s internal IT environment leading to an IT shutdown and a halt in global manufacturing operations, including its major UK plants at Solihull, Halewood, and Wolverhampton. Production lines were halted for several weeks, dealer systems were intermittently unavailable, and suppliers faced cancelled or delayed orders, with uncertainty about future order volumes.

The estimated loss for the event is £1.9 billion, with a range of £1.6 billion to £2.1 billion. However, unlike other cyber events where malware propagated across multiple organisations, the JLR event was concentrated on a single primary victim, with systemic effects arising indirectly through economic rather than IT system interdependencies.

London Heathrow Airport (March 2025)

Source:
https://www.heathrow.com/content/dam/heathrow/web/common/documents/news/Kelly-Review-Report-May-2025.pdf

In the late evening of Thursday 20 March 2025, a fire started at a National Grid 275kV High Voltage electricity substation. This took out all three transformers and the supply to Heathrow's substation was cut off.

The Airport stopped operations approximately 90 minutes after the power outage, and fully restored operations on 22 March.

Significant disruption was experienced by approximately 200,000 passengers, and financial losses were caused to airlines and other businesses within the airport. The timing and extent of reopening depended on restoring connection to its original three electricity supply points as many systems could not switch.

Maersk (May 2017)

Source: https://committees.parliament.uk/writtenevidence/113939/pdf/

When the shipping giant Maersk was hit by the NotPetya ransomware virus in 2017, almost all their software systems including those on-board ships were destroyed. But in accordance with normal procedures, most ships in transit had printed out hard copy documents of the customs and harbour documentation they needed for their next port. So, those ships could continue their current voyages until they were safely docked in port. Other shipping continued with manually prepared paperwork taped to containers.

Lesson: it is possible to operate a minimum viable product to keep the business running.

Marks & Spencer (April 2025)

Source: https://conosco.com/in-the-news/marks-and-spencer-what-it-means

A cyberattack crippled its online retail operations for over six weeks. The breach was first detected in late April and traced to a targeted impersonation campaign linked to the Scattered Spider group, a known ransomware-as-a-service (RaaS) affiliate. The group deployed DragonForce ransomware following a successful social engineering compromise.

The compromise began with a simple but highly effective tactic: impersonation. The breach exploited a procedural weakness, not a misconfigured firewall or unpatched server. Threat actors posing as an M&S employee contacted a third-party provider and convinced them to reset access credentials. The provider granted the request, unknowingly handing over a critical entry point. This allowed the attackers to escalate privileges and move across connected systems.

Once the attackers had access, they deployed DragonForce ransomware to encrypt key digital infrastructure, including customer-facing platforms, backend systems, and internal coordination tools. This halted online shopping entirely and disrupted click-and-collect, delivery fulfilment, and returns management.

Marks & Spencer estimates up to £300 million in lost profit tied directly to the attack. While stores remained operational, online retail; spanning food, clothing, and home, was offline for more than six weeks during key trading periods. Attempts to shift demand back to physical channels led to logistical strain and customer dissatisfaction.

Engineering Resilience for IT Systems
Annex: Public Reports

Meta (October 2021)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Meta -- the parent company of Facebook, Instagram and WhatsApp -- found out that fail-safes are only useful if they don't fail, the hard way, on Oct. 4, 2021.

While performing routine maintenance, a command was issued to assess Facebook's network capacity. Instead, it disconnected all the company's data centres across the globe. According to Meta, this was an event that should have been prevented by their systems' auditing, but a bug in the audit tool allowed the command to go through. All of Meta's services went dark for nearly seven hours. While the damage caused by the outage is difficult to quantify, Facebook lost $47.3 billion in market value during the downtime, according to Bloomberg.

Microsoft Azure outages (2023 and 2024)

Source: https://www.xurrent.com/blog/it-outages

Azure's scale makes it a backbone for thousands of enterprises, which means that the effects of failures cascade far beyond Microsoft's own systems.

An outage in January 2023, was triggered by a faulty router update. Services across multiple continents were affected, disrupting compute, storage, and collaboration tools for hours. The second, in July 2024, was more complex. A distributed denial-of-service (DDoS) attack coincided with a regional power flicker. The combined stress caused a chain reaction that disrupted multiple Azure regions, affecting customers in North America, Europe, and Asia.

The root cause of the January 2023 outage was a straightforward operational misstep due to a network configuration update that escalated without sufficient isolation. The July 2024 incident was a convergence of external and internal stressors. The DDoS attack increased system load at the same time the power event reduced available capacity, and the interplay of those failures magnified the downtime. This is a reminder that large-scale incidents are often the product of multiple smaller failures colliding.

Microsoft's response in 2023 was to roll back the router update and restore services region by region. The 2024 incident required both repelling the DDoS traffic and rebalancing workloads to unaffected regions, all while bringing power systems back to stability. Customers received detailed post-incident reports, though for many, recovery hinged on their own architecture decisions, particularly whether they had secondary hosting arrangements.

Exact financial losses for Azure customers vary, but the scale was significant. High-impact cloud outages have a median annual downtime of 77 hours. For businesses built entirely on Azure services, even a few hours can mean millions in lost revenue and productivity. During both incidents, dependent SaaS platforms and enterprise applications went down, leading to secondary IT outages. For organizations without multi-cloud failover, these events effectively took their core operations offline.

Microsoft 365 Outage (January 2026)

Source: https://medium.com/@ismailkovvuru/microsoft-365-outage-2026-root-cause-forensic-analysis-77712e57b9fc

Microsoft's most critical shared services failed simultaneously on January 22-23, 2026, and automated failover systems couldn't prevent an 8-9 hour global outage.

Why This Matters: This wasn't a single-region failure with clean failover. This was likely a control plane catastrophe that bypassed multiple layers of redundancy, the kind of failure that automated systems cannot self-heal.

The Critical Questions:

- Why did recovery take 8-9 hours when automation should enable sub-hour recovery?
- What failure modes can defeat multi-region redundancy at this scale?
- What should enterprises actually DO with this information?

Key Insight: Even the world's most sophisticated cloud infrastructure can experience prolonged outages. Understanding why automated resilience failed is more valuable than knowing the outage occurred.

NATS (August 2023)

Source: https://www.gov.uk/government/speeches/caa-publication-of-independent-report-into-nats-technical-failure

A major failure of NATS (En Route) Plc's (NERL) flight planning system had wide impact.

An error in the Lookup Table caused the NATS outage in August 2023. Software created a flight plan that included two waypoints with the same name: Deauville in France and Devils Lake in North Dakota. Another software system rejected the flight plan as the exit point from UK airspace (Deauville) was interpreted as Devils Lake, which was not credible. The main flight processing system flagged this as a critical exception error and shut down. The secondary system made the same mistake, leading to shutdown of the entire system. The effect was magnified by delays in restarting the system, with staff not trained to manually enter flight plans. An engineer from the software supplier was needed to resolve the problem.

The CAA has estimated that there were over 700,000 passengers and others who were affected by the failure, often for several days. Costs to airlines were approximately £65m. In addition, substantial costs were incurred by passengers, airports, tour operators, insurers, and others. It is likely that the total cost was in the region of £75m to £100m.

Lesson: Redundant systems sharing a common validation algorithm are not redundant against logic faults. Primary and secondary IBS systems need to consider independently developed or cross-checked validation logic, N-Version Programming, or heartbeat check for the parallel process availability, to prevent common-mode failures from propagating through architecturally separate components.

Lesson: Input validation is not a back-office data quality concern. It is a front-line resilience control. Validation failures should be a monitored, alerted, and SLA-governed event category.

