

Tool Layer (MCP)

- [Tool Layer / MCP-Compatible Ecosystem](#)

Tool Layer / MCP-Compatible Ecosystem

Use standardized tool interfaces where possible. MCP is now the dominant standard for this layer, adopted beyond Anthropic by OpenAI, Google DeepMind, Microsoft, and thousands of teams, with roughly 97 million monthly SDK downloads and 19,000+ servers indexed across public registries as of early-to-mid 2026.

MCP 2026-07-28: What Changed

The 2026-07-28 specification (release candidate published May 2026, final expected end of July 2026) is the largest revision since MCP's launch. Key changes relevant to a production deployment:

- Stateless protocol core — servers no longer require sticky sessions or a shared session store; a remote MCP server can run behind a plain round-robin load balancer.
- MCP Apps extension — server-rendered UI surfaces delivered through the protocol itself (this is the mechanism behind interactive, in-chat tool UIs).
- Tasks extension — a call-now / fetch-later pattern for long-running tool operations.
- Authorization hardening — clients must validate the `iss` parameter per RFC 9207 to mitigate mix-up attacks; OpenID Connect `application\_type` declaration during Dynamic Client Registration; credentials bound to the issuing authorization server.
- Formal deprecation policy — the protocol can now evolve without breaking existing deployments.
- MCP Server Cards (in progress) — structured server metadata at a `.well-known` URL so registries and crawlers can discover a server's capabilities without connecting.

Recommended Standards

Standard	Purpose
MCP	Tool interoperability, resources, prompts, sampling, and now Apps/Tasks/Skills extensions

OpenAPI	REST tools
JSON Schema	Input validation
OAuth scopes / OIDC	Permission control and authorization