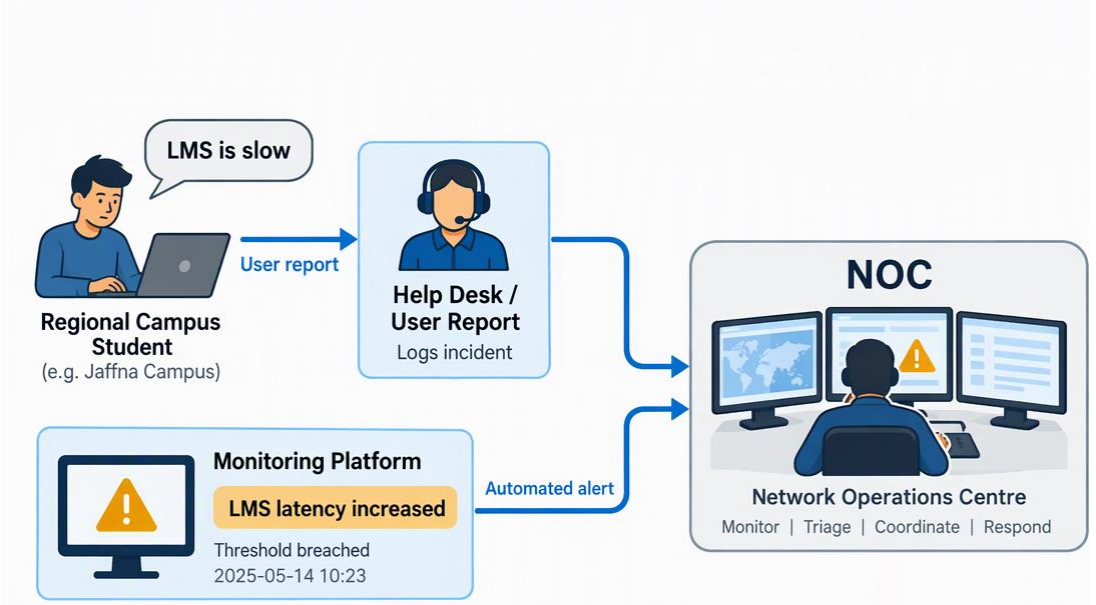
Network Operations Center (NOC)

ITN 4252 Advanced Topics in Networking

Based on resources from Akamai, Google, IBM, ServiceNow, and Splunk.

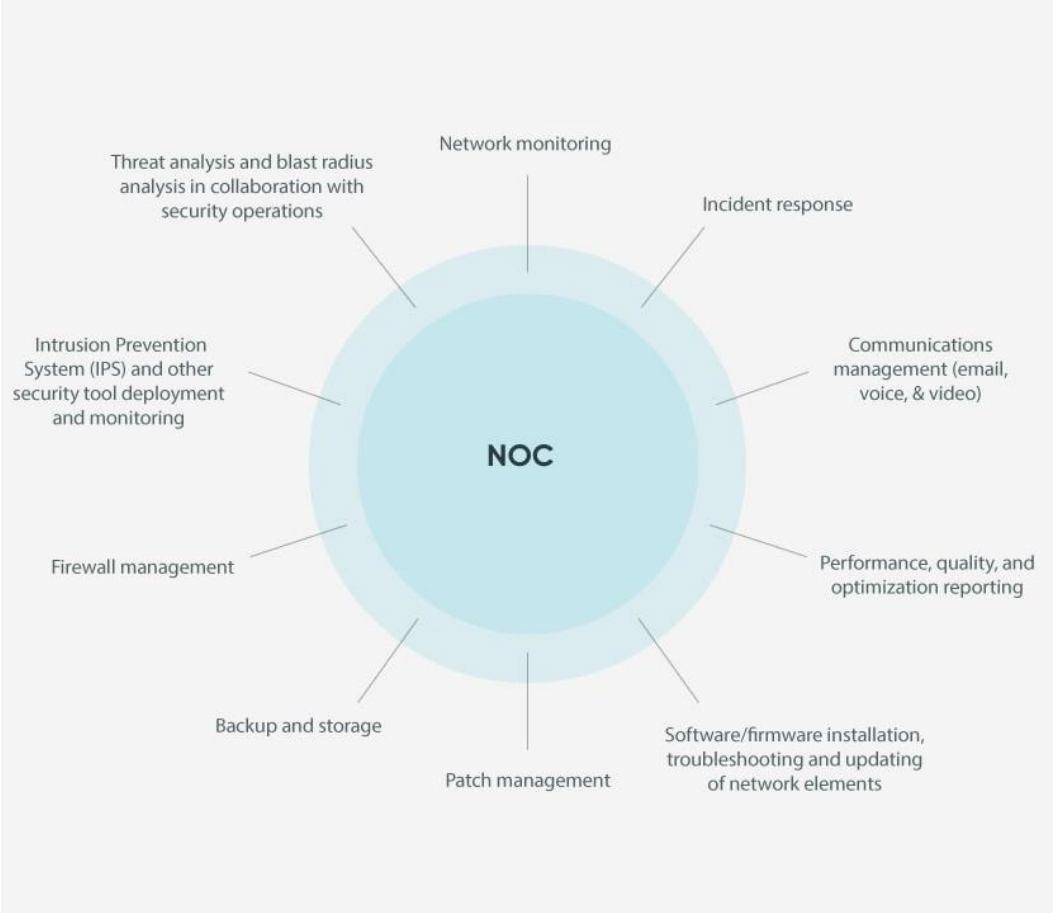
The LMS incident reaches the NOC

- The LMS is degraded, but the cause is unknown.
- Initial signals reaching the NOC:
  - **Monitoring alert**: LMS response time has increased
  - **User report**: A regional campus reports slow pages and timeouts
  - **Service impact**: Students may be unable to use the LMS reliably
  - **Unknown scope**: One user, one campus or a wider issue?
- **Before changing anything: Is this actionable, and who is affected?**



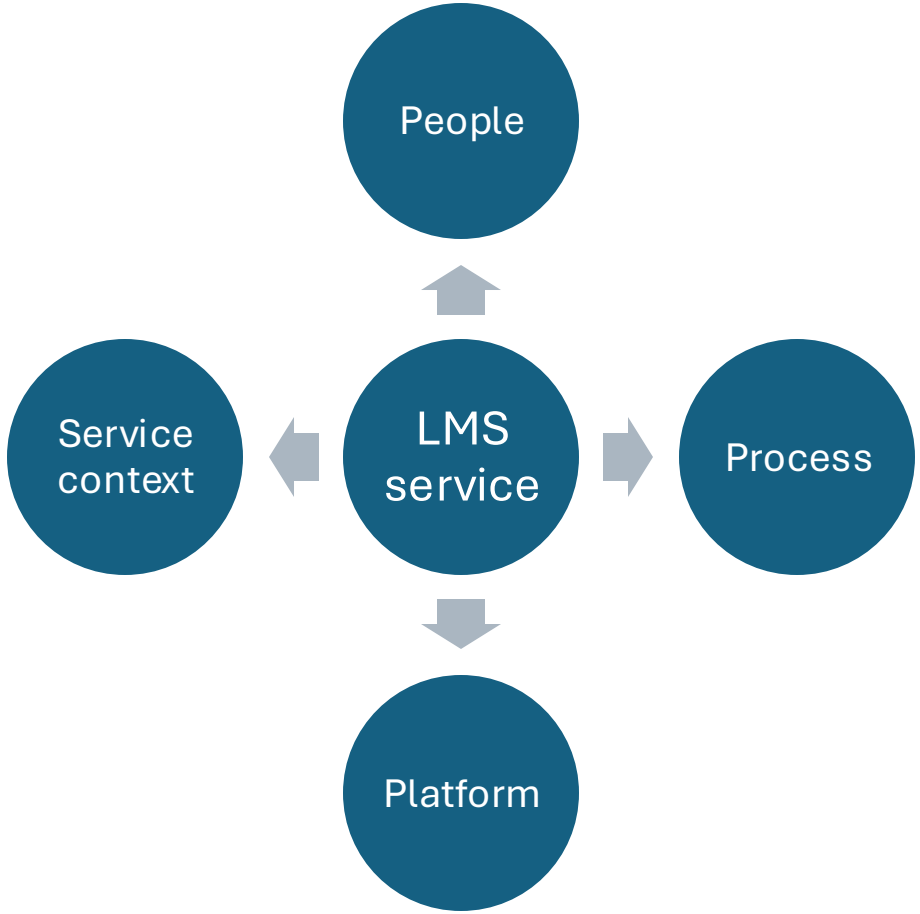
What is a NOC function?