Southwest Airlines (December 2022)

Source: Southwest Airlines (2023). Review of December 2022 Operational Disruption. U.S. Department of Transportation Records.[1]

What happened: A winter storm triggered cascading failures in crew-scheduling software. Southwest cancelled 72% of flights over a week while competitors

[1] See https://www.transportation.gov/sites/dot.gov/files/2023-12/Southwest%20Order%202023-12-11.pdf

operated in single digits. Voice phone calls as a crew-scheduling fallback were overwhelmed by a 9-hour wait time.

Key failures: Decades of IT underinvestment; no tested Degraded Mode Operating Procedures (DMOP) for crew-scheduling failure at scale; manual backup that could not absorb the volume, no saturation scenario in rehearsal to test capacity and delay.

Lesson: A DMOP needs to be designed as a Scalable System of Survival. Rehearsal should include the saturation scenario. Wargaming should be considered to simulate the saturation of manual workarounds, not just the initial switchover.

Rogers Communications (July 2022)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Canadian telecom provider Rogers Communications suffered a massive outage due to a routing issue similar to the 2019 Verizon outage. The outage affected more than 12 million customers nationwide, according to a report published by the Canadian Radio-television and Telecommunications Commission.

Human error was the root cause. While configuring the distribution routers, which are responsible for directing internet traffic, Rogers staff removed a key filter known as the access control list. As a result, all possible routes to the internet ended up passing through the routers of Rogers' core network, ultimately exceeding their capacity and causing them to crash. The outage lasted for nearly a day, with mobile networks, internet and 911 service unavailable during that time.

UK NHS (2024, 2025)

Source: https://www.digitalhealth.net/2025/12/nhs-gp-software-supplier-hit-by-cyber-attack/

NHS patient data has been the target of a number of attacks, eg

- DXS International which provides healthcare technology for 2,000 NHS doctors was subject to a cyber-attack affecting its office servers on 14 December 2025.

- Ransomware group DevMan took credit for the breach. The hackers claim to have stolen 300 gigabytes of data.

- A cyber-attack on Barts Health NHS Trust led to personal patient and staff information being posted on the dark web after a criminal group, known as Clop, exploited a loophole in the Oracle E-business Suite software.

- Pathology supplier Synnovis is contacting NHS organisations which had data stolen and published online following a major cyber-attack in June 2024, which led to a patient death and disrupted services throughout London.

Verizon/BGP (June 2019)

Source: https://www.techtarget.com/whatis/feature/8-largest-IT-outages-in-history

Verizon experienced an outage lasting approximately three hours beginning around 6:30 a.m. on June 24, 2019. The failings of a protocol known as the Border Gateway Protocol (BGP), which is responsible for joining networks together and directing traffic between them, caused all Verizon's internet traffic to be routed through DQE Communications, a small ISP in Pennsylvania.

DQE was using a BGP optimizer from Noction that shared routing information to a customer -- Allegheny Technologies -- before also sharing it with Verizon. Verizon, in turn, sent out this specific routing information to the rest of the internet. Internet traffic was subsequently routed through DQE and Allegheny, unequipped to handle the heavy loads. Besides Cloudflare, major services such as Amazon, Google and Facebook were also disrupted.

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Purpose

This Annex consolidates the Key Deliverables into a single accountability matrix, assigning Responsible, Accountable, Consulted and Informed status against the seven stakeholder roles used throughout. It fulfils the commitment in the Introduction to identify 'the various responsibilities and accountabilities' for resilience deliverables, and gives boards and leadership a single reference for who owns, produces, shapes and consumes each artefact.

The RACI entries are intended to provide a starting point for organisations - each may assign responsibilities and accountabilities differently. The essential point is that only one function can be Responsible.

RACI Key

R	Responsible	Performs the work to produce and maintain the deliverable. More than one role may be Responsible.
A	Accountable	Owns the outcome and answers for it — approves the deliverable. Exactly one Accountable role per deliverable.
C	Consulted	Provides input before completion; two-way communication.
I	Informed	Kept up to date on progress and outcome; one-way communication. For Students / Young Professionals, 'I' denotes a recommended study artefact per the chapter's foundational-concepts takeaway.
A/R	Accountable and Responsible	The same role both owns and produces the deliverable — common for function-owned artefacts (e.g. risk models, ISMS).
–	No assigned involvement	The role has no defined part in producing or governing this deliverable.

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Role Key

CEO / NED	Chief Executive Officers and Non-Executive Directors — strategic, board-level governance
CIO / CISO	Chief Information Officers and Chief Information Security Officers — technical governance
BC & Risk Mgr	Business Continuity and Risk Managers — operational resilience and continuity
SRE / Avail. Team	Site Reliability Engineers and Availability Teams — engineering, service management and operations
Enterprise Architect	Enterprise Architects — architecture governance
Supply Chain / Vendor Mgr	Supply Chain and Vendor Managers — third-party and vendor governance
3rd-Yr Student / Young Prof.	3rd Year Students and Young Professionals — foundational learning audience (never R, A or C)

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Conventions Applied

1. Exactly one Accountable (A) role is assigned per deliverable; where a single function both owns and produces an artefact, A/R is used.
2. Assignments are derived directly from the accountabilities stated in each chapter's role-specific Key Takeaways boxes; where a takeaway names an activity (e.g. "build and test a Kill Switch", "ensure DR exercises are carried out"), the matrix reflects it.
3. Board (CEO/NED) accountability is reserved for deliverables the regulatory frameworks place with the governing body: IBS declaration and Impact Tolerance approval (FCA/PRA SS1/21), AI adoption as governed change, culture and governance charters, and pre-delegated kill-switch authority.
4. The Introduction (Chapter 0) defines the Handbook's framing and commits to this Annex; it contributes no deliverables of its own and therefore has no matrix rows.
5. Organisational titles vary; roles here are functional. In smaller organisations one person may hold several columns — the discipline of a single named A per deliverable still applies.

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Chapter 1 — Why Resilience Matters

Ref /Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
1.1 IBS Register *Inventory of Important Business Services with named owners; annual review*	A	C	R	I	C	C	–
1.2 Impact Tolerance Statements *Per-IBS maximum tolerable disruption, Board-approved, traceable to RTO/RPO/MTTR*	A	C	R	I	C	I	–
1.3 Availability Profile Map *Per-IBS assessment against the four degradation modes*	I	A	C	R	C	C	I
1.4 Regulatory Applicability Assessment *Determination of applicable instruments (CSRB/NIS, DORA, FCA/PRA) and obligations*	I	A	R	–	–	C	–
1.5 Standards Conformance Baseline *Mapping to ISO 22301, ISO/IEC 25010, 5055, ISO 31000, with gaps*	I	A	R	C	C	–	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Ref /Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
1.6 Architectural Principles Statement *Commitment to graceful degradation, blast-radius minimisation, observability*	I	A	I	C	R	–	I
1.7 AI Adoption Policy *AI treated as software; human-in-the-loop; manual fallbacks; assigned accountability*	A	R	C	I	C	C	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Chapter 2 — Making Resilience A Reality

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
2.1 Resilience Governance Charter *Three-tier governance model; designated Board Member for Resilience; BCSC charter*	A	C	R	I	I	I	–
2.2 RACI Matrix for Resilience *Roles assigned across IT, BC, finance and suppliers; lines of defence defined*	A	C	R	C	C	C	–
2.3 Risk and Cost Model *Probability × impact translated into budget decisions (NIS metrics, Impact Tolerance logic)*	A	C	R	C	C	–	–
2.4 Just Culture Policy *Blameless post-incident learning, safe-space reporting, near-miss capture*	A	R	C	C	I	I	I
2.5 Skills and Competency Plan *SfIA-aligned resilience competencies in role profiles; AI-enabled operations baseline*	I	A/R	C	C	C	–	I

