MAXIRON
BRING STRENGTH TO YOUR DELIVERY

Maximo health check and audit guide

IBM
Gold Partner

Table of Contents

Document ID	MXR-GUIDE-004
Version	1.0
Date	2026-04-24
Owner	MaxIron Ltd
Audience	Procurement, IT, asset management and operations leaders evaluating IBM Maximo and MAS
Classification	Public — share freely with credit
Canonical URL	https://maxiron.com/guides/maximo-health-check-audit-guide/

A Maximo health check is not a single tool you run. It is a structured audit of seven areas of the platform — data, work-management discipline, integrations, performance, security, operability, and upgrade-readiness — that decides whether the system is on a good path or quietly drifting into the kind of state that costs money to recover. The deliverable is a triaged list of findings with a remediation plan and an honest view of what the platform is actually capable of supporting today.

The health check itself is the easy part. The hard part is what happens next: who fixes the findings, how quickly, with what evidence, and how the platform is kept healthy after the consultants leave. The reason MaxIron approaches health checks differently from most partners is that we run them on top of our own product set — Sentinel, Diagnose, Autoheal, AI Smart Data, Cloud Manager and Change Control — so the audit is not a one-off PDF. It is the start of an operating model where the platform stays healthy continuously.

This guide sets out how we run a health check on a production Maximo or MAS environment, what we use to find issues, and what we use to fix them. It is the same framework we use on engagements, written so an internal team can use it as well.

1. When to commission a health check

The honest trigger list is short:

- Performance has degraded to the point that users complain, but no single root cause is obvious.
- An upgrade to MAS is on the executive's roadmap and the receiving team needs an evidence-based view of readiness.
- A new partner has taken over support and needs to understand what they have inherited.
- A major change is coming — a new sector, an acquisition, a mobile rollout, a regulator audit — and the system needs a defensible baseline first.

If none of those is true, a health check is a nice-to-have and probably can wait. If any of them is true, it is the single most useful thing the IT and operations leads can spend a fortnight on, because every other decision afterwards depends on it.

2. The seven areas, in order

The order matters. Each area depends on the one before it being trustworthy.

2.1 1. Data quality and the asset hierarchy

Maximo is an asset management system. If the asset register is wrong, everything downstream — work, parts, KPIs, Monitor, Health, regulator returns — is wrong in subtle ways that are very hard to spot.

We look at:

- The asset hierarchy: depth, consistency of naming, orphaned assets, duplicates by serial number or location.
- Criticality coding: is it populated, is it consistent, does it actually drive PM and inspection scheduling, or is it a field someone filled in once.
- Failure coding: is the ISO 14224 (or equivalent) hierarchy in use, is it being used, is the long tail of "other" coded failures growing each year.
- Locations and operating context: is the spatial and operational hierarchy something a planner can navigate without phoning someone.
- Inventory master data: duplicate items, items without a class, items with no on-hand record, dead stock.
- Vendor and contract master data: how many active vendors, how many of those have had a transaction in 24 months.

Good looks like: an asset hierarchy a new joiner can navigate without help, criticality that is meaningful at audit, failure coding consistent enough to be the basis of Predict one day. Bad looks like: 30,000 assets, 18,000 of them with the same description, criticality mostly null.

This is also where we point MaxIron AI Smart Data at the estate during the health check. It surfaces duplicates, gaps, broken hierarchies and inconsistent classifications at a depth a manual review will not reach in a fortnight, and — more importantly — it is the same tool we use afterwards to actually fix what was found, at scale, with audit. The data-quality finding and the data-quality remediation are the same product, not a finding handed to a separate clean-up project that never happens.

2.2 2. Work-management discipline

The point of Maximo is the work loop — request, planning, scheduling, execution, completion, history. We audit it as a loop, not as a list of screens.

- What share of work goes through proper planning, with task plans and labour estimates? What share is reactive?
- What is the work order completion rate, by craft and by site, and what is the open-work-order ageing profile?

- What share of completed WOs have meaningful close-out data — actuals, failure codes, downtime hours, parts used? "Meaningful" excludes "see comments" in a memo field.
- Is PM compliance measured, reported, and acted on? What is the PM-to-corrective ratio over twelve months?
- Are work order priorities driven by criticality, or by who shouted loudest?