- A NOC is an operational function, not just a room.
- A Network Operations Centre coordinates:
  - **Visibility**: Monitor network and service health
  - **Response**: Triage and manage operational incidents
  - **Coordination**: Connect people, teams and suppliers
  - **Control**: Track actions, changes and service recovery
  - **Continuity**: Maintain operational knowledge and ownership
- **The NOC exists to maintain service. Its physical location is secondary.**



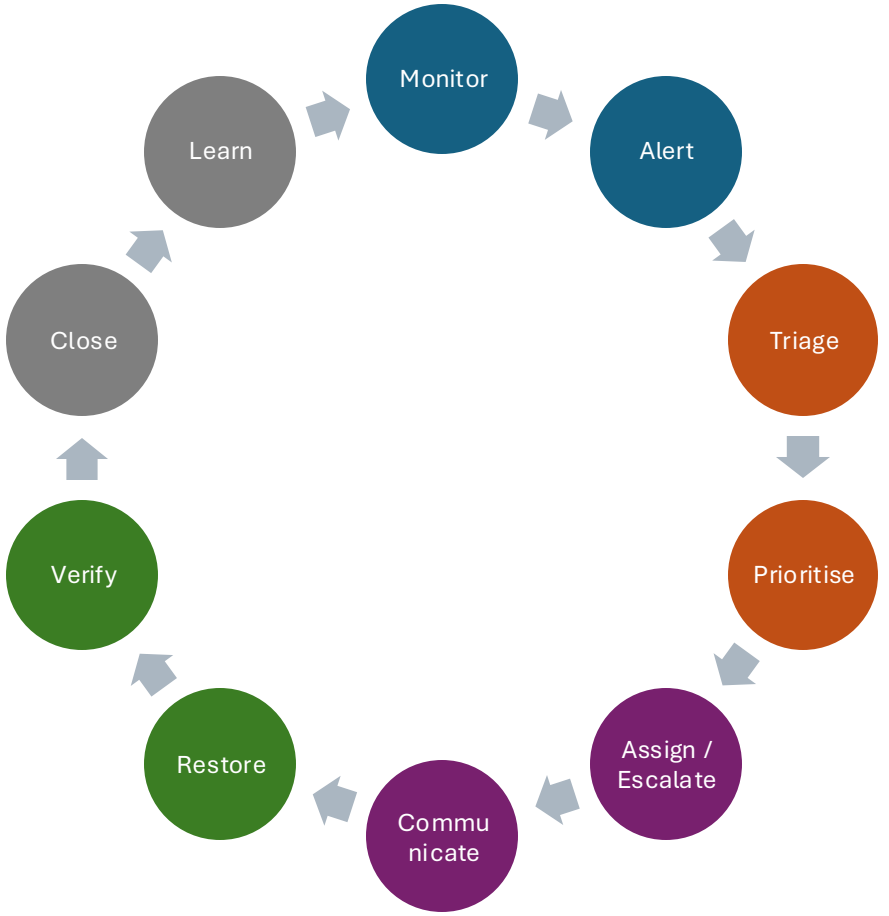
People, process, platform and service context

- Effective NOC operations require four elements
  - **People**: Skills, judgement, ownership and communication
  - **Process**: Consistent triage, escalation and verification
  - **Platform**: Monitoring, evidence, workflow and collaboration
  - **Service context**: Users, dependencies, impact and priorities
- **Technology shows what is happening; operational context guides what happens next.**



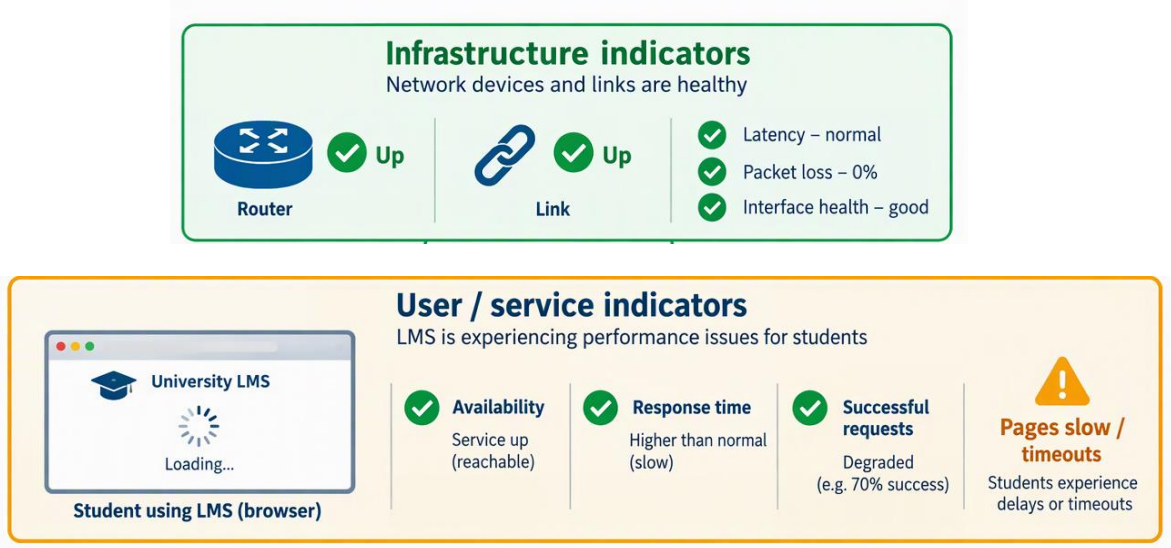
The NOC incident lifecycle

- Move every incident from signal to learning.
  - **Monitor → Alert**: Detect potential service impact
  - **Triage → Prioritise**: Confirm scope, impact and urgency
  - **Assign / Escalate → Communicate**: Maintain ownership and coordinate expertise
  - **Restore → Verify**: Recover the LMS and confirm user experience
  - **Close → Learn**: Record the outcome and improve future response
- **The incident is not finished when action is taken. It ends after recovery is verified and learning is captured.**



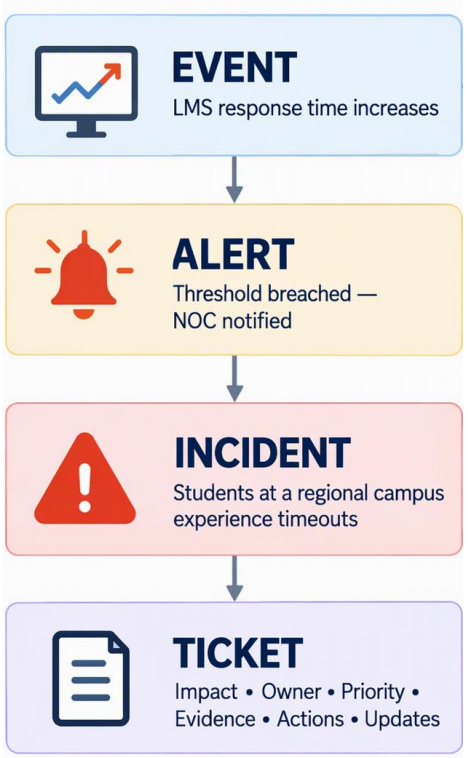
Monitor services, not only devices

- A healthy device does not guarantee a healthy service.
- Service indicators
  - **Availability**: Can students use the LMS?
  - **Response time**: Are pages loading acceptably?
  - **Successful requests**: Are transactions completing?
- Supporting network indicators
  - Latency and packet loss
  - Interface and path health
- **Device health is evidence; service health determines user impact.**