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
2.6 Communications Toolkit *Simulation/wargame materials, prepacked post-incident workshop, cost/time-to-fail visual*	I	A	R	I	–	–	–
2.7 AI Enablement Register *Documented AI use cases with assigned human accountability*	I	A	I	C	R	C	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Chapter 3 Supply Chain and Vendor Management for Resilience

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
3.1 Vendor and Third-Party Register *Inventory of third-party arrangements mapped to the IBS they support; for FS, maintained as the DORA Register of Information*	I	A	C	I	C	R	I
3.2 SBOM Registry *SBOM for all production systems incl. AI-generated code; updated per deployment, reviewed quarterly*	–	A	C	R	C	C	I
3.3 Supplier Contract Resilience Schedule *Per-contract notification, maintenance windows, staged rollout, rollback/remediation SLAs; gaps escalated to Board (absorbs 5.7 SLA recovery obligations)*	I	C	C	I	–	A/R	–
3.4 Concentration Risk Assessment *Provider-diversity position per IBS; reviewed at each renewal and after material architecture change*	I	C	A/R	I	R	C	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
3.5 Third-Party Dependency Test Results *Synthetic transaction, contract test and injected vendor-outage results per IBS; feeds SLM and SLA negotiation*	–	A	C	R	C	C	–
3.6 Supplier Contact and Escalation Schedule *Named contacts and out-of-hours escalation paths per critical supplier; validated in incident simulations*	–	C	C	I	–	A/R	–
3.7 Exit and Substitution Plans *Documented substitution path, data egress method and costed transition timeline per critical supplier*	I	C	R	–	C	A	–
3.8 AI Asset Register *AI-generated code, AI-assisted applications and agentic processes, risk-tiered with named ownership; maintained alongside the SBOM*	I	A	I	R	C	C	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Chapter 4 — Planning for Improved Availability

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
4.1 SRE-to-Stakeholder Contract *Agreed SLOs/SLIs, RTO/RPO mapping, escalation paths linked to Impact Tolerances*	I	A	C	R	C	I	I
4.2 Accurate Configuration Baseline *Current CMDB / automated discovery: inventory, dependencies, drift reports*	–	A	I	R	C	C	I
4.3 Infrastructure & ADM Visualisation *Logical/physical topology, Critical Path per IBS, SPOF register, recovery sequencing*	–	A	C	C	R	I	I
4.4 FMEA & Deficiency Log *Service-focused failure analysis, architectural decision record, accepted-risk register*	I	C	A	C	R	C	–
4.5 SLO & Error-Budget Framework *Per-IBS SLOs, multi-burn-rate alerting, error budgets linked to Impact Tolerances*	I	A	C	R	I	–	I
4.6 ISMS Artefacts (ISO/IEC 27001) *Scope, risk method, risk register, Statement of Applicability, audit actions*	I	A	R	I	I	C	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
4.7 SfIA-Aligned Resilience Competency Map *Availability/recovery skills in role profiles; capability gaps and remediation*	I	A	R	C	C	–	I
4.8 Validated Recovery Architecture *4TO/RPO-matched patterns, IaC templates, PITR and immutable backups, proven via game days*	I	A	C	R	R	C	I
4.9 NIST-aligned contingency and incident response artefacts *CSF outcomes; SP 800-53 CP/IR/CM/CA*	I	A	R	I	I	C	–

Chapter 5 — Service Delivery

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
5.1 ART Test Results *Automated Recovery Testing results against each IBS, reported to the Board*	I	A	C	R	C	–	–
5.2 DMOP Set *Rehearsed, offline-accessible degraded-mode procedures per IBS, named-role owned*	I	A	R	C	I	C	–
5.3 Data Validation Standards *Per-IBS validation rules as acceptance-testing criteria*	–	A	I	R	C	–	–
5.4 Deployment Runbooks *Staged deployment procedures per IBS-supporting system, with automated rollback*	–	A	I	R	C	I	–
5.5 SRE Resilience Scorecards *Availability KPIs, near-miss trends, RTO/RPO test results — quarterly to Board*	I	A	C	R	I	–	–
5.6 CSRB Incident Reporting Readiness Pack *Templates and tooling for 24-hour initial and 72-hour detailed incident reports*	I	A	R	C	–	–	–
5.7 Resilience Scorecard (Enhanced) *Impact Tolerance Headroom and Recovery Confidence Scores, quarterly to Board*	I	A	C	R	I	–	–

Engineering Resilience for IT systems:
Annex - RACI for Resilience Deliverables

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
6.1 Recovery Playbooks *Per-system recovery steps and historic-incident resolutions; automated restarts where safe*	–	A	C	R	C	I	I
6.2 Incident & Major Incident Records *Classified, time-stamped logs: business impact, outage duration, actual recovery time*	I	A	C	R	I	–	–
6.3 DR Exercise Schedule & Reports *At least annual DR tests with gap analysis and remediation actions*	I	C	A	R	C	C	–
6.4 Post-Incident Reports *Nine-section template with tracked owners, priorities and dates*	I	A	C	R	C	C	I
6.5 Root-Cause Register *Agreed root causes (5 Whys / Fishbone) and contributing factors; recurring patterns*	–	A	C	R	C	–	I

Ref / Deliverable	CEO / NED	CIO / CISO	BC & Risk Mgr	SRE / Avail. Team	Enterprise Architect	Supply Chain / Vendor Mgr	3rd-Yr Student / Young Prof.
6.6 Communications Pack *Holding pages, customer update templates, out-of-band communication channels*	C	A	R	I	–	–	–
6.7 Kill-Switch Authority Matrix *Pre-agreed terms of reference: who may disable key systems, when, under whose authority*	A	R	C	C	C	–	I

Engineering Resilience for IT Systems:
Annex – Review Guide

This annex provides an overview of the chapters for each of the roles that we considered while writing the handbook. Here we summarise what might be expected as deliverables for each role and these deliverables are linked to those in the Annex RACI for Resilience Deliverables. The detail in this Annex is greater than that in the RACI Annex because delivery on RACI responsibilities will often require additional information or reporting. One use of this Annex is for the reader to self-test that they understand the explicit and implicit requirements for their role with respect to each of the Chapters. Another use is to examine what is expected of other roles than their own so the reader might understand what to expect others will know. The deliverables are arranged by chapter for each of the following roles:

Engineering Resilience for IT Systems:
Annex – Review Guide

CTO family: Technical governance and architecture accountabilities:

Chapter 1: Why resilience matters

- Create a mapping for every IBS to its supporting systems, dependencies, and data flows, you cannot protect what you have not mapped.
- Translate Impact Tolerances for every IBS and underpinning systems into engineering targets; Recovery Time Objective (RTO), Recovery Point Objective (RPO), and Mean Time To Recover (MTTR) for each service and its data.
- Ensure internal operating level agreements exist for networks, server infrastructure, storage database services and telephony, or supply contracts are built around the impact tolerance requirements.
- • Align Financial Services obligations explicitly: DORA for ICT risk management, incident reporting, and third-party risk; FCA/PRA UK Operational Resilience for IBS and Impact Tolerances.

Chapter 2: Making resilience a reality

- Translate Board strategy into operational governance: establish a "Defend as One" model that crosses silos rather than leaving teams with overlapping, conflicting remits.
- Establish the common language e.g. IBS and Impact Tolerances, that lets IT, business continuity, finance, and risk operate under a unified approach.
- Implement the lines-of-defence model: day-to-day controls (ISO 27001, ISO 22301), internal audit, management controls/remediation, and independent external audit.
- Build the channels to engage the C-suite on business priorities and risk, and communicate technical exposure in business terms, for example a cost / time-to-fail curve.
- Lead on the skills gap: add resilience and availability competencies to IT role profiles, create the relevant posts within the organization aligned to the Skills Framework for the Information Age (SfIA), and build the blended skill set for AI-enabled operations.
- Govern AI deployment across documentation, manuals, testing, release management and AIOps, capturing the uplift while assigning clear human accountability for AI outcomes.

