



# Putting Your AI Agents to Work Inside Maximo

with the Maximo Model Context Protocol (MCP) Server



## Highlights

- ❑ One standards-based endpoint that any MCP-capable agent can use
- ❑ AI configurations, automation scripts, workflows and object structures published as agent tools
- ❑ Authenticated tool calls that carry the requesting user's identity into Manage
- ❑ Bring your own agent alongside the IBM Maximo Assistant

An AI agent is only as useful as the systems that it can reach. In asset-intensive operations that system is Maximo and until now, every agent that needed to read a work order, classify a failure or start an approval needed its own integration, credentials, and security review.

The Maximo Model Context Protocol (MCP) server, introduced in Maximo Application Suite 9.2, removes that barrier. It exposes Maximo Manage data and functionality to AI agents as standardized tool calls over the MCP, the open standard for connecting language models to the systems they need to act on. Publish a capability once, and any MCP-compliant AI agent can seamlessly discover and use it, accelerating AI adoption while reducing integration complexity

## Understanding MCP as a Principle

MCP defines a single, vendor-neutral contract between AI agents and the applications they operate. A server publishes tools, which are named functions with a strict input and output schema and a plain-language description. The agent's planner reads those descriptions to pick a tool, calls it, reads the structured result, and decides what to do next. This ability to chain tool calls enables AI agents to complete sophisticated, multi-step tasks, moving beyond simple question-and-answer interactions.

The value is architectural. Capability is exposed once and centrally, with authentication and change control applied at the point of exposure rather than rebuilt inside every agent. Swap the agent, and the integration still stands.

In Maximo, a dedicated MCP server treats Manage as its single source of truth, refreshes its tool list as configurations change, and carries the requesting user's identity into Manage on every call it passes through.

## Benefits

 **Standard Integration:** One MCP route serves every compliant agent and eliminates the need for a bespoke connector per assistant and rework in a shifting AI landscape.

 **Governed Access:** API key or JSON web token at the edge, MAS certificates internally, and the requesting user's identity are carried into Manage. Tool-level control differs by type, so publish in line with the risk.

 **Reuse What You Already Have:** Activated AI configurations, active automation scripts, enabled workflows and object structures become agent tools that are built with Maximo skills your team already holds.

# What becomes an agent tool in Maximo

Every MCP tool is a Maximo REST call, but not every Maximo API is an MCP tool. Manage publishes five categories, each with its own schema and its own path to deployment.

| Tool type         | What it exposes                                                                                                                                               | Request & response schema                                                                                                              | Publishing and control                                                                                                              |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| AI Configuration  | Activated AI configurations on the supported inference list: classification (MCC/PCC), similarity, assistant (nl2oslc), document search and condition insight | Defined by the AI configuration itself; the supported list is held in a registry inside Manage                                         | Deployed automatically on activation, eliminating the need for a separate enablement step                                           |
| Object Structure  | Targeted slices of an object structure rather than its full unified schema, which would be too large for a model to consume reliably                          | Query templates, or the signature of a service @WebMethod; update tools gain an href element automatically                             | Explicit enablement that can be bound to a SigOption so Maximo security groups govern access                                        |
| Automation Script | Logic that already lives in a script: data shaped a particular way, a call to an external system, or a rule that no object structure operation covers         | In variables with Literal binding form the request; declare the response schema explicitly; validate required inputs inside the script | Enabled from the Automation Scripts app; active scripts only. Read-only and destructive annotations, but no SigOption binding today |
| Workflow          | Existing Maximo workflows, launched by an agent against a specific record                                                                                     | A single href identifying the target record                                                                                            | Explicit enablement. Active and enabled workflows only                                                                              |
| API Route         | IBM-supplied administrative and system APIs, such as API schema lookup                                                                                        | The underlying OpenAPI route                                                                                                           | Provided out of the box; not customer-configurable                                                                                  |

A tool's description is what the planner reads when choosing between tools, so write it as a routing instruction. Script tool descriptions are capped at 100M characters today. Confirm what will publish with GET /api/oas?aitools=1.

## Some practical examples of how to design tools that agents use well

- ✓

**Ask across the asset base:**  
"Which pumps at Site B have open corrective work above priority 3?" The assistant tool turns the question into an OSLC query and returns the matching records.
- ✓

**Classify and compare:**  
Classification and similarity tools propose a failure class for a new work order and surface the comparable jobs that were raised before it.
- ✓

**Write back, not just read:** An object structure or script tool reports downtime, updates a record or raises a follow-on work order.
- ✓

**Reach past Maximo and come back:** A script tool calls a trusted external service (e.g., weather, ERP stock, GIS), so an agent can flag a rain risk before a planner commits an outdoor job to a date.

Get in touch  
[www.cohesivesolutions.com](http://www.cohesivesolutions.com)



Cohesive is your trusted global advisor and system integrator for IBM Maximo. Owner-operators at asset-heavy organizations partner with us to bring together siloed processes and deliver insights to decision-makers at every level. As an IBM Platinum Partner, we help customers maximize the return on their Maximo investment, improving reliability, safety, and efficiency, extending asset life, and reducing risk.