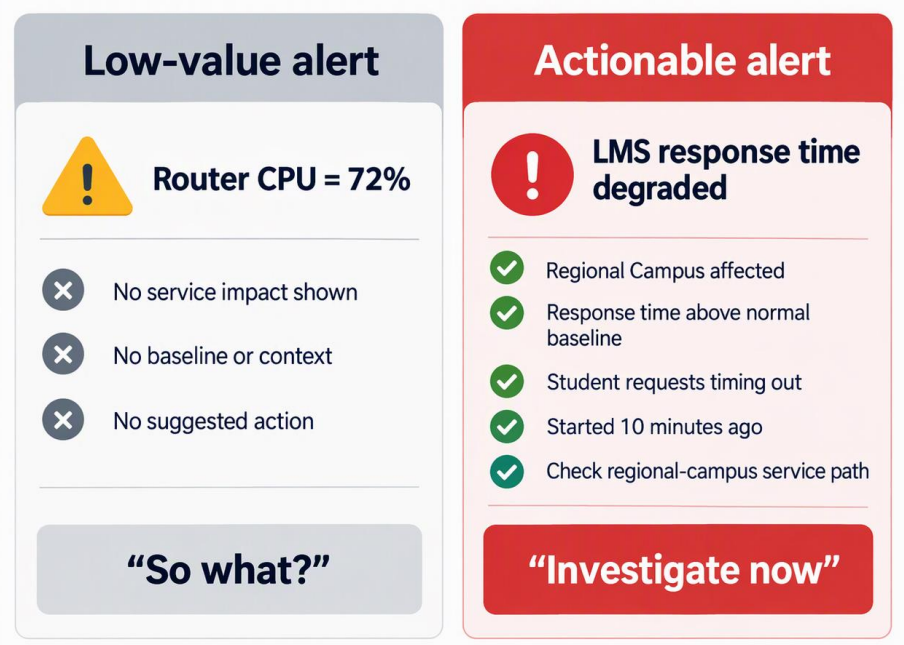
Event, alert, incident and ticket

- Not every event becomes an incident
  - **Event**: Something observable happened
  - **Alert**: A condition requires operator attention
  - **Incident**: Service is degraded or disrupted
  - **Ticket**: The tracked record of the operational response
- **Signals become incidents when they require coordinated action to restore service.**



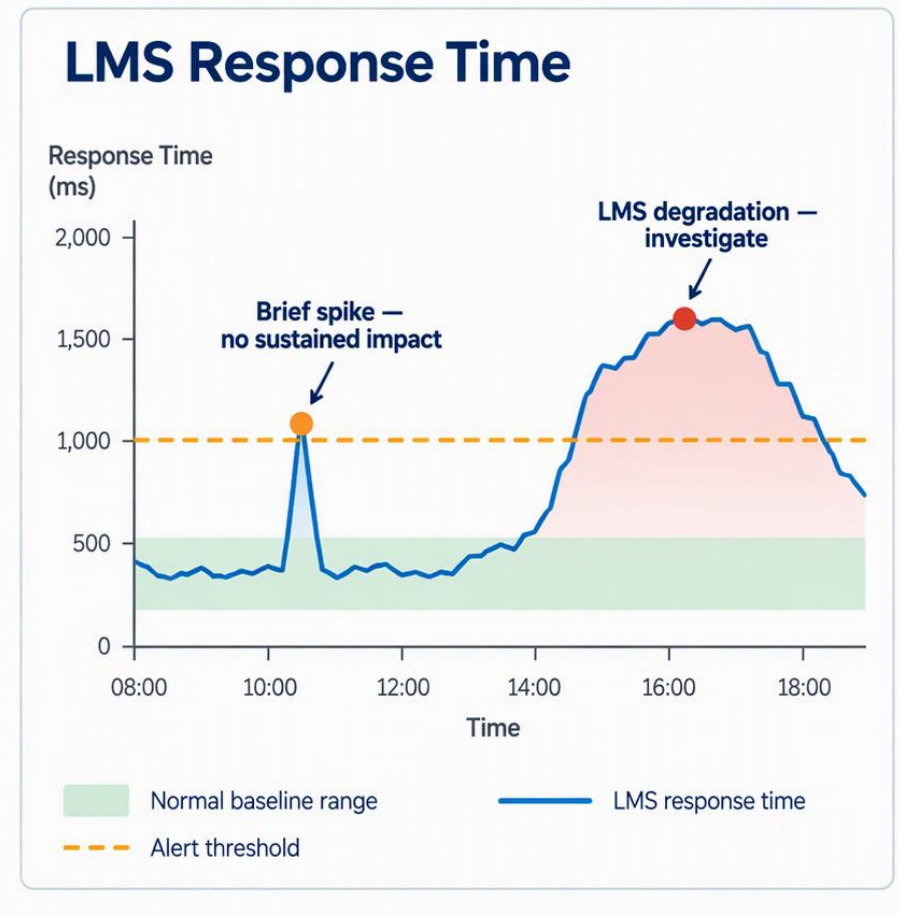
What makes an alert actionable?

- A useful alert should tell the NOC to act.
- An actionable alert should indicate:
  - **Impact**: Which service or users may be affected?
  - **Urgency**: Does someone need to respond now?
  - **Evidence**: What condition triggered the alert?
  - **Context**: Where is the problem occurring?
  - **Next step**: What should the operator investigate?
- **If every signal becomes an alert, important problems disappear into the noise.**



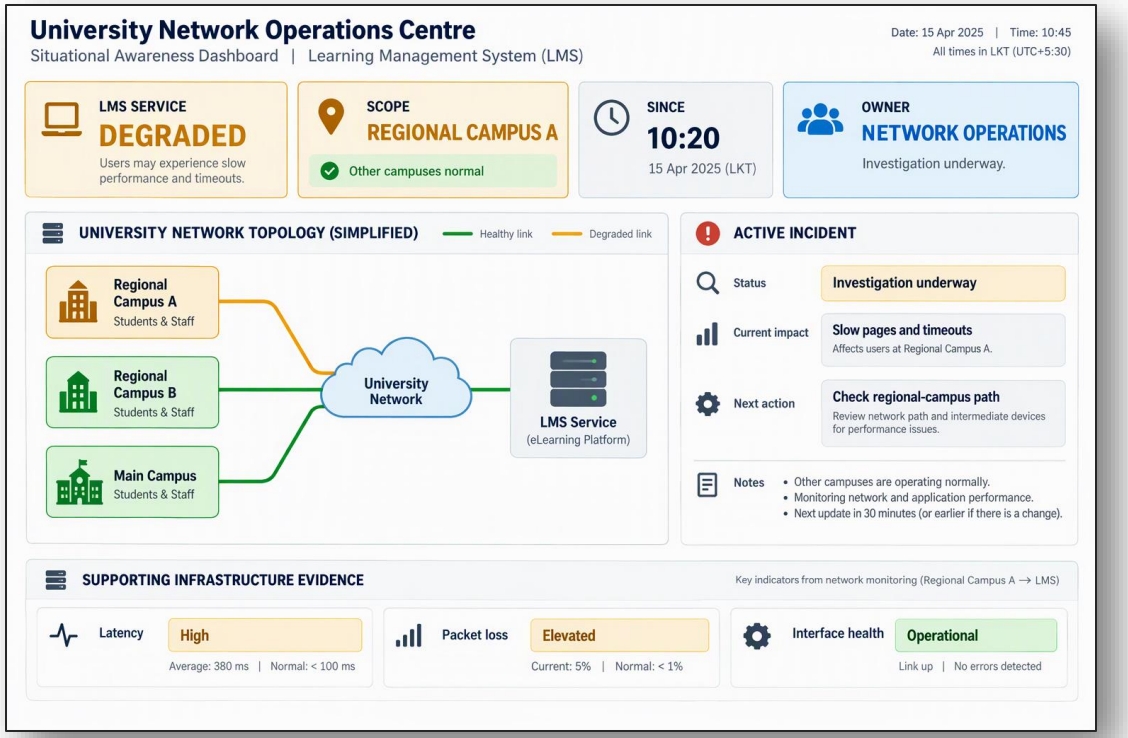
Thresholds, baselines and alert noise

- Alert when behaviour matters — not whenever a number changes
  - **Threshold**: Alert when a metric crosses a defined limit
  - **Baseline**: Compare current behaviour with what is normally expected
  - **Duration**: Avoid reacting to brief, harmless spikes
  - **Grouping**: Combine related alerts from the same incident
  - **Tuning**: Review alerts that are noisy, missed or unactionable
- **Too sensitive creates noise; too insensitive hides real problems.**