Chapter 3: Supply chain and vendor management

- Mandate SBOM disclosure and staged rollout for all third-party software entering production; no third-party update enters production without validation in a mirrored environment
- Extend the CMDB and dependency maps to cover vendor services, their APIs, and sub-dependencies on the critical path of each IBS
- Ensure supplier health monitoring feeds the same dashboards as internal telemetry, so vendor degradation is detected as a service event
- Treat AI-generated code and agentic processes as a supply chain channel: inventory in the SBOM/AI Asset Register, gate, validate, and define non-agentic fallbacks
- Align third-party change management with DORA Article 25 and FCA/PRA SS1/21 obligations

Chapter 4: Planning for improved availability

- Maintain a current Change Management Database (CMDB) and Application Dependency Map covering on-premises and cloud, with the Critical Path identified for each IBS.
- Commission a structured architectural review that formally validates RTO/RPO are technically achievable for every critical service.
- Establish a Site Reliability Engineering (SRE) or platform function with cross-functional authority over both design and operations; resource and tool it against IBS criticality (ISO/IEC 27001 Clause 7).
- Adopt data-driven SLOs and error budgets, and link internal error budgets directly to externally committed Impact Tolerances.
- Align resilience practice with ITIL Change Enablement and Incident Management to meet FCA and PRA operational resilience expectations.
- Govern AI used in planning as software and monitor model drift and automation bias. Retaining human oversight of AI-generated runbooks prior to implementation.

Chapter 5: Service Delivery

- Maintain a current Change Management Database (CMDB) and ensure all IBS dependencies are mapped to infrastructure components (Chapter 3 prerequisite)
- Mandate SBOM disclosure and staged rollout for all third-party software entering production
- Establish and test automated rollback procedures for all production deployment pipelines
- Ensure monitoring covers the four Golden Signals (Latency, Traffic, Errors, Saturation) across all IBS-supporting systems
- Establish Zero Trust Architecture and micro-segmentation to limit blast radius
- Align ICT change management (Financial Services) with DORA Article 9 and FCA/PRA SS1/21 obligations
- Ensure AI/ML components embedded in IBSs have non-AI fallback paths defined in the DMOP

Chapter 6: Response, recovery and learning

- Ensure everyone involved in recovery understands their role and authority before an incident : clear roles, on-call rotas, and out-of-band communication channels.
- Maintain a Cyber Security team readiness process: current threat intelligence, defined criminal-demand and legal-engagement steps, and boardroom reporting.
- Run regular recovery simulations for key systems to flush out gaps and build team confidence and readiness.
- Confirm recovery depends on accessible essentials such as emote access, admin credentials, MFA, and third-party contacts are held securely and available out-of-hours.
- Establish severity ratings (for example Bronze/Silver/Gold) mapped to IBS and regulator thresholds, with a defined path into Major Incident Management.
- Ensure detection covers the failure modes that evade simple health checks: high latency (“slow is down”), elevated error rates, and corrupted-but-successful responses.

Engineering Resilience for IT Systems: Annex – Review Guide

Supply Chain and Vendor Managers

Chapter 2: Making resilience a reality

- Establish procurement and audit practices aligned to the organisation's resilience objectives (e.g. high availability for IBS's) and admit only suppliers who can demonstrably help meet them.
- Build supplier obligations into contracts; service level management, notification, audit rights, and alignment to your Impact Tolerances and IBS register.
- Treat critical third-party and supply-chain risk as a top-tier annual risk priority requiring active management, not just periodic review.
- Recognise the contractual challenge that multi-party supply and maintenance have for incident response, delineate and agree responsibilities before the incident, not during it.
- Maintain external and industry networks to share intelligence on suppliers and products as the supply chain grows more complex.
- Apply AI-assisted contract lifecycle management to maintain visibility and enforce resilience clauses across the vendor portfolio.

Chapter 3: Supply chain and vendor management

- Aligned with DORA Article 25; ISO/IEC 19770-2
- Require SBOM disclosure as a contractual term in all software procurement agreements
- Mandate staged rollout and pre-production validation as SLA obligations for all supplier-driven updates, including content and signature updates, not only version releases
- Define rollback and remediation SLAs in all third-party contracts
- Maintain an up-to-date supplier contact schedule including out-of-hours support escalation paths
- Conduct annual resilience reviews of critical third-party dependencies aligned to the organisation's IBS register
  Escalate to the Board any supplier contract that does not include SBOM, staged deployment, and rollback obligations.

Chapter 4: Planning for Availability

- Assess concentration risk during architecture planning, avoid over-dependence on a single cloud provider, supplier, or technology.
- Require provider diversity and multi-region options where an IBS's Impact Tolerance justifies the cost.
- Ensure recovery-site and failover arrangements (GSLB, warm-standby regions) are contractually available and tested, not assumed.
- Align observability and platform tooling procurement to IBS criticality and integration need, including coverage of metrics, logs, and traces.
- Build recovery and RTO/RPO expectations into third-party SLAs at design time, before services enter production.
- Record third-party dependencies in the configuration baseline so supplier risk is visible on the Critical Path.

Chapter 5: Service Delivery

- Apply the Chapter 2.5 vendor controls at the service delivery boundary; SBOM contractual disclosure, staged rollout and rollback SLAs for all supplier-driven updates (including content and signature updates), supplier contact and escalation schedules, and annual resilience reviews aligned to the IBS register
- Escalate to the Board any supplier contract that does not include SBOM, staged deployment, and rollback obligations, a standing agenda item, not a filed document (Chapter 2.5, §2.5.4.2)

Chapter 6: Response, recovery and learning

- Ensure third-party support is covered by SLAs with defined out-of-hours availability; mobilise any unarranged support as a priority, as it is a common source of delay.
- Maintain a current schedule of third-party providers' availability, escalation paths, and technical contacts, including SaaS components.
- Confirm suppliers' recovery obligations: rollback support, incident response times, and reporting, make sure these are contractually defined.
- Include supplier and customer-facing dependencies in incident diagnosis planning, for example, ensure the supplier-manager role is reachable out-of-hours.
- Review critical third-party recovery performance after material incidents and feed findings into contract renewals.

- Align vendor arrangements with regulatory recovery obligations — including the DORA annual Digital Recovery exercise for financial services.

Engineering Resilience for IT Systems: Annex – Review Guide

Enterprise Architects

Chapter 1: Why resilience matters

- Treat resilience as a named non-functional requirement, not an emergent property, include in architecture models, business architecture, high level and low level designs specify; availability, recoverability, and performance targets per service.
- Design to RPO and RTO; data durability (how much data can be lost) and recovery time (how quickly to return online) are distinct objectives needing distinct controls.
- Assess concentration and supply-chain risk, evergreen SaaS and global software supply chains introduce dependencies that are neither visible nor under your control.
- Apply ISO/IEC 25010 quality characteristics. Reliability, with its Availability, Fault-Tolerance, and Recoverability sub-characteristics as design criteria, and ISO/IEC 5055 to find structural weakness before deployment.
- Account for legacy: older systems (the SS7 example) embed assumptions including absent security that constrain modern resilience; document and compartmentalise them.
- Build observability and dependency mapping into the architecture from the outset, not as a retrofit.

Chapter 2: Making resilience a reality

- Embed “Secure and Resilient by Design”: establish common architectural patterns and profiles so new services are resilient from inception. (expanded in Chapter 3)
- Use the enterprise architecture map to expose interdependencies and focus resilience effort on Important Business Services and their supporting systems and infrastructure.
- Design compartmentalisation to localise the impact of failures and limit access for bad actors.
- Specify resilience and availability as non-functional acceptance criteria in service design, not as later additions.
- Plan legacy modernisation realistically; rewriting legacy applications, including AI-assisted re-coding, can be part of a plan but is not a complete plan; resilience must still be engineered in.
- Apply AI-driven mapping tools (for example for estate configuration, or network sensing) to regenerate lost documentation and surface code vulnerabilities, while validating their output.