The output is a one-page picture of how the work-management loop is actually behaving. It is usually not what the senior team thinks it is. Where the loop is breaking on configuration — start centres that no role actually uses, KPI definitions that have drifted from the business question — the fix is a configuration change managed through MaxIron Change Control so the change has approvals, evidence and a rollback path, not a "someone changed it on Friday" story.

2.3 3. Integrations

Most Maximo estates have between four and twenty integrations: ERP, finance, GIS, HR, identity, payroll, document management, time and attendance, SCADA via Monitor, parts catalogues, vendor portals. Each one is a potential single point of failure.

We document:

- Integration inventory: name, direction, transport (REST, MIF, ESB, file, DB link), throughput, owner, last incident.
- Error volume per integration over 30 and 90 days. What gets retried, what fails silently.
- Whether any integration is doing real-time updates that would surprise the operations team if they paused.
- Authentication: where credentials are stored, how they are rotated, when they last were.

Most "Maximo is slow" tickets are integration-shaped. This is where we usually find them. On managed estates we run MaxIron Sentinel over the integration layer so the failure is visible the minute it starts, not three days later when a downstream system queries Maximo for the missing record. Sentinel turns "Maximo lost the integration last Wednesday" into "the integration backed up at 02:14, here is the alert, here is the queue depth".

2.4 4. Performance

Only after the first three is performance worth measuring honestly. Otherwise it gets fixed at one layer and the cause is at another.

The areas we measure:

- Application response time on the busiest screens (work order, asset, inventory) at peak hours, broken down by user role.
- Database health: index coverage, fragmentation, statistics freshness, locking and blocking, top queries by elapsed time and by execution count.

- JVM behaviour on the application servers: heap, garbage collection patterns, thread pool saturation.
- MIF and Integration Framework throughput and queue depths.
- Cron task and escalation execution time and overrun frequency.
- BIRT / report queue length, slow reports, reports nobody opens any more.

The output is a ranked list of the five things actually hurting users, with the evidence behind each one. Not a generic “tune the JVM” recommendation. The deeper how-to on each lever is in our Maximo performance optimisation guide. Where the same operational events are recurring — a stuck cron at 03:00 every Tuesday, a queue that backs up after every release — MaxIron Autoheal closes the loop on the boring ones automatically and safely, so operators stop being woken up to do the same thing every month.

2.5 5. Security and access

Often the most uncomfortable area. We work to a checklist, not a vibe.

- User and group inventory: dormant users, leavers still active, shared accounts, accounts with site-level admin no one remembers granting.
- Security group definitions: how many groups exist, how many are actually in use, how many give effectively-equivalent privilege under different names.
- Privileged access: who has system-administrator-equivalent rights, when those were last reviewed.
- Audit trail coverage: what is being audited, what is not, how long audit records are retained, whether anyone reviews them.
- Identity integration: SSO posture, MFA enforcement, where local Maximo passwords still exist and why.
- Data residency, encryption at rest and in transit, key management.
- A short read against ISO 27001 and (for UK regulated estates) the relevant regulator’s expectations.

Our deeper take on this is in the Maximo security and compliance guide.

2.6 6. Operability and platform health

How does the platform behave the day after we leave?

- Backup posture: what is backed up, how often, when was the last successful restore test.
- Monitoring and alerting: what is monitored, where the alerts go, who acts on them, what the false-positive rate looks like.
- Patching cadence: what version of MAS, OpenShift and OS, when each was last updated, what the gap to current is.
- Environment topology: how many non-prod environments, are they in sync with production, when was the last refresh.
- Change control: how changes are deployed to production, who approves, whether there is an audit trail.

- Runbooks: do they exist, are they current, can a new engineer act on them at 03:00.

This is the area that decides whether the next twelve months are quiet or noisy. On managed engagements, this is where MaxIron's product set carries the most weight: Cloud Manager for one estate-wide control plane across whichever clouds you run on, Sentinel for always-on synthetic and platform-signal monitoring, Diagnose for two-minute root cause when something fails, and Change Control so every change has an approval and an audit trail. Together they turn operability from a runbook problem into an operating-model property of the platform.

2.7 7. Upgrade-readiness

If MAS is on the roadmap, the upgrade-readiness view sits at the end:

- Customisation profile: number of Java customisations, automation scripts, custom screens, custom integrations. What is genuinely needed, what is leftover.
- Configuration vs customisation balance: how much of the system is configuration that ports cleanly, how much is custom code that needs assessment.
- Database vendor and version: implications for the target MAS topology.
- Data volumes and history retention: what is the upgrade window cost.
- Integrations that will need re-pointing or re-authenticating against MAS.
- Skills and training gap on the receiving team.