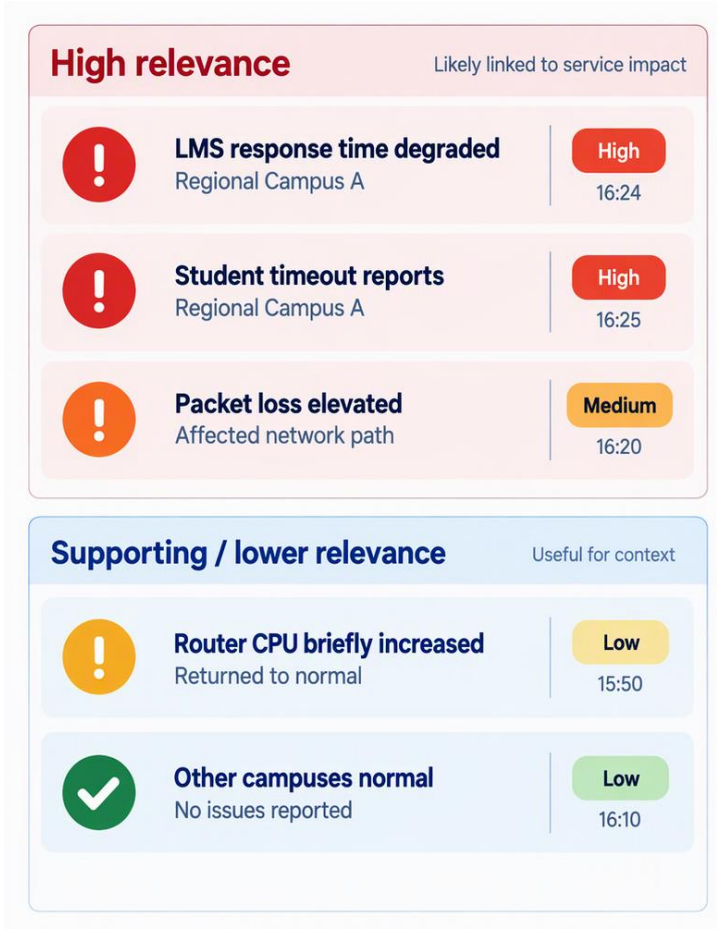
Dashboard

- A dashboard should explain the incident at a glance.
  - **Service health**: Is the LMS available and responsive?
  - **Scope and impact**: Which campuses and users are affected?
  - **Time**: When did the degradation begin?
  - **Supporting evidence**: What is happening across the network path?
  - **Status and ownership**: Who is responding, and what happens next?
- **What is affected, how badly, since when and who owns it?**



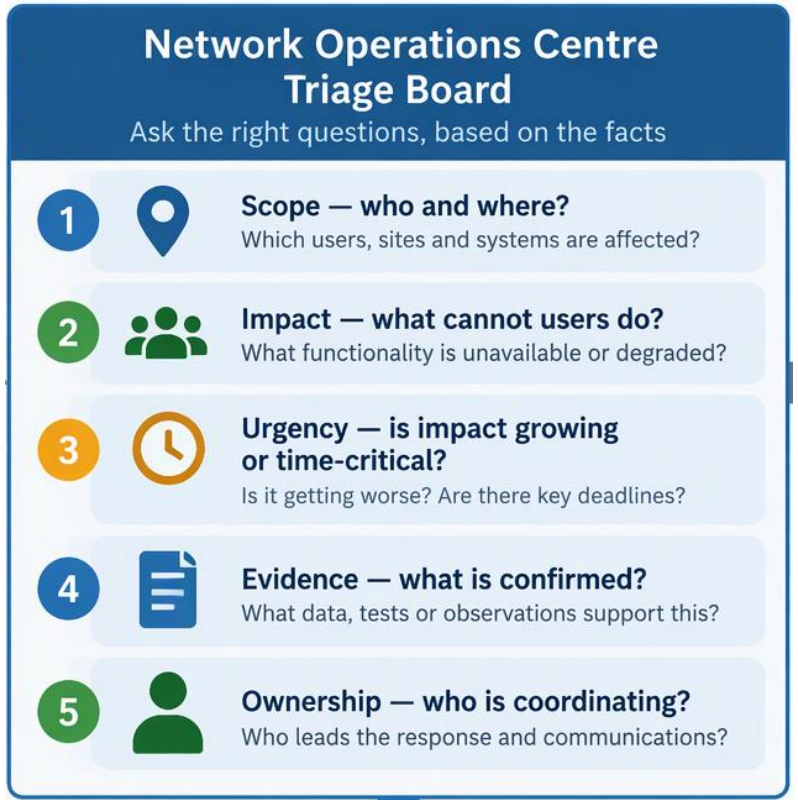
Signals

- Not every alert deserves equal attention.
- The NOC sees several signals at once:
  - **LMS response time**: High at Regional Campus A
  - **Student reports**: Pages slow and timing out
  - **Packet loss**: Elevated on the affected path
  - **Router CPU**: Brief increase, now normal
  - **Other campuses**: LMS operating normally
- **Prioritise signals that help explain service impact.**



Triage

- Triage defines the incident before deeper diagnosis
  - **Scope**: Which users, campuses and services are affected?
  - **Impact**: What can affected users no longer do reliably?
  - **Urgency**: Is the disruption growing or time-critical?
  - **Evidence**: What is confirmed, and what remains unknown?
  - **Ownership**: Who coordinates the next actions?
- **Triage narrows the problem. It does not prove the root cause.**



Severity vs. priority

- Impact describes the incident, priority drives the response
  - **Severity**: How much is the service affected?
  - **Scope**: How many users, sites or services are affected?
  - **Urgency**: How quickly could the impact worsen?
  - **Priority**: How quickly should the NOC respond?
- **A technically serious fault is not always the highest operational priority.**



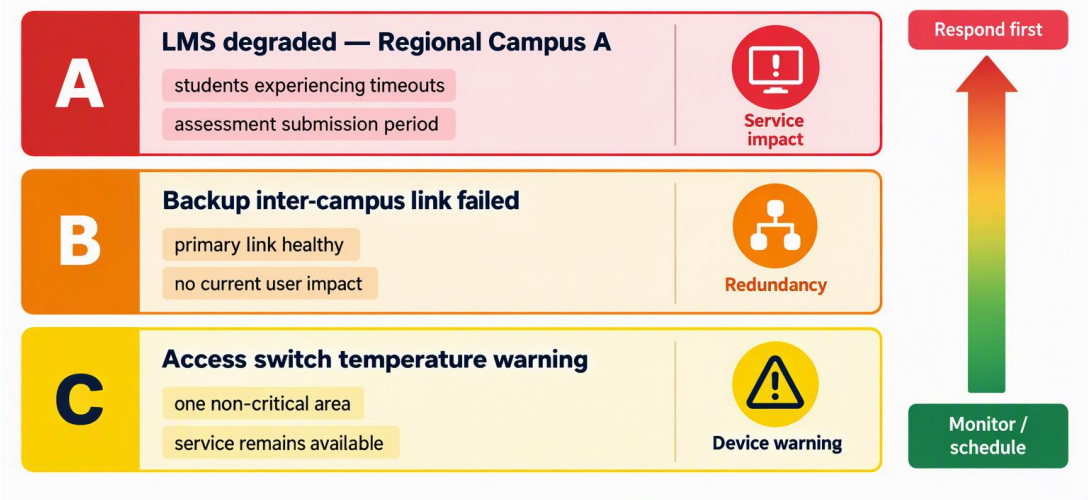
SLIs, SLOs and SLAs

- Service targets give operational priority context
  - **SLI**: What we measure about the service
  - **SLO**: The target level we aim to achieve
  - **SLA**: A service commitment between parties
  - **NOC use**: Monitor performance and respond when service targets are at risk
- **Measure the service → define the target → understand the commitment.**