Engineering Resilience for IT Systems:
Annex – Review Guide

Chapter 3: Supply chain and vendor management

- Assess concentration and supply chain risk in every design: evergreen SaaS and global software supply chains introduce dependencies that are neither visible nor under your control (Chapter 1, §1.6.2)
- Design isolation, failover, and circuit-breaker boundaries around third-party services so a vendor failure is a degraded mode, not an outage
- Weigh buy-or-build decisions with resilience consequences priced in: buying imports another organisation's release discipline into your critical path
- Keep the critical path per IBS current as vendor services are added — a supplier that appears on every critical path is a single point of failure.

Chapter 4: Planning for Availability

- Provide a clear, stakeholder-specific architectural view: logical and physical topology, microservices, and critical paths, maintain it as a living document.
- Design in the core resilience patterns: redundancy across failure domains, loose coupling, defence in depth, and graceful degradation.
- Apply recovery patterns deliberately: pilot light, warm standby, or multi-site active-active, appropriately chosen against each IBS's RTO/RPO and budget.
- Protect the data layer: asynchronous replication, event sourcing, WORM immutable backups, and point-in-time recovery (PITR) against corruption and ransomware.
- Use quorum or consensus mechanisms (Paxos, Raft) or Triple Modular Redundancy for critical state to prevent split-brain.
- Treat environments as reproducible code (IaC) so a recovery site is an identical twin, avoiding configuration drift.

Chapter 5: Service Delivery

- Require resilience as a named, explicit non-functional requirement in every service design
- Implement circuit breaker patterns and bulkhead compartmentalisation in all new IBS-supporting architectures
- Apply Zero Trust principles and micro-segmentation to limit blast radius across service boundaries
- Ensure data validation logic; format, range, presence, lookup, check digit, specified as acceptance criteria before go-live

- Maintain deployment pipelines with automated rollback capability as standard
- Validate that every new or changed IBS has a corresponding DMOP before it enters production.

Chapter 6: Response, recovery and learning

- Ensure third-party support is covered by SLAs with defined out-of-hours availability; mobilise any unarranged support as a priority, as it is a common source of delay.
- Maintain a current schedule of third-party providers' availability, escalation paths, and technical contacts, including SaaS components.
- Confirm suppliers' recovery obligations : rollback support, incident response times, and reporting, make sure these are contractually defined.
- Include supplier and customer-facing dependencies in incident diagnosis planning, for example, ensure the supplier-manager role is reachable out-of-hours.
- Review critical third-party recovery performance after material incidents and feed findings into contract renewals.
- Align vendor arrangements with regulatory recovery obligations — including the DORA annual Digital Recovery exercise for financial services.

Engineering Resilience for IT Systems: Annex – Review Guide

Site Reliability Engineers and Availability Teams

Chapter 1: Why resilience matters

- Implement the high-availability patterns named in Chapter 3: active-active across regions, blast-radius minimisation via microservices and modular design, auto-scaling, and health checks that kill and restart unhealthy instances.
- Treat latency as contagious; a slow downstream dependency triggers timeouts and retry storms that present to users as a hard outage; use circuit breakers to fail fast.
- Engineer for graceful degradation, when one component fails, the whole service must not.
- Guard against the silent killer: detect corrupted or stale results that still return success codes, because standard monitoring reports the system as healthy while bad data propagates.
- Build the data capture pipeline — data capture tools, logs, metrics, and traces — to alert before a local bottleneck becomes a total outage, tie in with incident and problem management.
- Pilot AI in low-risk monitoring before automating remediation, and monitor for drift, hallucination, and alert floods that can make AI an outage multiplier.

Chapter 2: Making resilience a reality

- Maintain robust continuous threat detection and monitoring (for example via an incident Operations Centre or a Security Operations Centre) to anticipate and detect events early. (expanded in Chapter 4)
- Ensure failure mode documentation is available and refreshed, systems change, documentation must match.
- Contribute the operational and observability skills; SLIs/SLOs, telemetry pipelines, post-incident reviews that underpin an AI-enabled operations capability.
- Build and test Kill Switch for each IBS: a clear, authorised means for operations staff to disable a failing system fast, the difference between Knight Capital's collapse and Co-op's contained incident.
- Support human-centred recovery; ensure operators are trained in the correct recovery response, because untrained roll-back actions have turned minor incidents into multi-day outages. (expanded in Chapter 5)
- Capture near-misses and intermittent faults systematically, treat them as early-warning signal, not noise and determine patterns that need escalation.

- Use AI to filter alerts and automate routine decisions, set up alerts to allow human confirmation for high-consequence actions such as failover.

Chapter 3: Supply chain and vendor management

- Implement synthetic transaction testing against vendor APIs and contract tests to validate integration assumptions
- Inject vendor slowness and outage conditions to validate timeout and circuit-breaker logic around every third-party dependency
- Monitor supply chain outages (vendor status pages, external services such as DownDetector) alongside internal telemetry
- Validate rollback procedures for all automated third-party update mechanisms as part of every ART exercise
- Capture vendor-attributed transient faults as near-misses and feed them into the supplier review evidence base.

Chapter 4: Planning for Availability

- Instrument full-stack observability across the three pillars — metrics, logs, and traces, with synchronised time (NTP) for reliable forensic timelines in reporting, security and audit logs.
- Adopt multi-window, multi-burn-rate alerting on user-impacting symptoms rather than static thresholds, to catch acute incidents while avoiding alert fatigue.
- Implement transient-failure controls: exponential backoff with jitter, circuit breakers, and bulkheads to prevent retry storms and cascading failure.
- Automate drift detection where possible to alert when a resource appears without the DR tags or replication controls that policy requires.
- Run controlled chaos experiments and automated restore tests into isolated sandboxes to prove recovery, not assume it.
- Externalise state and design for automated node replacement (statelessness, Infrastructure as Code) so failed nodes rebuild without loss of session data.

Chapter 5: Service Delivery

- Run Chaos Engineering experiments and ART exercises regularly against IBS-supporting systems
- Validate rollback procedures as part of every ART — not as a separate exercise
- Capture every transient and intermittent fault as a near-miss to determine if there is a pattern requiring logging and investigation
- Ensure provisioning is held at ≤80% of peak capacity for all IBS-supporting services
- Implement and test load shedding automation; confirm thresholds are documented in the DMOP
- Monitor AI/ML model drift; ensure non-AI fallbacks are available, documented, and tested.

Chapter 6: Response, recovery and learning

- Create and maintain recovery playbooks for every valued system, capturing historic incidents and proven resolution steps, possibly automate known-good restarts where safe.
- Keep monitoring and alerting current; where possible, alert on early indicators so degradation is caught before it becomes a full outage.
- Maintain error logs and interpreted error-code glossaries that give clear, actionable failure information.
- Provide a change-activity view, e.g. what changed in the last 24 hours, as a first-line diagnostic input.
- Support diagnosis with an accurate, end-to-end architecture and dependency view (Chapter 4) accessible at incident time.
- Feed lessons from every retro back into playbooks, monitoring thresholds, and the architectural backlog.