Combined with our MAS upgrade checklist, this gives a realistic upgrade scope and timeline.

3. The MaxIron product set behind the health check

Most partners hand a customer a PDF and walk away. We run health checks on top of our own product set, which means the audit is not a snapshot — it is the start of a continuous health regime, with the same products doing the finding and the fixing.

- **MaxIron Sentinel** — always-on Maximo health monitoring. Synthetic checks against the journeys users actually depend on (login, work order create, asset search, mobile sync, integration round-trip) plus the platform signals that matter (JVM, queues, cron, database). During a health check, Sentinel is the source of truth for what is genuinely slow or broken in production at peak. After the engagement, it stays on and the health check becomes a live dashboard, not a six-monthly memory.
- **MaxIron Diagnose** — root cause in two minutes. Turns a Sentinel alert (or a user complaint) into a clear cause, a clear location and a clear next action. This is what cuts MTTR by a factor we measure in hours, not percentages, and what makes the operability finding ("nobody knows where to look when it breaks") go away as a finding.
- **MaxIron Autoheal** — automated remediation for the recurring operational events Maximo throws up: stuck cron tasks, blocked queues, integration retries, expired

tokens, the usual list. The boring third of an operations team's week disappears, and the health check stops re-finding the same five issues every cycle.

- **MaxIron AI Smart Data** — purpose-built AI for getting data into and out of Maximo. In a health check it surfaces the data-quality problems that a manual audit would miss in two weeks. Afterwards, it is the same product that does the cleanup at scale — duplicates, hierarchy gaps, classifications, missing failure codes — with audit. Healing the data, not just naming the problem.
- **MaxIron Cloud Manager** — one control plane for every Maximo and MAS environment a customer runs, across whichever cloud they use. For a multi-estate health check, this is what makes "the same audit, on every estate, with comparable evidence" feasible at all.
- **MaxIron Change Control** — governed approvals and a clean evidence trail for every change going into production Maximo. The remediation actions out of a health check are themselves changes; doing them through Change Control means the customer ends the engagement with a stronger change posture than they started, not a weaker one.
- **MaxIron Blueprint** — the curated configuration baseline distilled from years of MaxIron deliveries. Where the health check finds genuine configuration drift (KPI definitions that no longer match the business question, role start centres no one uses, broken workflows), Blueprint is what we move the customer back towards, sector-aware, so the platform stops drifting in the first place.

The business story is the simple one: a traditional health check tells a customer where the platform is broken. A MaxIron health check uses the same products that find the problems to also fix them, and to keep finding them in production after the consultants have left. That changes the conversation from "what do we do with this report" to "what is the platform's health score this week, and what is healing itself automatically".

4. What the deliverable actually looks like

A health-check engagement produces three things, in this order:

1. A triaged findings register. Each finding has area, severity, evidence, recommended action, estimated effort, and a link to the underlying data extract or query that produced it.
2. A one-page executive summary that tells the senior sponsor whether the platform is in good, acceptable or fragile shape, and the three things that matter most to address.
3. A remediation roadmap, grouped into quick wins (fortnight), structural fixes (quarter), and platform-level work (programme). Each item has a cost band and a dependency note.

It does not produce a 200-slide deck nobody reads.

5. How long it takes

A focused health check on a single estate, with read-only access provided up front and named SMEs available, takes between ten and fifteen working days. Multi-estate or pre-acquisition due-diligence work takes longer because the integration and data-quality work multiplies per estate.

If a supplier offers to do a health check in a week without seeing the data first, the deliverable will be a generic checklist with the customer's logo on it. That is not what a health check is.

6. What we do next

Once the findings are accepted, the natural follow-on work is one of three engagements: a stabilisation programme where Sentinel, Diagnose and Autoheal are stood up against the issues the health check identified, an upgrade prepared on real evidence and run through Change Control and Pipelines, or a transition into managed hosting and operations where the entire MaxIron product set runs the platform continuously. The point of the health check is to make that next decision a small one, not a leap of faith.

If you are weighing up whether to commission one, see our Maximo Health Check & Heal service for what an engagement covers, the deliverable, and the products behind it — or talk to us directly. We can usually tell from a thirty-minute call whether a full audit is warranted or whether two or three targeted pieces of work — usually a Sentinel pilot, an AI Smart Data data-quality scan, or a focused performance review — would be a better starting point.