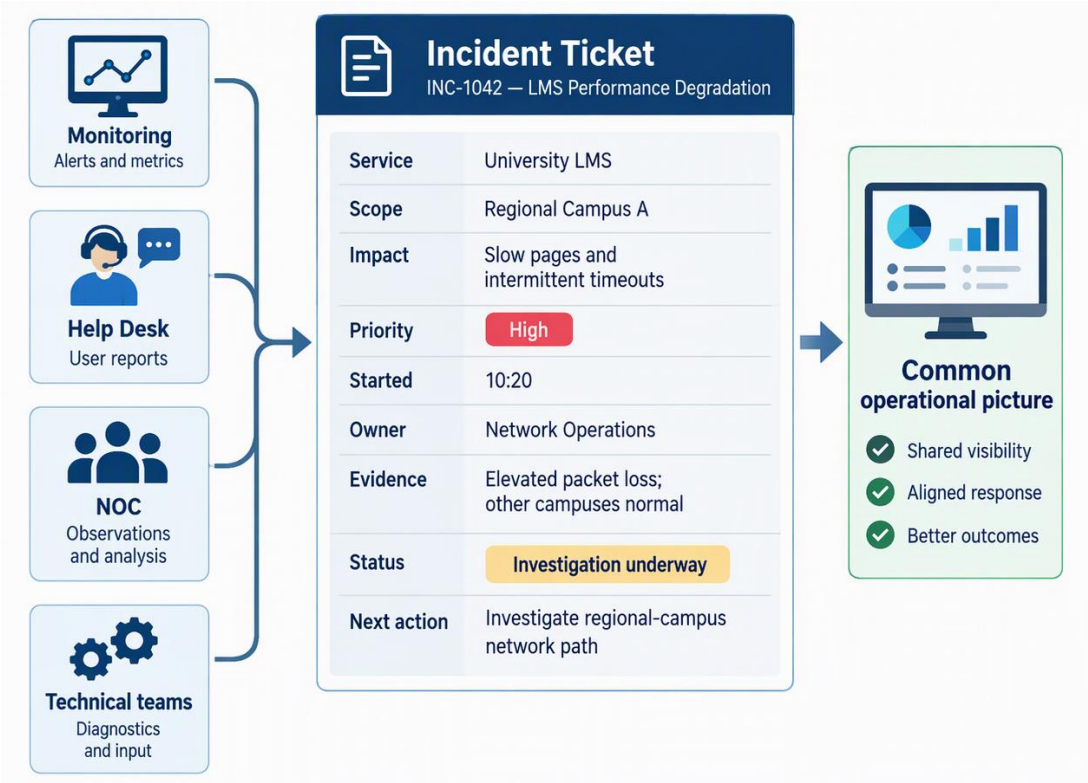
Prioritisation

- Which incident should the NOC handle first?
  - **A. LMS degraded**
    - Regional Campus A
    - Student timeouts
    - Assessment submission period
  - **B. Backup link failed**
    - Primary link healthy
    - No current user impact
    - Redundancy reduced
  - **C. Device warning**
    - One access switch
    - Service available
    - No reported user impact
- **Prioritise using service impact, scope, urgency and available workarounds.**



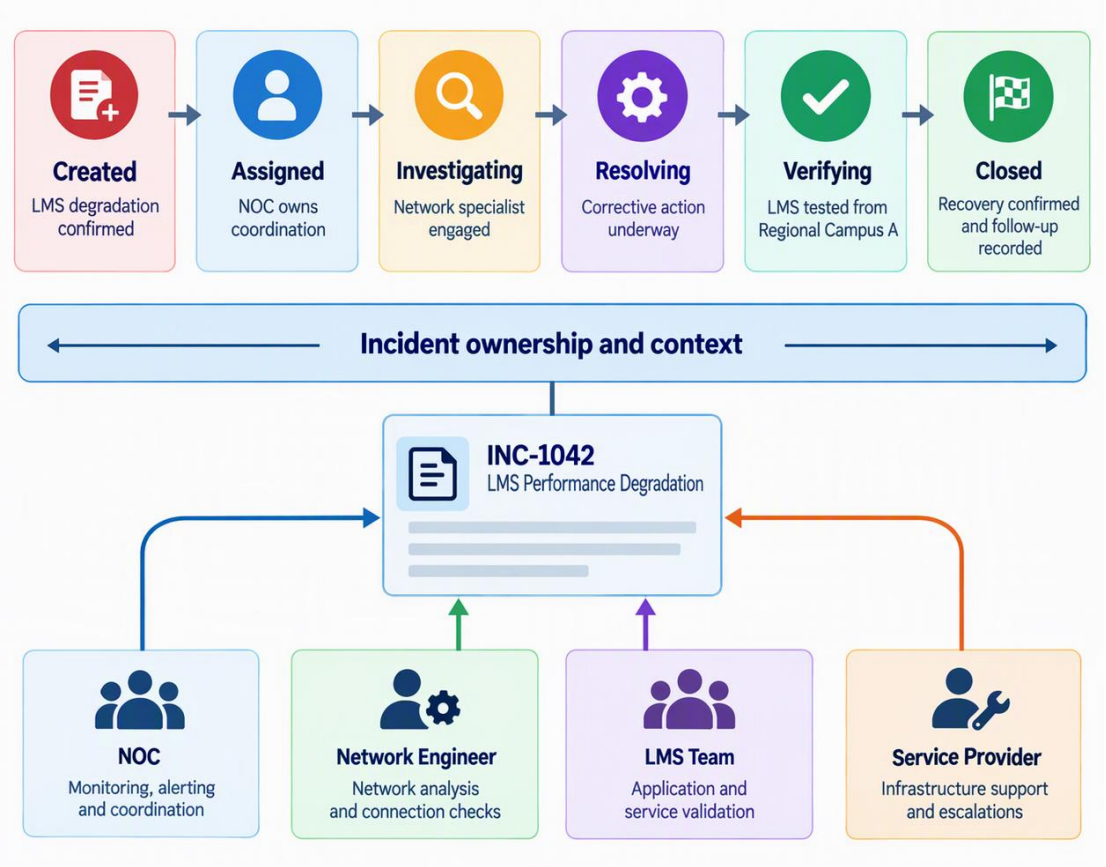
Incident record

- One incident needs one reliable source of truth.
- Record the essentials:
  - **Impact**: What service and users are affected?
  - **Priority**: How urgently should the incident be handled?
  - **Evidence**: What facts and observations are confirmed?
  - **Ownership**: Who is coordinating the response?
  - **Actions**: What has been done, and what happens next?
- **The ticket should tell the incident story as it unfolds.**



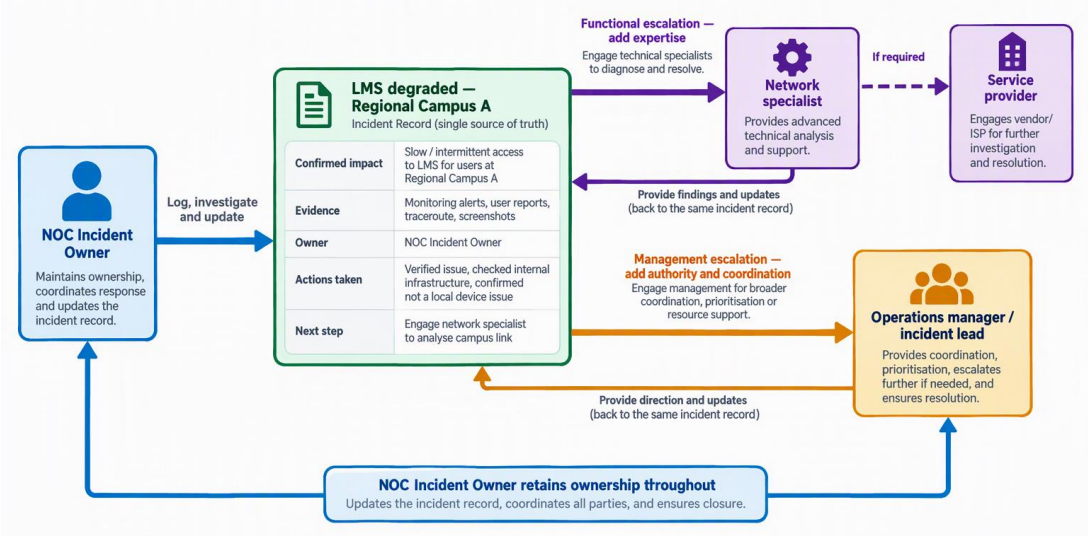
Ownership

- The incident needs an owner from start to finish
  - **Assign**: Make responsibility clear
  - **Investigate**: Record evidence, decisions and actions
  - **Update**: Keep status and impact current
  - **Resolve**: Apply the agreed restoration action
  - **Verify & close**: Confirm service recovery before completion
- **Work may move between teams, but ownership must not disappear.**