Engineering Resilience for IT Systems: Annex – Review Guide

Business Continuity and Risk Managers

Chapter 1: Why resilience matters

- Create a common language between the Board, IT, and risk, aligned to severe-but-plausible scenario testing, IBS and Impact Tolerance should be understood at all levels.
- Express IT failure as user and regulatory impact using the NIS Framework metrics; Availability (lost user-hours), Integrity, Risk to Public Safety, and Material Damage.
- Apply a consistent IBS-definition discipline (for example the six-point PR01–PR06 checklist) so services are defined comparably and tested end-to-end, including shared and third-party dependencies.
- Direct continuity planning at change and supply-chain risk, not only at malicious actors.
- Maintain risk registers that distinguish risk assumptions, constraints, and tolerances; quantify exposure using stress testing and Value at Risk where the service permits.
- Seek to build continuity certification in ISO 22301 and align organisational resilience to BS 65000 and ISO 22316.

Chapter 2: Making resilience a reality

- Own the tactical tier: Business Continuity Teams, plan administrators, and the BCI six-element lifecycle and also for managing critical incidents.
- Maintain standardised, offline-accessible downtime procedures that let critical functions continue during an outage, maintaining where appropriate the ‘Paper Trail’ principle, as Maersk demonstrated during NotPetya.
- Quantify risk for budget decisions by combining probability and impact, using NIS Framework metrics for user impact and Impact Tolerance logic for the cost of disruption.
- Operate the “thinking the unthinkable” discipline, with pre-mortems and What-If workshops for severe-but-plausible and catastrophic scenarios.
- Rehearse, do not merely document: regular dry runs must test the full response cycle, including restoration of normal service.
- Distinguish risk assumptions, constraints, and tolerances; track Value at Risk and stress-testing results as standing inputs to the continuity plan and risk logs.

Engineering Resilience for IT Systems:
Annex – Review Guide

Chapter 3: Supply chain and vendor management

- Ensure third-party SLAs include notification requirements, maintenance windows, rollback obligations, and support availability for supplier-driven updates
- Include vendor outage scenarios in continuity plans and exercises: confirm RTO/RPO validity if a critical vendor is unavailable for four hours
- Review SBOM completeness quarterly as part of the resilience governance cycle
- Maintain the concentration risk assessment and the exit/substitution plan for each critical supplier as living risk artefacts
- Direct continuity planning at change and supply chain risk, not only at malicious actors (Chapter 1, §1.6.5)

Chapter 4: Planning for improved availability

- Ground FMEA on business services, not components; maintain a documented record of architectural deficiencies and single points of failure.
- Maintain an IBS register with mapped dependencies and a validated recovery path for each service.
- Confirm recovery objectives (RTO/RPO) are defined per IBS and matched to an economically realistic recovery pattern, backup-and-restore, pilot light, warm standby, or active-active.
- Operate a deficiency log and architectural decision record so accepted risks and unresolved weaknesses stay visible rather than becoming institutional folklore.
- Organize game days and restore testing to validate recovery paths before they are relied upon (aligned with ISO 22301).
- Keep the resilience view current for every significant project/IBS updates it before go-live, to prevent architecture drift.

Chapter 5: Service Delivery

- Build and rehearse DMOPs for each IBS; confirm they are offline-accessible and assigned to a named role
  Include DMOP invocation in every ART exercise and annual wargame
- Operate a Near-Miss Register, aligned with ITIL problem management and use it to drive the preventive action cycle
- Enforce data validation rules as resilience controls — include validation failure rates in resilience monitoring dashboards

- Review SBOM completeness quarterly as part of the resilience governance cycle
- Ensure third-party SLAs include notification requirements, maintenance windows, rollback obligations, and support availability for supplier-driven updates.

Chapter 6: Response, recovery and learning

- Maintain clarity on which applications are IBS, each with a plan for recovery in the event of a full outage.
- Hold and govern a system of record for all disaster recovery documents; ensure DR exercises are carried out regularly, at least annually where a regulatory requirement, and for any material new component.
- Ensure the BC function provides recoverable backups, feasible degraded-mode operation (DMOP, Chapter 5), and the ability to resume critical functions once IT is restored.
- Prove restoration, not just backup existence, where possible trial restores into isolated environments, warm-standby go-live trials, and database restore validation.
- Coordinate the crisis hand-off: when a major incident becomes a crisis, ensure the Crisis Management Team owns stakeholder management while recovery continues.
- Track post-incident action items to closure with owners, priorities, and dates, consider a target of for example within 90 days.

Engineering Resilience for IT Systems: Annex – Review Guide

3rd year students and young professionals

Chapter 2: Making resilience a reality

- Learn that resilience is an organisational discipline driven by business operations with board level governance. Ownership, culture and human response determine outcomes as much as technology does.
- Understand RACI: Responsible, Accountable, Consulted, Informed, and the principle that the Board is accountable while operations is responsible.
- Study the human-centred techniques: Paper Trail (Maersk), Kill Switch (Knight Capital versus Co-op), and "thinking the unthinkable", low-cost preparations that change disaster outcomes (discussed further in Chapter 5).
- Recognise that a Just Culture — blameless learning over firefighting heroics — is what lets organisations learn from failure; psychological safety is an engineering asset.
- Build the emerging competencies: SfIA-aligned resilience and availability skills, and the blended operations, data, and automation skills of AI-enabled operations teams.
- Develop the rare combination employers value — the judgement to do the right thing under pressure, and the ability to move between whole-system perspective and implementation detail.

Chapter 3: Supply chain and vendor management

- Learn the anatomy of the software supply chain: source code, open-source dependencies, CI/CD pipelines, build systems, artifact registries, cloud runtimes, and third-party services and why it changes daily
- Study the CrowdStrike (2024) and Cloudflare (2025) incidents as supply chain case studies: in both, the customer organisations had no control over the change that took them down
- Understand the SBOM (ISO/IEC 19770-2); inventory-driven governance is one of the most transferable disciplines in enterprise technology
- Recognise that vendor management is a resilience discipline, not an administrative one: contracts, staging gates, and exit plans are engineering controls expressed in commercial form
- Third-party risk management is a growth profession: DORA, NIS2, and the UK CSRB have made supply chain resilience a named regulatory obligation across sectors

Engineering Resilience for IT Systems: Annex – Review Guide

Chapter 4: Planning for Availability

- Learn to read and build dependency maps — knowing the baseline is the quickest way to prevent or resolve an outage.
- Understand RTO and RPO, and how they drive the choice between backup-and-restore, pilot light, warm standby, and active-active.
- Study SLOs, error budgets, and burn rates: they translate engineering telemetry into board-level resilience commitments.
- Master the core resilience patterns: circuit breakers, bulkheads, backoff with jitter, quorum consensus, these recur across every distributed system.
- Explore Site Reliability Engineering as a discipline at the intersection of software engineering and operational resilience; demand is growing rapidly.
- Consider Infrastructure as Code, as reproducible environments are the backbone of predictable recovery.

Chapter 5: Service Delivery

- Understand that most production failures are caused by managed change; updates, patches, configuration drift, not dramatic external events. Controlling the introduction of change is the core operational discipline.
- Study the CrowdStrike (2024) and NATS (2023) case studies: they illustrate how both update risk and data validation failure manifest at scale, and what controls would have changed the outcome.
- Learn the difference between functional and non-functional requirements. Writing clear, testable NFRs; for availability, recoverability, and performance is one of the most practically valuable skills in enterprise technology.
- Explore Site Reliability Engineering (SRE) as a discipline: it sits at the intersection of software engineering and operational resilience, and demand for professionals who bridge both is growing rapidly.
- Chaos Engineering is an emerging specialism: organisations are building dedicated chaos engineering functions, and engineers who understand adversarial testing will have a material career advantage.

Chapter 6: Response, recovery and learning

- Understand that clear error logging, precisely what is impacted as this is what makes fast diagnosis and recovery possible.

Engineering Resilience for IT Systems:
Annex – Review Guide

- Design code and applications that fail gracefully, containing impact rather than causing a full outage.
- Learn root-cause techniques such as; the 5 Whys and Fishbone (Ishikawa) analysis and why “human error” is never the terminal answer.
- Study contrasting cases: Knight Capital (no kill switch, ~$460m loss) and Co-op versus M&S (fast isolation limited the damage) — controls change outcomes.
- Understand the blameless post-incident culture: the goal is systemic fix, not blame, and openness surfaces the information that speeds recovery.
- Explore where AI assists recovery; automated restarts, log analysis, and guiding the human through recovery steps while keeping human oversight.

Engineering Resilience for IT systems: Annex - Standards

Why standards are important

Standards provide a basis for sharing expectations about a product, process or service. They provide a common language, essential for instance in supply chains. They can also be the basis of certification, applied to products, processes, or people.

Standards Bodies and Standards Development

- ISO (International Standards Organisation) standards are often the basis of national standards e.g. BSI (British Standards Institute), ANSI (American National Standards Institute). National standards are voluntary, though legislation can incorporate them.[1]
- IEC (International Electrotechnical Commission) prepares and publishes international standards for all electrical, electronic and related technologies. The IEC also manages global certification for equipment, systems and components.
- ISO and IEC often work together and publish many joint standards.
- British Standards (BS) are national standards developed by the BSI Group, the UK's national standards body, to set quality benchmarks for goods and services and promote their adoption.
- ISO/WD stands for "ISO Working Draft," which is a preliminary version of a standard that is under development by the International Organization for Standardization (ISO). It represents the early stages of the standardization process before it is finalized and published.
- ISO/DIS stands for Draft International Standard, which is a stage in the development of an ISO standard where the draft is circulated for approval by member countries before it becomes a formal standard.
- ISO/FDIS stands for Final Draft International Standard, which is the penultimate stage in the development of an ISO standard before its official publication. The FDIS is submitted for a final approval vote by ISO member bodies, and once approved, the standard is formally published within 60 days.

Other Authoritative Sources

- BCS (The Chartered Institute for IT): Availability: Resilience by Design (BCS IT Leaders Forum Availability Working Group).
- IET (Institution of Engineering and Technology): Engineering a more resilient world (IET Resilience Working Group) and relevant IET codes of practice/guidance publications as applicable.

[1] Certifiable standards contain mandatory requirements which can be audited whereas non-certifiable standards provide guidelines and best practices.

- ISA/IEC: ISA/IEC 62443 series for industrial automation and control systems (IACS/OT) security, including risk assessment for system design and the use of zones and conduits for segmentation.
- NIST: Cybersecurity Framework (CSF) 2.0; SP 800-53 Rev. 5 Security and Privacy Controls for Information Systems and Organizations; SP 800-160 Vol. 2 Rev. 1 Developing Cyber-Resilient Systems.

Alignment for governance assurance

ISO 27000

- Use Business Impact Analysis (BIA) and an ISMS risk assessment (ISO/IEC 27001 Clause 6) to justify prioritisation decisions. Record selected controls and rationale in the Statement of Applicability (SoA) to demonstrate that resilience and recovery decisions are risk-based and auditable.
- Maintain documented assurance artefacts (ISO/IEC 27001 Clauses 7 and 9): risk register, incident response plan, backup and recovery strategy, test results, internal audit outcomes, and management review actions.

Software product certification

Software product certification[2] is usually aligned with international ISO/IEC standards or local standards such as BSI.

- ISO/IEC 19770-2:2015[3] establishes specifications for tagging software to optimize its identification and management, for instance in a SBOM.
- The dominant model for software and system product quality is ISO/IEC 25010-2. This defines eight quality characteristics, each with sub-headings. One of the eight quality headings is Reliability. Its sub-headings include Availability, Fault-Tolerance, Recoverability. Software code quality is not covered.
- ISO/IEC 5055:2021[4] measures the internal structure of a software product on five business-critical factors: Security, Reliability, Performance Efficiency, and Maintainability. These are the factors that determine how trustworthy, dependable, and resilient a software system will be. ISO 5055 provides before-the-fact measures of the product's software during development to identify and eliminate structural weaknesses before they cause operational problems.

[2] https://www.bcs.org/media/11134/itlf-service-resilience.pdf

[3] https://www.iso.org/standard/65666.html

[4] https://iso25000.com/index.php/en/

- ISO/IEC 25023 – Metrics for Software and System Quality. This standard defines metrics for the external behaviour of a (software) system. These include downtime, number of incidents, and speed of recovery[5].

Software, Systems and Enterprise - Architecture standards

- ISO/IEC/IEEE 42020:2019 - Software, systems and enterprise — Architecture processes - establishes a set of process descriptions for the governance and management of a collection of architectures and the architecting of entities.
- ISO/IEC/IEEE 42030:2019 Software, systems and enterprise — Architecture evaluation framework. The entity being evaluated can be of several kinds: enterprise, organization, solution, system, subsystem, business, data (as a data element or data structure), application, information technology (as a collection), mission, product, service, software item, hardware item, etc.
- ISO/IEC 9837:2026[6] Systems and software engineering — Systems resilience concepts. This document establishes concepts for understanding and improving systems resilience. It is applicable to human-created systems that can be either physical or conceptual, or a combination of both. It applies to systems as defined in ISO/IEC/IEEE 15288, including services and products. It is not intended to apply to naturally occurring systems.

Process certification

- ISO 22301:2019 'Security and resilience — Business continuity management systems - requirements' sets a framework for reviewing and certifying the preparedness of an organisation to recover from disruptions[7].

Standards also cover processes to monitor the likelihood of occurrence and the scale (impact) of consequence.

- ISO 9001:2015 includes risk analysis as an important step in identifying potential problems and deciding how to deal with them[8].
- ISO 31010:2019 provides guidance on selecting and applying techniques for assessing risk in various situations[9].

---

[5] ISO/IEC 25023 - European Standards (en-standard.eu)
[6] https://www.iso.org/standard/83604.html
[7] https://www.iso.org/standard/75106.html
[8] https://advisera.com/9001academy/blog/2015/09/01/methodology-for-iso-9001-risk-analysis/
[9] https://www.iso.org/standard/72140.html

- ISO 31000:2018 provides principles and guidelines for managing risks that could negatively impact organisations[10]. The recommendations in ISO 31000 can be customized to any organisation and its context.

Cybersecurity

- ISO/IEC 27031:2025[11] Cybersecurity — Information and communication technology readiness for business continuity. This document provides guidance on ensuring that information and communication technology (ICT) is prepared to support business continuity. It outlines a framework for ICT readiness that aligns with broader business continuity objectives, helping organizations to prevent, respond to and recover from ICT-related disruptions that could impact critical operations.

Quality standards and guidelines – operational performance

Several standards cover operational performance. They include:

- ISO 22316:2017 Security and resilience – Organisational resilience – Principles and attributes. This standard helps organisations future-proof their business. It details key principles, attributes and activities agreed by experts from all around the world[12].
- BS 65000:2022 - Organisational resilience. Code of practice. This standard identifies the defining attributes of organisational resilience. It offers practical measures to improve it[13].
- ISO/IEC 27001/2 – Information Security Management· This standard provides a framework to address risks to information such as loss of Confidentiality, Integrity, or Availability.
- ISO/FDIS 22316 – Security and resilience[14] provides guidance to enhance organizational resilience for any size or type of public or private organization

[10] https://www.iso.org/iso-31000-risk-management.html/

[11] https://www.iso.org/standard/27031

[12] ISO - Organizational resilience made simple with new ISO standard at https://www.iso.org/news/Ref2189.htm

[13] https://knowledge.bsigroup.com/products/organizational-resilience-code-of-practice?version=tracked

[14] https://resiliencefirst.org/news/draft-of-iso-dis-22316-security-and-resilience-organizational-resilience-released-for-review/

Engineering Resilience for IT systems: Annex - Standards

Infrastructure

Modern infrastructure has become more complex, interconnected and interdependent, which increases the risk of cascading failures from disruptive incidents such as natural hazards, technical failures, cyber incidents or human induced threats. Aging assets, climate change, demographic shifts and technological evolution further amplify these risks.