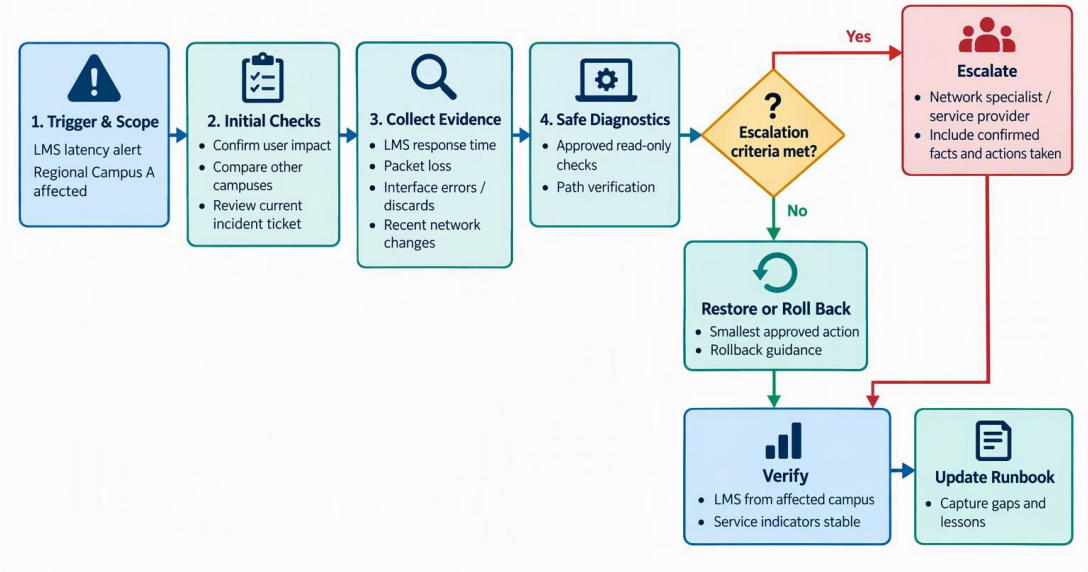
Escalation

- Escalation adds support. It should not remove ownership.
  - **Functional escalation**: Add specialist technical expertise
  - **Management escalation**: Add authority, coordination or resources
  - **Supplier escalation**: Engage an external service provider when required
  - **Preserve context**: Carry forward impact, evidence, actions and next steps
  - **Retain ownership**: Someone continues coordinating the incident
- **Escalate the capability, not the confusion.**



Runbooks

- A runbook guides repeatable incident response
  - **Trigger and scope**: When should the runbook be used?
  - **Checks and evidence**: What should the operator examine?
  - **Decision points**: When should the incident be escalated?
  - **Recovery guidance**: How should service be restored or rolled back?
  - **Verification**: How do we confirm that the LMS has recovered?
- **Follow the runbook but apply judgement when the incident differs.**



Handover

- A handover transfers responsibility without losing momentum
  - **Impact**: Current service condition and affected users
  - **Evidence**: Confirmed facts and remaining uncertainty
  - **Actions**: Completed, active and pending changes
  - **Risks**: What could worsen or require attention
  - **Ownership**: Next action, owner and update point
- **The incoming responder should continue the incident, not reconstruct it.**

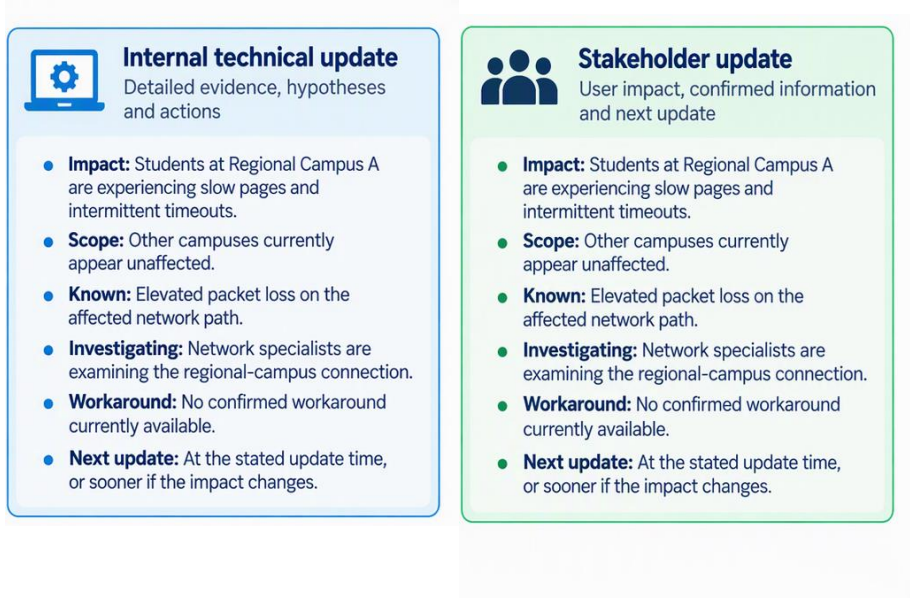


LMS Incident — Regional Campus A
Shared Incident Record

Current service impact	LMS unavailable for students and staff at Regional Campus A. Users unable to log in or access courses.
Confirmed facts	• Outage started at 10:24 AM (LKT) • Affects Regional Campus A only • Core LMS servers not responding • Issue confirmed by monitoring and user reports
Remaining uncertainty	• Exact root cause not yet identified • Potential wider impact unknown (other campuses) • ETA for full restoration unclear
Incident owner	NOC (Applications Team)
Priority	High (P1)
Actions completed	• Issue detected and escalated • Initial troubleshooting completed • Application and infrastructure teams notified • Stakeholders (IT and campus) informed
Active or pending changes	No changes made. No planned changes during incident.
Risks	• Ongoing disruption to teaching and learning • Increased support centre volume • Potential reputational impact
Next action	Application team to continue investigation and identify root cause.
Next update time	1:00 PM (LKT)

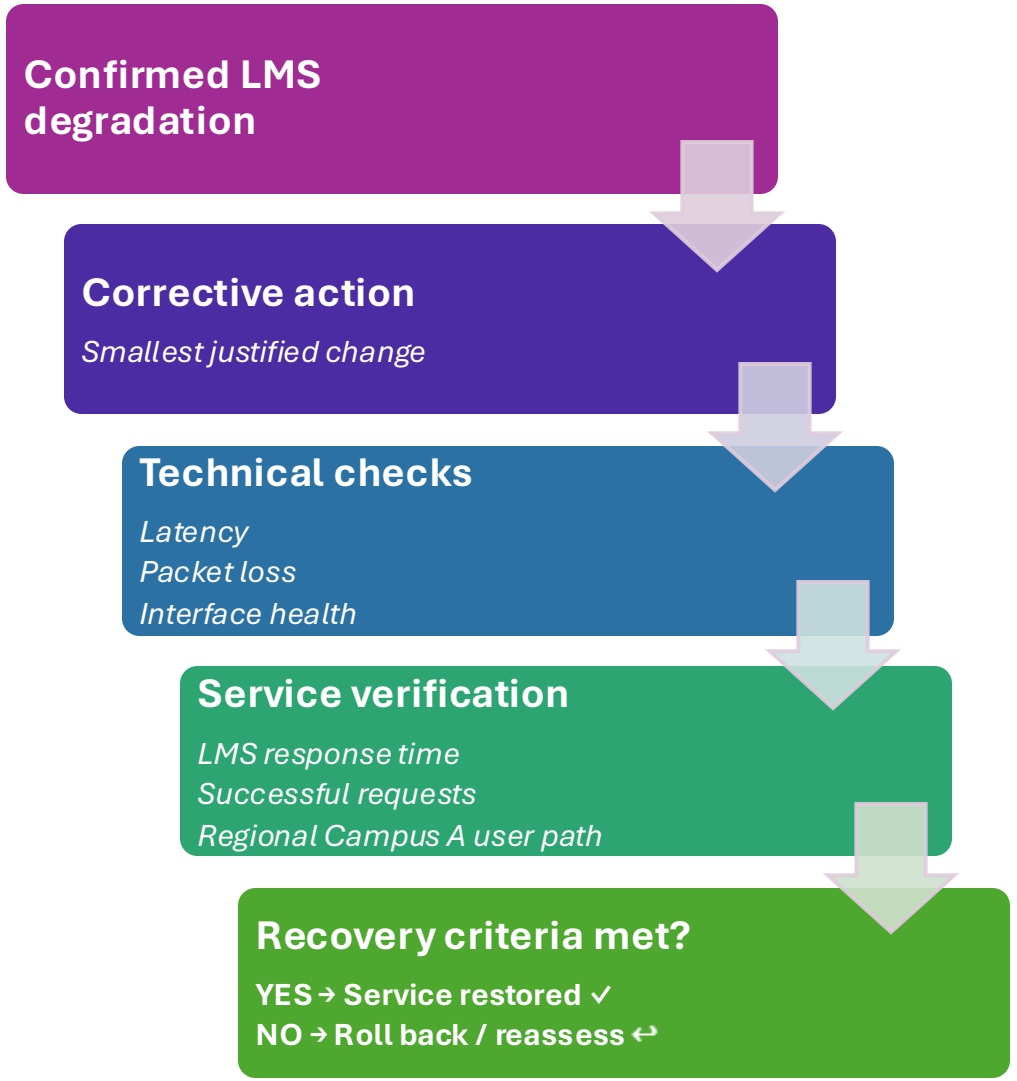
Communicate what is known and unknown

- Incident updates should be clear, accurate and timely.
  - **Impact**: What are users experiencing?
  - **Scope**: Which services and locations are affected?
  - **Confirmed facts**: What do we know so far?
  - **Action**: What is being investigated or restored?
  - **Next update**: When will stakeholders hear from us again?
- **Communicate uncertainty honestly, do not turn a hypothesis into a cause.**



Restore the service and verify

- Restoration is not complete until users recover
  - **Restore**: Apply the smallest justified corrective action
  - **Observe**: Watch for immediate side effects or new alerts
  - **Verify**: Repeat the checks that demonstrated the problem
  - **Confirm**: Test the LMS from the affected campus
  - **Rollback**: Revert if recovery criteria are not met
- **“The change worked” is not the same as “the service recovered.”**



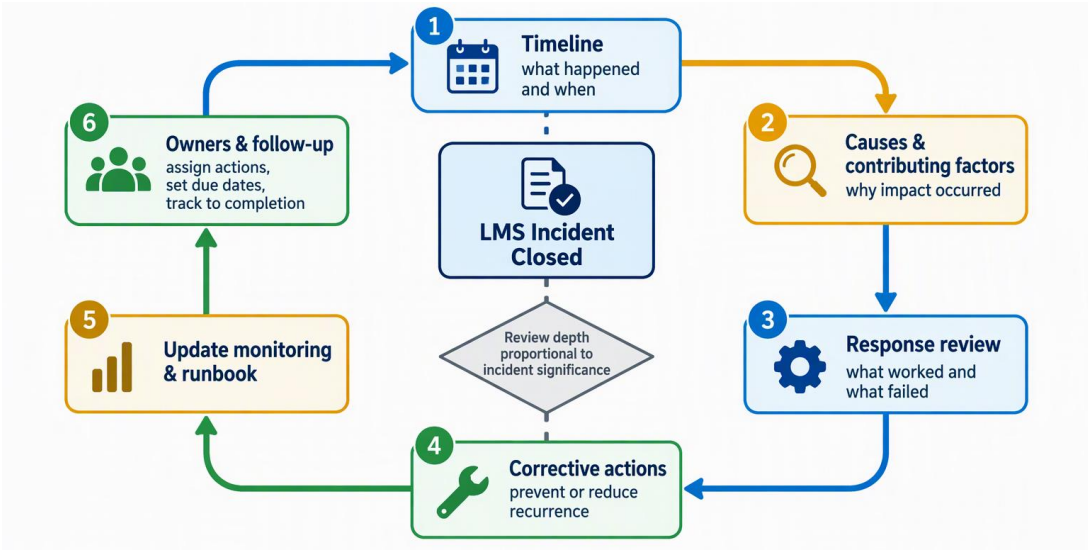
When is an incident ready to close?

- Close only when recovery is confirmed.
  - **Service recovered**: Students can use the LMS reliably
  - **Monitoring stable**: Latency, loss and service indicators remain normal
  - **Stakeholders updated**: Recovery has been communicated
  - **Record complete**: Resolution and verification evidence are documented
  - **Follow-up owned**: Remaining actions have clear owners
- **Closure ends the active incident, not every long-term improvement.**



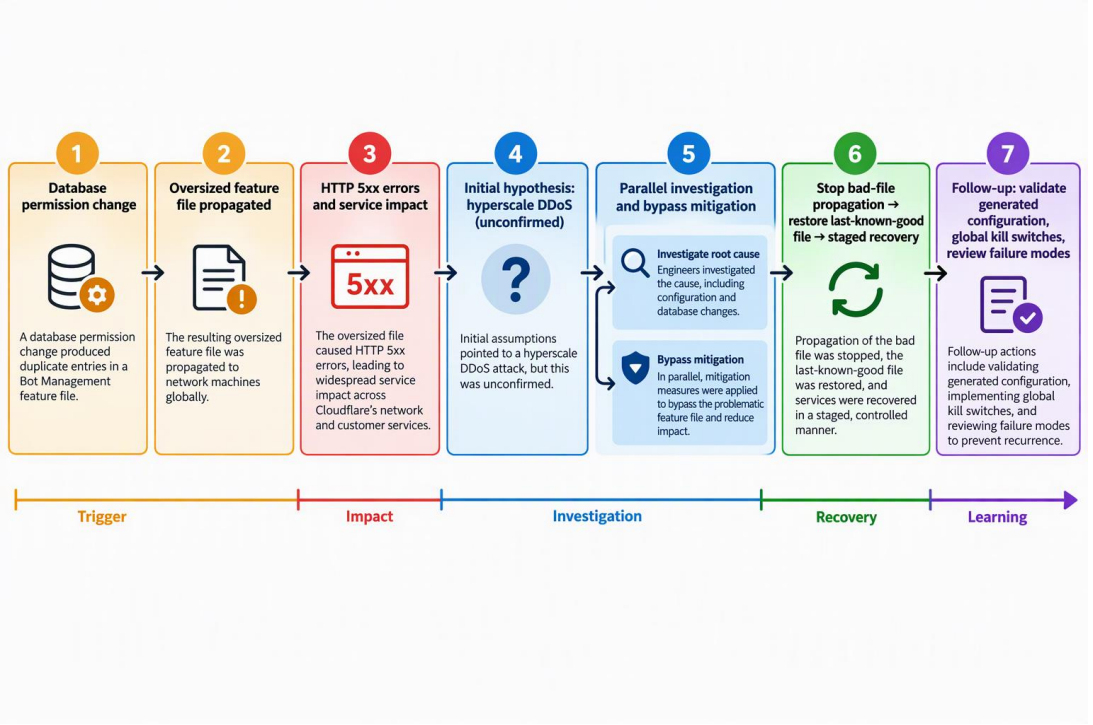
Post-incident learning

- Turn the incident into operational improvement
  - **Reconstruct**: What happened, when and with what impact?
  - **Understand**: What causes and contributing factors were involved?
  - **Evaluate**: What worked well, and what slowed the response?
  - **Improve**: Update monitoring, alerts, runbooks and safeguards
  - **Track**: Assign owners to agreed follow-up actions
- **Learn from the system, not by blaming individuals.**