- ISO 22372:2025 Security and resilience — Community resilience — Guidelines for infrastructure resilience[15] provides guidelines for establishing, maintaining, monitoring and continually improving infrastructure resilience to ensure the continuity and robustness of essential services. Infrastructure is defined broadly as interdependent systems of people, resources, facilities, equipment and services that enable society and organizations to function, including energy, transport, water, telecommunications, finance, health, emergency services and IT infrastructure.

[15] https://www.iso.org/obp/ui/es/#iso:std:iso:22372:ed-1:v1:en

Other regulatory frameworks

CRA

One regulation having increasing effect is the EU's Cyber Resilience Act (CRA). As IT systems integrate an increasing array of other subsystems, some are covered by the CRA Act. This covers all the electronic devices that communicate data. This includes Internet of Things, ATMs, and other data communication devices. The CRA requires reporting and disclosure of known vulnerabilities of such systems. These devices will incorporate software, and it is often not possible to establish the provenance of the embedded software in order to satisfy the regulator.

Regulation of AI

AI adds complexity, but does not change principles. Under UK law, AI itself cannot be held liable. Instead, accountability sits with the individuals and organisations under existing legal frameworks, including negligence, contract, data protection and equality law. So, for instance if an organisation's IT system violates GDPR, that organisation is liable whatever the cause of the breach. The Equality Act 2010 applies where an AI system produces discriminatory outcomes, including in recruitment, lending, or service provision.

The direction of UK law is clear. In its draft Legal Statement on Liability for AI Harms, published in January 2026, the UK Jurisdiction Taskforce (UKJT) reaffirmed that AI has no legal personality under English law and cannot be held liable for harm. Instead, responsibility rests with the individuals and organisations that design, deploy and use AI. In practice, courts are likely to treat AI as a tool, with liability determined under existing principles of negligence, duty of care and other applicable laws.

The EU has also passed new directives and amended existing ones to address AI liability, while Canada has drafted a new law to address this issue.

Conflicting regimes

There are different regulatory regimes covering automotive, shipping, and many other industries. As IT becomes more pervasive, conflicts of regulatory regimes will occur.

Engineering Resilience for IT systems: Annex - Standards

Checklist: Best-practice alignment and improvement

Use the checklist below as a practical gap assessment against recognised good practice. It applies to both new and inherited systems and links board-level assurance questions to concrete engineering and operational evidence.

- Governance and accountability (NIST CSF 2.0 'Govern'; NIST SP 800-53; BCS resilience governance): Assign ownership for each IBS and its resilience targets (RTO/RPO/SLO) and funding decisions; document decision rights and escalation paths; ensure resilience reporting translates operational telemetry into business impact.
- Measurable objectives and tolerances (BCS resilience-by-design; NIST CSF 'Recover'; NIST SP 800-53 CP): Define RTO/RPO and impact tolerances per IBS; align these with SLOs/SLIs and error budgets; review objectives after major changes, incidents, and assurance cycles.
- Resilience engineering across the lifecycle (BCS resilience-by-design; NIST SP 800-160 cyber-resiliency engineering): Treat resilience as a non-functional requirement with explicit design principles (e.g., redundancy, segmentation, graceful degradation, secure recovery); maintain architectural decisions and a deficiency log; use threat-informed scenario planning.
- Operational readiness and learning (NIST SP 800-53 IR/CP; NIST CSF 'Respond/Recover'): Maintain runbooks and recovery playbooks; rehearse using exercises and game days; validate restoration order and data integrity; capture lessons learned and feed them into architecture and change governance.
- Configuration, change, and drift control (ISA/IEC 62443 programme requirements; NIST SP 800-53 CM; ITIL): Maintain authoritative configuration baselines (CMDB and/or automated discovery); ensure changes are risk-assessed, tested, and reversible; monitor for architecture drift between the documented recovery design and deployed reality.
- Segmentation and blast-radius reduction (NIST Zero Trust principles; ISA/IEC 62443 zones & conduits): Partition services into clear failure domains; implement network and identity segmentation to contain both cyber compromise and cascading dependency failure; document trust boundaries and fail-safe modes.
- Third-party and cloud concentration risk (NIST CSF supply chain outcomes; ISA/IEC 62443 supplier practices where applicable): Map critical supplier dependencies and shared infrastructure; define contractual resilience requirements (RTO/RPO, testing evidence, incident communications); design for provider failure where impact tolerance demands it.
- Evidence-based assurance (BCS emphasis on proving effectiveness; NIST SP 800-53 CA/CP/IR evidence): For each IBS, maintain evidence packs:

dependency maps, test results, backup-and-restore reports, failover validation, monitoring coverage, and key risk indicators; use these packs to answer assurance questions with measurable evidence.

- ISMS governance (ISO/IEC 27001): Define ISMS scope, risk assessment method, the Statement of Applicability (SoA), audit cadence, and management review so resilience evidence is maintained as an auditable system.

These checklist items align to the outcomes in the NIST Cybersecurity Framework (CSF) 2.0 across Govern, Identify, Protect, Detect, Respond, and Recover, and can be evidenced using control families in NIST SP 800-53 (notably CP, IR, CM, and CA). For engineering resilience-by-design, NIST SP 800-160 Volume 2 provides cyber-resiliency goals—anticipate, withstand, recover, and adapt—and associated design principles that can be applied to both new development and legacy modernisation.

NIST alignment for resilience governance (CSF 2.0 and SP 800-series)

The table below provides a practical crosswalk that can be used to evidence governance and assurance using either ISO/IEC 27001 ISMS artefacts or NIST outcomes and control families (or both).

The NIST families are:
AC - Access Control
AU - Audit and Accountability
AT - Awareness and Training
CM - Configuration Management
CP - Contingency Planning
IA - Identification and Authentication
IR - Incident Response
MA - Maintenance
MP - Media Protection
PS - Personnel Security
PE - Physical and Environmental Protection
PL - Planning
PM - Program Management
RA - Risk Assessment
CA - Security Assessment and Authorization
SC - System and Communications Protection
SI - System and Information Integrity
SA - System and Services Acquisition

Engineering Resilience for IT systems: Annex - Standards

Governance objective	ISO/IEC 27001 (clauses & typical evidence)	NIST CSF 2.0	NIST SP 800-53 (families)
Scope, context, and stakeholder requirements defined	Clauses 4–5: ISMS scope; interested parties; policy; roles & responsibilities	Govern (GV)	PL, PM
Risk-based prioritisation and objectives for IBSs (RTO/RPO/SLO)	Clause 6: risk assessment & treatment; objectives; Statement of Applicability (SoA)	Govern (GV); Identify (ID)	RA, CP, PL
Operational control and documented procedures (runbooks, change, recovery)	Clauses 7–8: documented information; competence; operational planning & control	Protect (PR); Respond (RS); Recover (RC)	CM, IR, CP, AT
Monitoring, detection, and incident response coordination	Clauses 8–9: operational controls; monitoring evidence; post-incident review inputs	Detect (DE); Respond (RS)	AU, SI, IR
Testing, exercises, and learning (game days / fire drills)	Clauses 8–10: exercise records; corrective actions; continual improvement	Respond (RS); Recover (RC)	IR, CP, CA
Assurance: audit, management review, and evidence packs	Clause 9: internal audit; management review; evidence of effectiveness	Govern (GV)	CA, PM
Corrective action and continuous improvement tracked to closure	Clause 10: nonconformity & corrective action; continual improvement log	Govern (GV); Recover (RC)	CA, PM