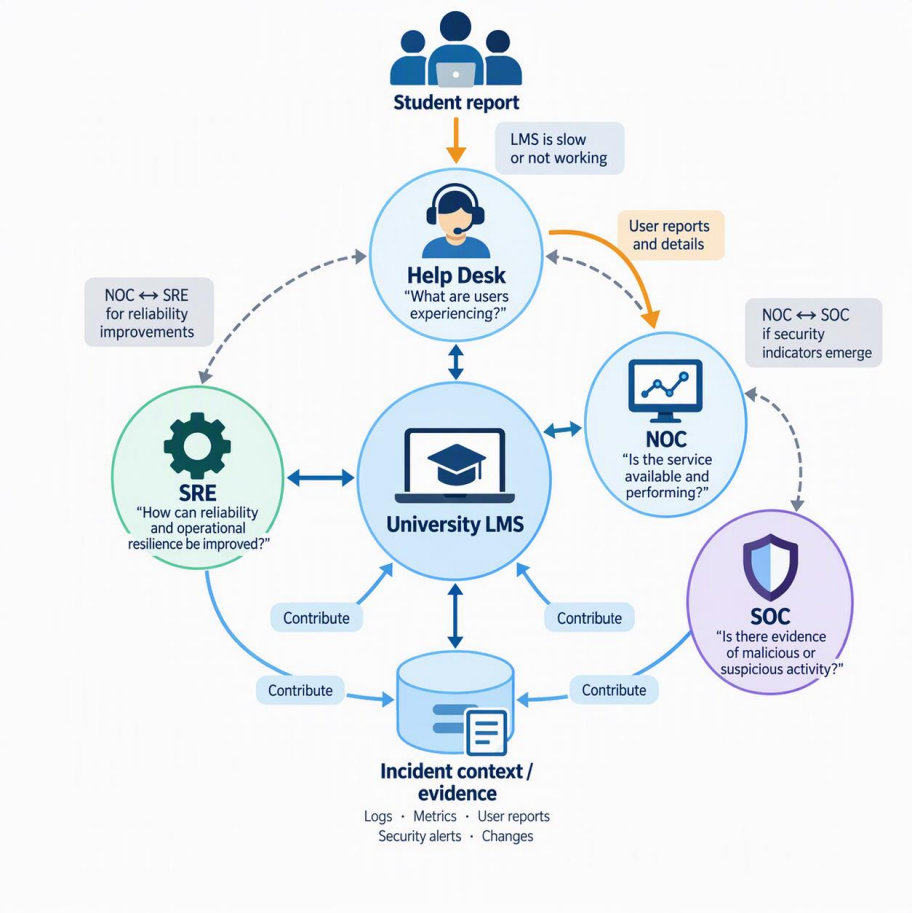
Case study: Cloudflare outage, 18 Nov 2025

- An internal change became a widespread service incident
  - **Trigger**: A permissions change produced an oversized Bot Management feature file
  - **Impact**: Core proxy failures generated HTTP 5xx errors
  - **Initial hypothesis**: A hyperscale DDoS attack was suspected—but not confirmed
  - **Restore**: Stop propagation, bypass affected paths and restore known-good state
  - **Learn**: Strengthen validation, kill switches and failure safeguards
- **Treat hypotheses as hypotheses, restore safely, verify fully and learn systematically.**



NOC, Help desk, SOC and SRE

- Different teams see the same service from different perspectives
  - **Help Desk**: User contact, initial support and service requests
  - **NOC**: Availability, performance and service restoration
  - **SOC**: Security monitoring, investigation and response
  - **SRE**: Reliability engineering, service objectives and automation
  - **Collaboration**: Share evidence, context and ownership when incidents overlap
- **The boundaries vary. The service still needs coordinated ownership.**



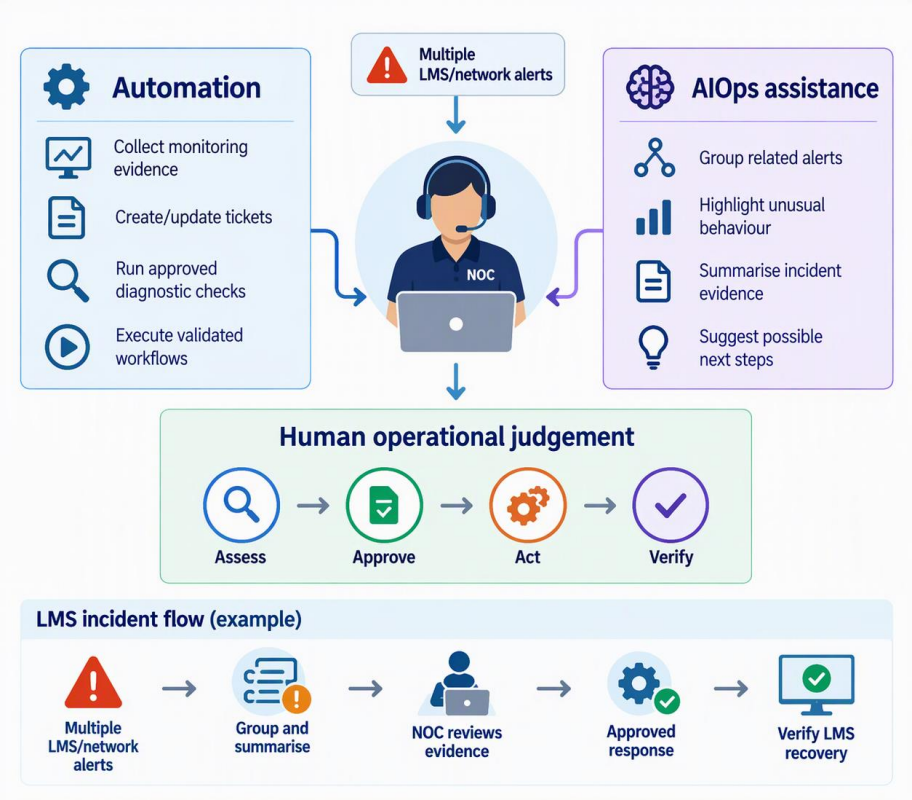
In-house, outsourced or hybrid NOC

- The NOC operating model changes who does the work
  - **In-house**: Strong organisational context and direct control
  - **Outsourced**: External expertise, capacity or extended coverage
  - **Hybrid**: Internal ownership combined with provider capability
  - **Responsibilities**: Define monitoring, escalation and decision boundaries
  - **Integration**: Share evidence, tickets and communication paths
- **The best model is the one with clear ownership, not simply the lowest cost.**



Automation and AIOps

- Automation can reduce workload, but judgement still matters.
  - **Automate**: Repeatable checks, data collection and routine actions
  - **Reduce noise**: Group or deduplicate related alerts
  - **Assist diagnosis**: Highlight anomalies and summarise evidence
  - **Forecast**: Identify emerging capacity or performance patterns
  - **Validate**: Keep people involved when impact or cause is uncertain
- **Automate known processes, do not automate uncertain assumptions.**



Can you manage the LMS incident?

- For the university LMS incident:
  - **Detect & triage**: Confirm service impact and scope
  - **Prioritise & own**: Set priority and maintain one incident record
  - **Escalate & communicate**: Add expertise and keep stakeholders informed
  - **Restore & verify**: Recover the service and prove it is healthy
  - **Close & learn**: Record lessons and assign follow-up actions
- **A NOC turns signals into coordinated, verified and learnable action.**

